Current sensorless control of a boost-type switch-mode rectifier using an adaptive inductor model

by

Adrian Engel
Diplom-Ingenieur (FH), Hochschule für Technik und Wirtschaft Dresden, 2010

A Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

Supervisory Committee

Current sensorless control of a boost-type switch-mode rectifier using an adaptive inductor model

by

Adrian Engel
Diplom-Ingenieur (FH), Hochschule für Technik und Wirtschaft Dresden, 2010

Supervisory Committee

Dr. Subhasis Nandi (Department of Electrical and Computer Engineering)
Supervisor

Dr. Panajotis Agathoklis (Department of Electrical and Computer Engineering)
Unit Member

Abstract

Supervisory Committee

Dr. Subhasis Nandi (Department of Electrical and Computer Engineering)
Supervisor

Dr. Panajotis Agathoklis (Department of Electrical and Computer Engineering)
Unit Member

The present work describes the development of a control scheme for boost-type switch-mode rectifiers. While controllers for this circuit commonly use a shunt resistor or a magnetic field sensor to measure the instantaneous input or inductor current, here the inductor current is computed from the measured inductor voltage. This calculation requires knowledge of the physical properties of the inductor, most importantly its inductance, which are prone to change with operating conditions of the converter and throughout the lifetime of the inductor. The parameters of the inductor model are estimated during normal converter operation, and the inductor model is adapted accordingly. Simulation and experimental results confirm the effectiveness of the devised scheme in reducing the distortion of the input current.

Table of Contents

List of Figures

List of Tables

Lists of Symbols, Sub- and Superscripts, Diacritics and Abbreviations

Symbols

a	controller coefficient
$\underline{A}$	system matrix
b	controller coefficient
$\underline{b}$	input vector
B	magnetic flux density
$\underline{c}$	output vector
C	capacitance
d	differential operator
d, D	duty cycle
e	control error
f	frequency (of line voltage unless otherwise noted)
g	gating signal
G	transfer function
h	harmonic number
H	magnetic field strength
i, I	electric current
$\underline{I}$	identity matrix
j	imaginary unit
k	k factor, switching cycle number, sample number
K	controller gain
l	length
L	inductance
N	number of turns, register value

p, P	electrical power, real power
PF	power factor
Q	electric charge
R	electrical resistance
s	Laplace variable
S	apparent power
t	time
T	period (of line voltage unless otherwise noted)
u	control plant input
v, V	electric voltage
x, X	system input
$\underline{x}, \underline{X}$	state vector
y, Y	system output
z	z-transform variable
α	phase shift of the duty cycle function
Δ	difference
η	efficiency
θ	electrical angle
κ	conductance emulated by the converter
σ	standard deviation
τ	time constant
φ	phase angle
ω	angular frequency (of line voltage unless otherwise noted)

Sub- and superscripts

0	initially implemented
av	average
b	boost
c	controller, gain crossover, corner, coercive
C	capacitor
d	direct
D	diode, drain
est	estimated
f	frequency
F	forward
G	gate
(h)	harmonic number
imp	implemented
i	current
in	input
k	sample number, switching cycle number, k factor
L	inductor
m	margin (phase margin), mean (mean magnetic path length)
nom	nominal
o	output
OL	open loop
on	on-state
p	pole
pl	plant
pp	peak-to-peak
Q	switch (transistor)
r	remanent
rip	ripple

RRM	repetitive reverse maximum
s	sampling
S	source, surge
sw	switching
tr	true
v	voltage
z	zero
$*$	reference value

Diacritics

$\hat{}$	peak value
$\bar{}$	average over one switching cycle
$\bar{=}$	average where $t_{av} > T_{sw}$
$\tilde{}$	small-signal perturbation
$'$	voltage divided by voltage divider, current signal in form of shunt resistor voltage, gating signal output by the controller
$''$	signal at the operational amplifier output

Abbreviations

AC	alternating current
ADC	analog-to-digital converter
AWG	American wire gauge
CCM	continuous conduction mode
DC	direct current
DCM	discontinuous conduction mode
DSP	digital signal processor
ESR	equivalent series resistance
FFT	fast Fourier transform
FPGA	field programmable gate array
MOSFET	metal oxide semiconductor field effect transistor
PFC	power factor correction
RMS	root mean square
RHPZ	right half-plane zero
SR	slew rate
THD	total harmonic distortion
ZCD	zero crossing detection

Acknowledgements

Throughout the development process, my supervisor Dr. Nandi provided useful ideas. They majorly contributed to the success of this work. He also supplied the components for the converter prototype.

Dr. Agathoklis was always available for fruitful discussions and to give valuable tips.

1 Introduction

Many electric devices require direct current (DC) for their operation. Direct current can be obtained from the alternating current (AC) utility supply through rectification. A single-phase diode bridge rectifier outputs a voltage waveform that is rich in harmonics. In order to reduce the ripple in the output voltage, a large capacitor may be connected across the output of the circuit, see Figure 1.1. This in turn leads to a highly distorted input (line) current. Only when the rectified line voltage exceeds the capacitor voltage, power is fed into the circuit from the supply mainly to recharge the output capacitor as illustrated in Figure 1.2. During the intervals in which no line current is flowing, the diodes are reverse biased and it is the capacitor that provides power to the load.

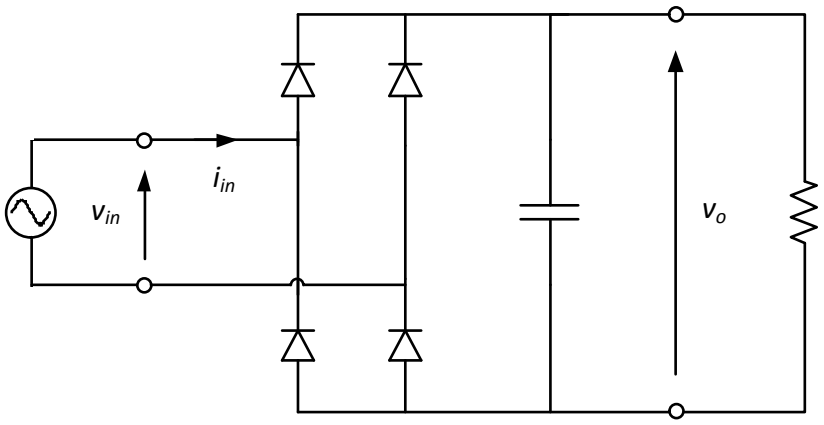


Figure 1.1: Uncontrolled bridge rectifier

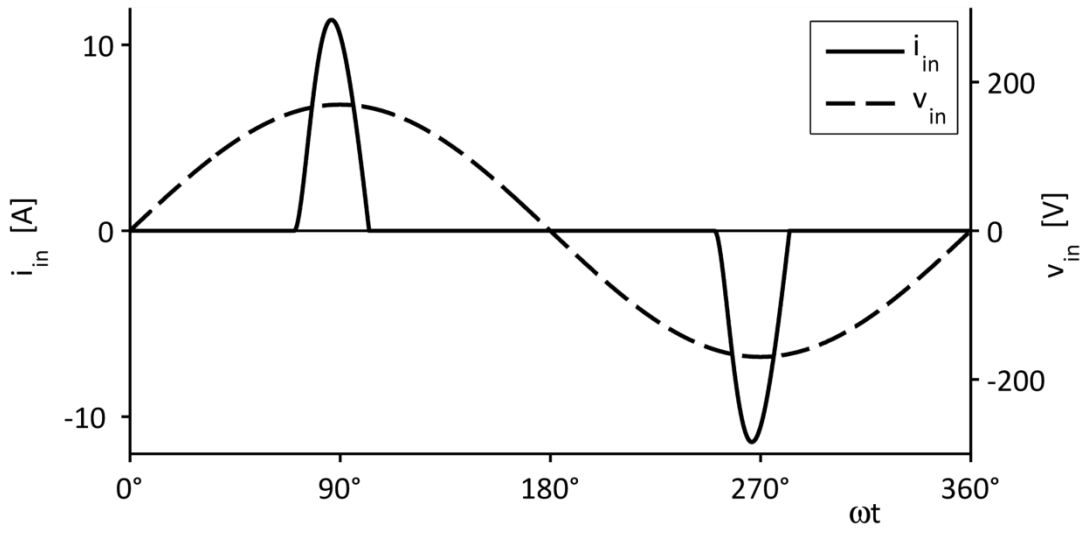


Figure 1.2 : Input voltage and input current waveforms of the uncontrolled bridge rectifier with capacitive load

Current waveforms with a high harmonic content are undesirable as input currents of appliances connected to the utility. They cause high stresses on the components of both the grid and the input stage of the device and result in a poor power factor. The power factor is defined as the ratio of the real power to the absorbed apparent power. In the presence of a sinusoidal (undistorted) utility voltage, only the fundamental of the current can contribute to the real power, and the power factor PF can be expressed via

$$\mathrm{PF} = \frac{P_{(1)}}{S} = \frac{V_{in}\, I_{in(1)} \cos\varphi_{(1)}}{V_{in}\sqrt{\sum_{h=1}^{\infty} I_{in(h)}^2}} = \frac{I_{in(1)} \cos\varphi_{(1)}}{\sqrt{\sum_{h=1}^{\infty} I_{in(h)}^2}} \tag{1.1}$$

Furthermore, other devices can be negatively impacted by the resulting distorted line voltage [1,2]. For this reason, regulations and recommendations exist that govern the permissible harmonic content of the line current drawn by consumer products and industrial equipment. Organizations that have issued standards on electromagnetic compatibility include the International Electrotechnical Commission (IEC) and the European Committee for Electrotechnical Standardisation (CENELEC).

One way to improve the strongly pulsating input current waveform from Figure 1.2 is to introduce an inductor on the DC side of the rectifier as shown in Figure 1.3:

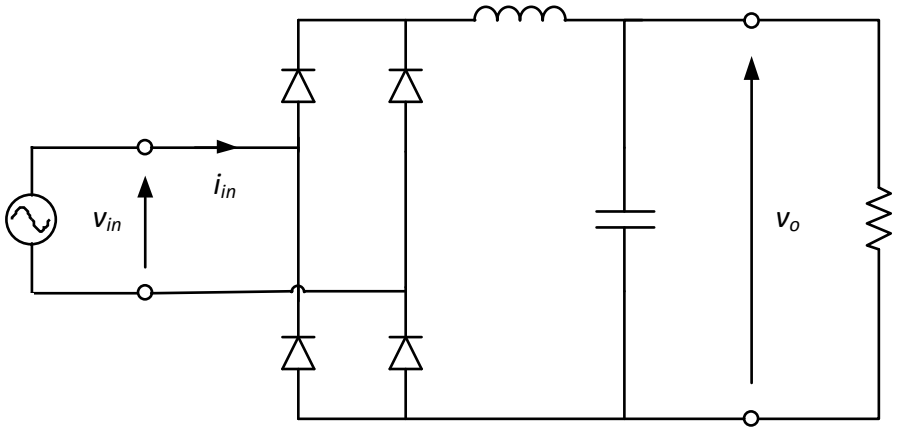


Figure 1.3: Uncontrolled bridge rectifier with DC side inductor

If the inductance is large enough, then the rectified input current becomes continuous. The resulting input current is plotted in Figure 1.4:

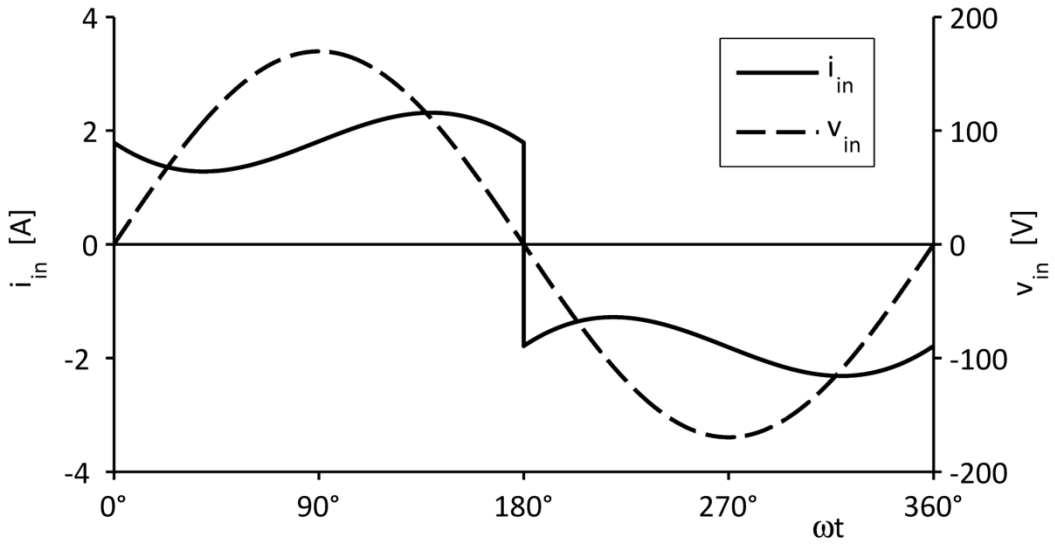


Figure 1.4: Input voltage and input current waveforms of the uncontrolled bridge rectifier with DC side inductor and capacitive load

Commutation of the inductor current from one pair or rectifier diodes to the other one takes place at the line voltage zero crossings. The larger the inductance is, the more the input current waveform approaches a square wave. The inductor eliminated the high peaks present in the current waveform in Figure 1.2. However, the input current is still strongly distorted. Furthermore, a large inductor makes the power supply bulky and react sluggishly to load changes.

Certain power electronic circuits are capable of actively shaping the input current waveform. In this way, it is possible to draw a line current that is nearly sinusoidal in shape and in phase with the line voltage. The power factor is thereby increased to almost unity, which is why the process is called power factor correction. Other terms used are active power factor correction, switch-mode rectifier, resistive input converter, pulse width modulation rectifier, unity power factor rectifier, and utility interface.

Power factor correction (PFC) converters belong to the group of switch-mode power converters. They employ power electronic switches that periodically alter the effective circuit topology. The switching is done in such a way that the input current, averaged over one switching cycle, has a sinusoidal shape. Then the converter behaves very much like a resistor from the grid's point of view, causes low distortion of the line voltage, draws mainly active power and hence complies with power quality standards. Superimposed on the fundamental component of the current is a ripple that is introduced through the switching. Figure 1.5 displays such a waveform where the switching frequency is very low for the purpose of illustrating the principle. As a result, the input current exhibits a large ripple. The switching frequency is usually in the kHz range, allowing the noise to be filtered out at the utility side with much smaller components compared to the large filter inductor in Figure 1.3 due to its high frequency.

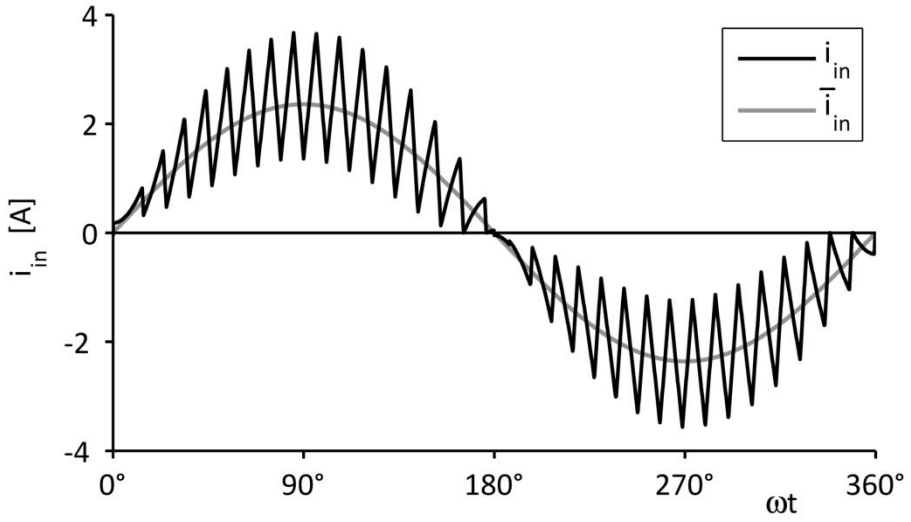


Figure 1.5: Input current waveform at low switching frequency

In brief, a switch-mode rectifier, as long as the line voltage and load stay within specified ranges,

1) converts an AC voltage into a DC voltage regulated to a certain value and
2) draws an input (line) current that is close to a sinusoid in shape and in phase with the line voltage.

Several circuit topologies exist which can be used to achieve these two goals. One of them is based on the DC/DC boost topology (Figure 2.1). This one is investigated in this work and described in the next chapter. The control scheme that is conventionally used for the circuit requires a current sensor. Sensing a current however might be undesirable in certain circumstances due to the required components. Various technologies have been proposed that are able to achieve the control action without the need for a current sensor. However, most of them come with the drawback of unsatisfactory accuracy.

The outcome of the undertaken research is a control scheme in which the inductor current is computed during operation and therefore need not be measured directly or indirectly. The control scheme uses an adaptive model of the boost inductor that is identified while the converter operates.

After a description of the converter that is to be controlled and a summary of published current sensorless control schemes in chapter 2, the development and operation of the designed control technique is described in chapter 3. The key feature in the realization of the accurate current sensorless control is the adaptive inductor model whose parameters are determined during regular converter operation. Simulation results are presented in chapter 4 that prove the feasibility of the method under a variety of operating conditions. Subsequently, a hardware prototype is built (chapter 5). Its performance, investigated in chapter 6, demonstrates the superiority of the devised technology relative to existing ones and verifies its usefulness.

2 Boost-type switch-mode rectifiers and their control

This chapter summarizes some characteristics of the investigated circuit and methods of controlling it. At first, important properties of the converter are examined and the conventional control scheme is explained in sections 2.1 and 2.2. Section 2.3 provides an overview of methods of current sensing. Then, in section 2.4, existing published techniques for controlling boost-type switch-mode rectifiers without current sensors are categorized and their principle of operation is briefly explained. Lastly, possibilities of determining relevant circuit parameters by the hardware and incorporating them into the control are presented in section 2.4.3.

2.1 Structure and pertinent properties of the circuit

Various circuit topologies are capable of providing a regulated DC voltage while drawing a sinusoidal line current. Overviews and comparative evaluations are provided in [3,4,5,6,7,8,9,10]. One suitable topology is the boost-type switch-mode rectifier that is used in this work. It consists of an uncontrolled diode bridge rectifier followed by a DC/DC boost converter as illustrated in Figure 2.1.

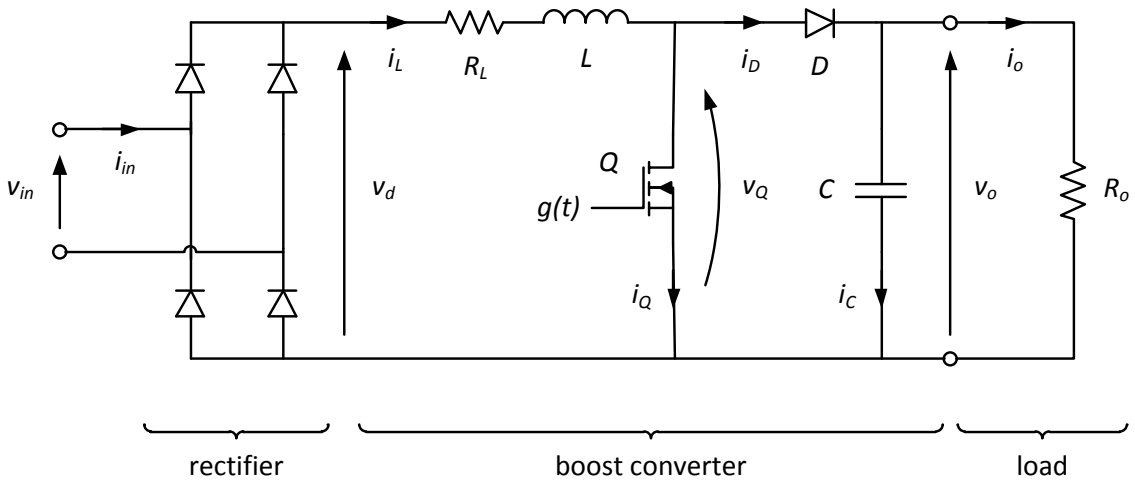


Figure 2.1: Boost-type switch-mode rectifier, simplified circuit diagram

This arrangement permits unidirectional power flow from the AC to the DC side. Energy recovery is not possible.

Advantages of the boost topology with respect to other DC/DC converter topologies are:

- Because the inductor is in the input branch of the boost part, the input current is inherently continuous as long as the converter operates in continuous conduction mode (CCM). This contributes to a low harmonic content of the input current.
- The gate driver need not be galvanically isolated because the source of the transistor is connected to ground.
- The transistor handles only a portion of input current.

The topology contains one power electronic device to be controlled, namely the transistor Q in the boost converter part. The two aims of control (regulating the output voltage V_o and shaping the input current i_{in} or the inductor current $i_L = |i_{in}|$) have to be achieved through the gating of the transistor with an appropriate pulse sequence $g(t)$.

To manage this, a multiple-loop control with two control loops can be implemented. An inner control loop is used to shape the input current to be

sinusoidal (apart from the ripple that is introduced through the switching). The outer control loop attempts to make the output voltage equal to its reference value. Refer to section 2.2 for a more detailed description of this concept.

The mentioned circuit topology contains a boost converter whose steady-state voltage conversion ratio is given by

$$\frac{V_o}{v_d} = \frac{1}{1-d} \tag{2.1}$$

Consequently, the output voltage is always larger than the peak of the rectified line voltage. In order to obtain lower output voltages, a second DC/DC converter may be used in cascade that is able to step down the voltage which the boost stage outputs. The resulting arrangement can be seen in Figure 2.2. This additional stage may also provide galvanic isolation between the AC line and the load by means of a transformer. It is further able to regulate the output voltage more tightly than the boost stage. Finally, the addition of energy storage devices also increases the converter hold-up time. In such a setup, the bridge rectifier and boost converter may together be referred to as power factor correction preregulator; this part realizes the input current shaping—thus accomplishing power factor correction—and provides a loosely regulated (preregulated) DC voltage to the step-down DC/DC converter stage, which then produces a tightly regulated DC voltage of the desired level [2,11].

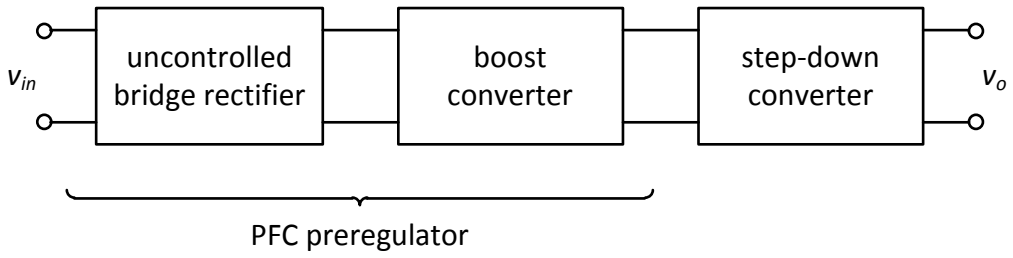


Figure 2.2: Switch-mode rectifier with step-down converter

PFC preregulators are also used to provide a direct voltage to inverters, the combination of the two forming a frequency converter.

The converter is oftentimes designed to operate in continuous conduction mode over most of the line cycle under rated conditions. At the beginning of a line half-cycle, however, it may happen that the inductor voltage cannot immediately follow its reference, as the available voltage across the inductor is too low. If the transistor is continuously turned on for a few switching cycles after the line voltage zero crossing, then

$$v_d = L\,\frac{di_L}{dt} \geq L\,\frac{di_L^*}{dt}\bigg|_{t=0} = L\,\hat{I}_L^*\,\omega \tag{2.2}$$

must be fulfilled for an undistorted line current. Equation (2.2) neglects the voltage across the equivalent series resistance of the inductor, which is a reasonable simplification given that the input current is very small around the line voltage zero crossings.

Depending on the design of the inductor, which can also be influenced by other considerations, (2.2) might be violated for a brief period. Then the inductor current will rise noticeably slower than its reference—it exhibits cusp distortion [12,13]. Cusp distortion of an acceptable level can be observed in Figure 4.2.

Care must be taken if the current controller (see section 2.2) includes integral action. In this case, during the time when the inductor current is not able to track its reference, the current error is integrated by the controller and may lead to an overshoot, possibly followed by oscillations, after i_L catches up with i_L^*.

The boost converter exhibits a distinctive feature in its transient behaviour that can pose difficulties for its control. When, for example, the load changes such that it demands a higher current, the output voltage initially starts to decrease. The output voltage controller uses this change to command a higher input current. This is achieved via a higher duty cycle. A higher duty cycle d entails a longer on time of the transistor Q. When however the transistor is turned on for a longer time, less current is transferred from the inductor L to the output because the diode D only conducts when the transistor is off, or

$$I_D = I_L\,(1-d) = I_L - I_Q \tag{2.3}$$

I_D, I_L and I_Q are the mean values of the diode, inductor and switch currents respectively. Therefore, while the higher duty cycle causes the inductor current to ramp up, the output voltage drops further. Only once the inductor current has risen does the diode current follow this trend; the now increased inductor current recharges the output capacitor and is able to provide the required higher load current. In brief, the response of the output voltage to the change in duty cycle is at first opposite to what ultimately happens. Likewise, a drop in the load current calls for lowering the duty cycle, which in the beginning drives more current through the diode and into the load. [14,15]

This phenomenon gives rise to a right half-plane zero (RHPZ) in the control to output voltage transfer function $\frac{\tilde{v}_o(s)}{\tilde{d}(s)}$ of the converter's small-signal model (see its derivation in section 3.8.1). Plants with an RHPZ are non-minimum phase systems [16,17]. Their behaviour is difficult to compensate, in particular because the frequency of the RHPZ of the boost converter is strongly dependent on the duty cycle and load resistance. A few design guidelines are provided in [18]. More often, though, the output voltage is regulated indirectly by means of the strategy explained in the following section.

2.2 Conventional control of switch-mode rectifiers

Switch-mode rectifiers are typically controlled using closed-loop control, also known as feedback control. The principle of closed-loop control is to measure the actual output value of the controlled variable y and compare it with the desired value (reference, command) y^*. The difference between the two, the error e, is then used to adjust the process in such a way as to correct the deviation between the reference and the actual output value. Figure 2.3 illustrates this concept [16].

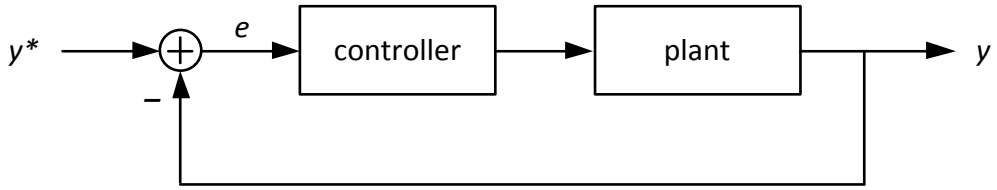


Figure 2.3: Structure of a classical closed-loop control system [16]

The classical control scheme for switch-mode rectifiers employs two cascaded feedback control loops; one for each of the control objectives (input current shaping and output voltage regulation). The controlled variables of the two loops are the converter output voltage and inductor current (rectified input current) respectively. First, the output voltage controller (error amplifier), depending on the difference between the reference and actual output voltage, determines the amplitude $\hat{I}_L$ of the inductor current that needs to be drawn from the supply to reach or sustain the desired output voltage. Based on this value and the sensed rectified line voltage v_d, the current controller computes the reference for the inductor current i_L^* as

$$i_L^* = \kappa \cdot v_d \tag{2.4}$$

Hence, i_L^* is determined by v_d in shape ($i_L^* \propto v_d \propto |\sin \omega t|$) through the control objective and by $\hat{I}_L$ in amplitude through the voltage controller. Together with the measured inductor current, the current controller is able to output the required duty cycle d for driving the transistor. The sequence is depicted in Figure 2.4. The duty cycle is then translated into a pulsewidth modulated gating signal g by comparing it with a sawtooth wave of the desired switching frequency.

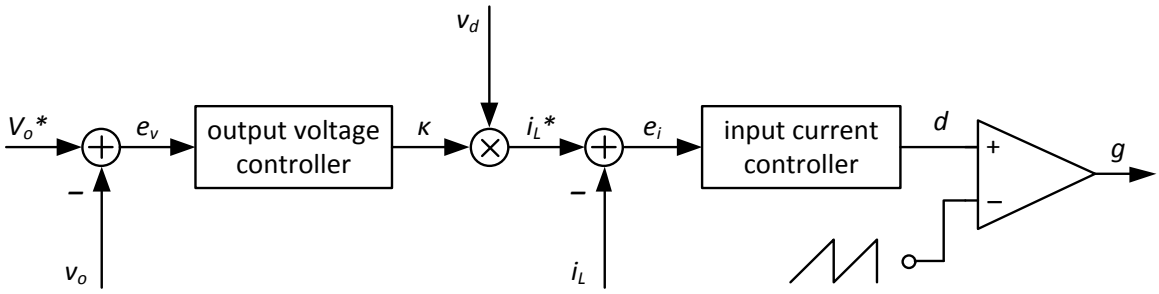


Figure 2.4: Conventional PFC converter control scheme

A fundamental feature of the two-loop control scheme is the large difference in the bandwidths of the two control loops. The bandwidth of the current control loop is several orders of magnitude higher than the one of the voltage control loop. Reasons for and consequences of this are explained in sections 3.7 and 3.8.

2.3 Methods of current sensing

The control scheme presented in the previous section requires measuring three quantities:

- the line voltage or, alternatively, the rectified line voltage,
- the converter output voltage and
- the inductor current.

While sensing voltages with a common reference potential poses little difficulty to both analog and digital implementations, the current measurement has to be performed indirectly. All methods presented below have in common that the current signal is converted into a voltage signal which is used by the controller.

Out of the variety of possibilities for sensing current, three general approaches are practicable for converters that have a rated power of not more than a few kilowatts, draw input currents of up to a few amperes and are suitable for mass production:

- measuring the voltage across a known resistance in the current path,
- using a magnetic field sensor and
- to determine the current through an inductor: using an RC series connection in parallel with it.

Different variations of the three are briefly discussed in the following. Other options include Rogowski coils, current transformers, magnetoresistance sensors and fiber-optic current sensors. Exclusion criteria for these are their high cost, incapability of measuring direct currents, low accuracy, or unsuitability for small currents. [19,20,21]

According to Ohm's law, the voltage across a resistor is a direct measure for the current flowing through it, provided the temperature and other physical factors affecting the resistance remain unchanged. Using a shunt resistor with a low inductance and temperature coefficient is a simple and low-cost way of current sensing. However, the shunt resistor creates losses that lower the converter's efficiency. Additionally, the resistor losses have to be dissipated. Small resistance values on the other hand require high-gain amplifiers and yield a low signal-to-noise ratio. The last point is an issue in particular with trace sensing where the voltage across a conductor trace on a printed circuit board, rather than a discrete resistor, is used to measure the current.

The on state resistance of the transistor in the converter has also been proposed as a current sensing resistor. Within a limited range, a MOSFET exhibits an approximately linear characteristic I_D versus V_{DS} when turned on. One factor complicating the use of this property for current measurement is the ringing on the drain-source voltage v_{DS} that occurs at the switching instants. Its strong temperature dependence and wide variations in the on resistance from transistor to transistor also limit its use as a current sensing resistor.

Another very popular type of sensor for this application is the Hall effect sensor. Hall effect sensors inherently provide electrical isolation between the current to be measured and the output signal. This feature gives greater flexibility in arranging the circuit configuration. The measurement accuracy is high. They are also often found in combination with other principles to combine various advantages. Hall sensors are much more expensive electronic components than shunt resistors. This and the fact that that they cannot be integrated as highly due to their large volume make them less desirable in mass applications.

A major disadvantage of both measuring techniques is temperature drift from which accuracy suffers.

A series connection of a resistor and a capacitor in parallel with an inductor may serve to determine the current through the inductor. If both branch time constants

are equal, the voltage across the capacitor is a measure for the inductor current. Alternatively, an equivalent operational amplifier circuit can be used. This method suffers from the fact that the time constants are difficult to match due to tolerances and variations in the circuit element parameters [22,23,24,25,26,27]. See section 2.4.1 for more details on this concept.

Because of the above-mentioned disadvantages it entails, efforts have been made to eliminate the need for sensing any current in the converter to determine the duty cycle. The existing methods are summarized in the following section.

2.4 Review of current sensorless control schemes

While the conventional control scheme for boost-type switch-mode rectifiers requires sensing the inductor current, there are reasons to avoid this necessity in some applications (see previous section). Various methods that do not need to measure the current to carry out the control action have been proposed. This chapter reviews the existing possibilities of current sensorless control for switch-mode rectifiers employing a boost converter. Some of them, presented in section 2.4.1, eliminate the need for sensing the current by determining it from measured voltages. Others by default do not require the knowledge of any current, removing the current control loop altogether (section 2.4.2).

When the instantaneous current is not measured, the sensitivity of the control performance to changes in the plant parameters becomes a major issue. Section 2.4.3 describes approaches to avoid degradation of the control outcome.

2.4.1 Methods that retrieve information about a current from measured voltages

The method presented in [28,29] is in large parts a digital implementation of the conventional control scheme with an input current and an output voltage control loop as set out in section 2.2. However, there is no current sensor. Instead, the inductor current is digitally computed and this computed current is used in the

current control loop. The rectified line voltage and the output voltage are measured. The inductor current is computed using these two signals and the gating signal $g(t)$. It is known that

$$\frac{di_L}{dt} = \frac{v_d}{L} \tag{2.5}$$

when the switch is turned on and

$$\frac{di_L}{dt} = \frac{v_d - V_o}{L} \tag{2.6}$$

when the switch is off. The two equations serve to compute the current for the (k+1)th sampling instant through discrete-time integration:

$$i_L(k+1) = \begin{cases} i_L(k) + \dfrac{v_d(k)}{L} T_s & \text{for } g(k) = 1 \\ i_L(k) + \dfrac{v_d(k) - V_o(k)}{L} T_s & \text{for } g(k) = 0 \end{cases} \tag{2.7}$$

These equations are implemented in a field programmable gate array (FPGA). The control scheme is claimed to be insensitive to tolerances in the inductance.

In [30], a detection of discontinuous conduction mode (DCM) is added and the necessary adjustments are made in the algorithm so that the converter can fulfil its function at lighter loads when its operation mode switches to discontinuous conduction mode around to the zero crossings of the line voltages (mixed conduction mode).

It is also possible to use a hardware network to obtain a signal that is proportional to the inductor current as briefly outlined in section 2.3 [27]. A series connection of a resistor and a capacitor is connected in parallel with the inductor, see Figure 2.5.

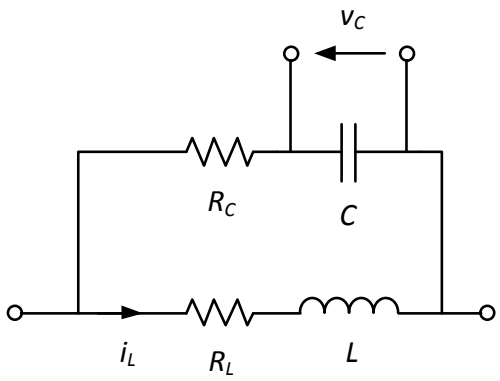


Figure 2.5: Hardware estimator for the inductor current

Because the voltages across L and R_L as well as C and R_C are equal,

$$\begin{aligned} v_{R_L} + v_L &= v_C + v_{R_C} \\ R_L i_L + sL\, i_L &= v_C + R_C\, sC\, v_C \\ i_L\,(R_L + sL) &= v_C\,(1 + sR_C C) \end{aligned} \tag{2.8}$$

Now the relation between i_L and v_C can be expressed through

$$\frac{v_C(s)}{i_L(s)} = \frac{R_L + sL}{1 + sR_C C} = R_L\,\frac{1 + s\,\frac{L}{R_L}}{1 + s\,R_C C} = R_L\,\frac{1 + s\tau_L}{1 + s\tau_C} \tag{2.9}$$

If $\tau_C = R_C C$ is chosen so that it matches $\tau_L = \frac{L}{R_L}$, the relation $\frac{v_C}{i_L}$ becomes frequency independent, signifying that the sensing preserves the shape of the inductor current. Under this condition, v_c is proportional to the current of interest and can be used in the current control loop.

2.4.2 Methods that do not require current signals

In [31,32,33,34,35], the duty cycle is determined from the rectified line voltage and the output voltage. A duty cycle function $d(\theta)$, $\theta = \omega t$ for one line half-cycle, the period of the rectified line voltage, is derived. The converter is considered ideal. Then the inductor current is determined by the voltage difference between the rectified line voltage and the voltage across the switch $\overline{v}_Q$

$$L\frac{d\bar{i}_L}{dt} = \overline{v}_d - \overline{v}_Q \tag{2.10}$$

in which the overbar denotes averaging over one switching cycle. For $\bar{i}_L$, for example, this means

$$\bar{i}_L(t) = \frac{1}{T_{sw}} \int_{t-T_{sw}}^{t} i_L(t_1)\, dt_1 \tag{2.11}$$

The voltage across the switch is given by

$$\overline{v}_Q = (1-d)\, V_o \tag{2.12}$$

It is desired that the inductor current $\bar{i}_L$ is proportional to $\overline{v}_d$,

$$\bar{i}_L = \kappa_1 \cdot \overline{v}_d \tag{2.13}$$

By plugging eqs. (2.12) and (2.13) into (2.10), the duty cycle function is found to be

$$d = 1 - \frac{\overline{v}_d}{V_o} + \frac{\kappa_1 L}{V_o}\frac{d\overline{v}_d}{dt} \tag{2.14}$$

The work reported in [32,33,34,35] implements a differentiator in the controller such that

$$d(\theta) = 1 - \frac{\overline{v}_d}{V_o} + \kappa_2 \frac{d\overline{v}_d}{d\theta}, \qquad \theta \in [0, 360°) \tag{2.15}$$

in which κ_2 is the output signal of the output voltage controller and proportional to the input current.

Reference [31] on the other hand assumes that the line voltage is purely sinusoidal such that $v_d \propto |\sin\theta|$. This simplification allows the substitution of the derivative of the rectified line voltage for each line half-cycle with a cosine function multiplied by a different factor $\kappa_3 \propto \kappa_2$:

$$d(\theta) = 1 - \frac{\overline{v}_d}{V_o} + \kappa_3 \cos\theta\,, \qquad \theta \in [0, 180°) \tag{2.16}$$

A similar technique is presented in [36]. This one also accounts for the inductor equivalent series resistance R_L and the voltage drop V_F across each diode in the rectifier and both the diode and the transistor in the boost converter. The duty cycle function in this case is

$$d(\theta) = 1 - \frac{\overline{v}_d}{V_o^*} + \frac{\sqrt{2}V_{in}}{V_o}\cos\theta + \frac{R_L}{\omega L}\frac{\sqrt{2}V_{in}}{V_o^*}\sin\theta + \frac{3V_F}{V_o^*}, \quad \theta \in [0, 180°) \tag{2.17}$$

The paper includes a detailed study of the behaviour of the converter with distorted input voltage. Especially at low power, this control scheme yields input current waveforms with a low distortion.

The approach described in [37] and [38] senses nothing but the line voltage. The reason that no output voltage regulation is required here is that the researchers assume a constant load and ideal circuit components. Like this, the generally output current dependent factors κ_2 in eq. (2.15) and κ_3 in eq. (2.16) can be determined with only v_d being measured:

$$d(\theta) = 1 - \frac{\overline{v}_d}{V_o^*} + \frac{\sqrt{2}V_{in} \cdot \omega L}{V_o^* \cdot R_o}\cos\theta\,, \qquad \theta \in [0, 180°) \tag{2.18}$$

The cosine function is realized through a 90° phase shift of the line voltage. Since the converter lacks a closed control loop, it is very sensitive to variations in the inductance and load resistance. When the load changes, the output voltage remains constant (provided the converter stays in CCM), but the line current will be distorted. Through the input voltage feedforward in eq. (2.18), the duty cycle automatically decreases when the input voltage rises (the term $\frac{\overline{v}_d}{V_o^*}$ is dominant) and vice versa. The line voltage feedforward also causes the input current to remain sinusoidal when the line voltage is distorted.

References [39,40,41,42,43] describe a different approach for implementation in a signal processor. It relies on duty cycle patterns $d(\theta)$ or $d(kT_{sw})$, where k denotes the switching cycle and T_{sw} is the switching period. Such sets of duty cycle patterns are computed in advance for one line half-cycle, stored in a memory chip and recalled during operation of the converter in each cycle of the rectified line voltage.

The computation of $d(kT_{sw})$ in [39] and [40] is based on the fact that the peak inductor current within a switching cycle $i_L(kT_{sw}+dT_{sw})$ deviates from its sinusoidal reference $i_L^*(kT_{sw}+dT_{sw})=\kappa\,|\sin(kT_{sw}+dT_{sw})|$ by the same value as does the valley current $i_L([k+1]T_{sw})$ from $i_L^*([k+1]T_{sw})$, or

$$\begin{aligned}|i_L(kT_{sw}+dT_{sw})-i_L^*(kT_{sw}+dT_{sw})|\\ =|i_L([k+1]T_{sw})-i_L^*([k+1]T_{sw})|\end{aligned} \tag{2.19}$$

This relation for controlling the average current makes it possible to determine the correct duty cycle for each switching cycle k numerically. The duty cycle pattern is computed such that it makes the inductor current sinusoidal and no current control loop and current sensor are required. The duty cycle pattern is synchronized with the line voltage at each zero crossing. Eight sets of $d(kT_{sw})$ for different input voltages are precomputed and stored in the memory. One of them is selected by the voltage controller so as to regulate the output voltage.

Naturally, this method of voltage control only works properly at eight particular operating points. Reference [41] therefore refines the duty cycle computation idea by including a closed-loop output voltage control that works for any input voltage and load within a certain range. Whereas [39] and [40] assume lossless circuit elements in the converter, [41] considers the inductor's equivalent series resistance R_L, the on resistance of the transistor R_{on} and the voltage drop V_F across the diode. The output voltage ripple $v_{o,rip}$ is taken into consideration as well. Another difference is that the valley current of each switching cycle, $i_L(kT_{sw})$, and not the average current tracks the current reference. This approximation works well for high loads when the inductor current's ripple to average ratio is small. The formula for obtaining the duty cycle function $d(k)$ is the digital version of (2.15) with the mentioned non-idealities of the circuit elements included. The basic version for the duty cycle function of the ideal converter is

$$d(k)=1-\frac{v_d(k)}{V_o^*}+\frac{L}{V_o^*}\frac{i_L^*(k+1)-i_L^*(k)}{T_{sw}} \tag{2.20}$$

which upon including the circuit non-idealities becomes

$$d(k) = \frac{V_o^* + v_{o,rip}(k) + V_F + R_L i_L^*(k) - v_d(k) + \frac{L}{T_{sw}}[i_L^*(k+1) - i_L^*(k)]}{V_o^* + v_{o,rip}(k) + V_F - R_{on} i_L^*(k)} \quad (2.21)$$

Reference [42] determines the gating sequence in the following way: One converter that serves as a master module that is equipped with a current sensor and the conventional two control loops (see section 2.2). During the converter's operation, the gating sequence is recorded in the memory of the controlling microprocessor. This step is repeated for a total of five load resistances such that a table of pulse sequences is obtained (similar to the principle in [39] and [40]). Identical converters then can use these experimentally acquired duty cycle sequences and thus operate without a current control loop. A voltage control loop selects the set of stored data that minimizes the output voltage error. Adjustments are made iteratively until the output voltage error falls below a certain threshold.

A stored duty cycle pattern is also the basis for the work presented in [43]. Voltage regulation however is achieved via offsetting the stored $d(k)$ values as needed. Yet by doing so, the resulting input current becomes distorted. This can be understood for example from eq. (2.16). If done properly, the load regulation affects the waveform of $d(k)$ through the last summand, rather than retaining the wave shape and altering the offset. The result is that the voltage regulation deteriorates the input current waveform.

A totally different control scheme is described in [44]. The output of the voltage control loop is a phase angle α that shifts a duty cycle pattern. The duty cycle pattern is such that the line current automatically becomes sinusoidal and shifting it regulates the output voltage. Because there is no current loop, no current sensor is required. For an ideal converter the duty cycle is given by

$$d(\theta) = 1 - \frac{\sqrt{2}V_{in}}{V_o} |\sin(\theta - \alpha)|, \qquad \theta \in [0, 360°) \quad (2.22)$$

When taking the circuit non-idealities into account [45,46], the duty cycle becomes

$$d(\theta) = 1 - \frac{\sqrt{2}V_{in}}{V_o}\,|\sin(\theta - \alpha)| + \alpha\frac{\sqrt{2}V_{in}R_L}{\omega L \cdot V_o}|\sin\theta| + \frac{3V_F}{V_o}, \qquad \theta \in [0, 360°) \tag{2.23}$$

where V_F is the forward voltage drop across each diode in the rectifier and both the diode and the switch in the boost converter. Mainly the shape of the term $\frac{\sqrt{2}V_{in}}{V_o}|\sin(\theta - \alpha)|$ contributes to the generation of a sinusoidal input current. From eq. (2.22) it becomes apparent how the voltage across the switch is delayed with respect to the line voltage to create the current flow. The last two terms in eq. (2.23) act as feed-forward signals to compensate the effects of the voltage drop across R_L and the forward voltages V_F. Noting that α is small, less than 5°, the input current is derived as $\bar{i}_{in} = \frac{\sqrt{2}V_{in}\,\alpha}{\omega L}\sin\theta$, from which it can be observed that the RMS value $I_{in} = \frac{V_{in}}{\omega L}\alpha$ is proportional to the phase shift of the duty cycle function α. With the duty cycle from eq. (2.22) plugged into (2.12), the inductor voltage averaged over one switching cycle can be expressed as

$$\overline{v}_L = \sqrt{2}V_{in}\,[|\sin\theta| - |\sin(\theta - \alpha)|] \tag{2.24}$$

Thus the inductor appears to be connected between two voltage sources outputting rectified sine waves with equal amplitude and a small phase difference between each other. The phase shift is used to regulate the flow of real power into the converter while keeping the reactive power small. The problem of the implementation of the scheme in a signal processor is the limited resolution of α. Under light load, this leads to a higher distortion of the input current and it becomes harder to regulate the output voltage.

A control scheme that is in some aspects similar to [44,45,46] is presented in [47]. It uses the rectified line voltage and the output voltage to obtain the duty cycle. The controller has two loops:

- a real power loop that controls the power flow and thus regulates the mean value of the output voltage and
- a power factor angle loop that controls the phase angle of the input current.

The voltage across the transistor, averaged over one switching cycle, is adjusted to about the rectified line voltage in magnitude, but with a small phase shift. By means of the phase shift one can control the power flow towards the output and achieve changes in the average output voltage. In order to regulate the input current phase angle, thereby forcing the current to follow a sinusoidal reference, the phase of the output voltage ripple is analyzed and adjusted accordingly.

This control scheme is implemented in a digital signal processor. Two extended Kalman filters obtain precise values of the rectified line and output voltage. A Kalman filter is similar to a discrete-time observer. It uses a plant model to estimate state variables. It can be used to extract useful information from noisy measurement data (analog and quantization noise). Here it is used to determine the magnitude and phase values of the voltages. Its design is based on considerations regarding the properties of the expected noise.

Table 2.1 provides an overview of relevant properties of the described control schemes:

Table 2.1: Overview of current sensorless control schemes

	[28] [29] [30]	[31] [32]	[33]	[34] [35]	[36]	[37] [38]	[39] [40]	[41]	[42]	[43]	[44]	[45] [46]	[47]	[48]
does not require current sensing	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
takes circuit non-idealities into account	no	no	no	no	yes	no	no	yes	yes	no	no	yes	no	yes
is proven to be little sensitive to circuit parameter variations	yes[1]	no	yes[2]	no	no	no	no	no	no	no	no	yes[3]	no	yes[4]

[1] — claimed, but not verified with waveforms
[2] — analyzes input admittance for inductance variations
[3] — tests with non-nominal inductance and capacitance are conducted, results are passable
[4] — paper focuses on parameter variations in detail, some results are passable

2.4.3 Methods adapting to circuit parameters

One advantage of feedback control is that the output is to a certain extent robust with regard to tolerances and variations in the plant parameters. With the inductor current signal missing, important information that could be used beneficially for the control action is not available. Methods computing the inductor current from the inductor voltage through an inductor model therefore have to cope with the fact that the model might be inaccurate. Equation (2.7) for example describes the inductor by its inductance only. A more precise inductor model could also consider the equivalent series resistance (ESR) R_L. The inductor ESR mainly comprises of the winding resistance, and further models the core losses.

In mass production, especially the inductance is subject to variations, which come from the tolerances in the core material properties. Depending on the core material and manufacturer, tolerances of ± 8 or 10% are typical for commonly used iron powder cores, but uncertainties of up to +35/-25% may be present [49,50]. Besides, iron powder cores undergo thermal aging at elevated temperatures, an effect that results in a permanent decrease in the inductance and an increase in eddy current losses. The stability of the magnetic properties of the core is also impacted by physical stress and moisture [50].

A reason for changes in the winding resistance is the changing temperature of the wire. Core losses are, as mentioned above, subject to change throughout the inductor's lifetime. Different schemes that estimate one or both parameters during the converter's operation and use the results to adapt their inductor model have been published.

Reference [51] presents a buck converter that employs an RC network as shown in Figure 2.5 to determine the inductor current. R_C is realized as a variable resistor by means of a series connection of fixed resistors that can be individually shorted by the controlling FPGA. The inductor time constant $\frac{L}{R_L}$ is estimated after load changes. Characteristic for a mismatch between τ_L and τ_C is a slowly decaying transient in the sensed capacitor voltage after a load change. This situation is recognized with a

bandwidth-limited differentiator, and the RC network time constant is adjusted accordingly.

The technique described in [52,53] uses a transconductance amplifier (a differential amplifier with a current output) with a parallel connection of a resistor and a capacitor connected to its output. This circuit, called a G_mC filter, is a variable active first-order low-pass filter. Its cutoff frequency and DC gain are adjusted to the response of the inductor at startup and reset of the converter before the regular operation commences. To make the filter adjustable, said resistor is implemented with another transconductance amplifier that emulates the behaviour of a variable resistor. Both transconductance amplifiers can be controlled via their bias currents.

In order to tune the filter to the inductor's cutoff frequency in [52], a sinusoidal voltage is applied to the inductor. The inductor voltage is leading with respect to the current, while the low-pass filter introduces a phase lag. The bias current of the transconductance cell emulating a resistor is varied until the phase difference between the inductor current and the filter output voltage disappears. Then the filter's cutoff frequency matches the inductor's one. The way described in [53] uses a triangular voltage across the inductor and increases the amplifier's transconductance until the peak of the filter output voltage assumes a reference value.

The adjustment of the DC gain in both publications is done by applying a direct voltage across the inductor. By varying the bias current of the transconductance amplifier that serves to amplify the inductor voltage, the filter gain is changed until its output equals the reference voltage. An offset cancellation technique improves the accuracy during this process.

This approach of calibrating the inductor model necessitates additional circuitry. Extra hardware is needed for the test signal generators and the switches to apply the test signals to the inductor. Besides, the calibration is not possible during regular operation.

The drawback of computing the inductor current by integration of the inductor voltage is that the computation relies on the knowledge of the inductance and the equivalent series resistance of the inductor. In [54], a method for controlling a buck converter is proposed that implements the relation between the inductor voltage and current,

$$\frac{i_L(s)}{v_L(s)} = \frac{1}{sL + R_L} \tag{2.25}$$

in an FPGA with adjustable parameters L and R_L. To obtain these, a known test current is drawn from the output of the converter in addition to the load current at startup and subsequently at regular intervals. This allows calculating the parameters and calibrating the inductor model as shown in (2.25). With the applied test current, the difference between the expected and actual output of the filter is used to make the necessary adjustment to R_L in the first step. A second introduced load current step allows to obtain a value for the time constant $\frac{L}{R_L}$, from which L is determined.

The performance as it is illustrated in the publication by various waveforms makes the strategy appear a powerful technology to realize converter control without current sensing. It is shown to be able to recognize and adjust to tolerances in the parameters of the inductor with high precision. Beneficial is furthermore that the principle behind it is straightforward and that a minimum of additional hardware is needed for the implementation.

2.4.4 Summary

After a thorough literature research it was found that a number of current sensorless control approaches have been published for the boost-type PFC converter. All of them require the parameters from the inductor equivalent circuit, at least the inductance. They therefore only perform properly when the inductor model parameters are known. Concepts that include an adjustable inductor

model along with a strategy to determine its parameters have only been used with buck converters. Except in [51], the model identification is done offline.

2.5 Objectives of the undertaken research

The objective of the undertaken work was to devise a control scheme

- that neither requires a current sensor nor uses a shunt resistor to measure any current within the converter,
- that has a high accuracy in current shaping, i.e. the converter input current has a low distortion,
- whose performance is not dependent on prior knowledge of the boost inductor's inductance and equivalent series resistance,
- whose performance does not degrade when the inductance and equivalent series resistance change during operation and
- that does not require additional circuit components while avoiding the current measurement.

3 Development of the control scheme

In this chapter, the new control scheme for the switch-mode rectifier is developed. Initially, the design goals are stated and the main ideas to achieve them are given in section 3.1. Section 3.2 then describes the principle behind the estimation of the inductor current with the help of an inductor model under the assumption of known inductor parameters. Consequences of a mismatch between the parameters of the inductor and those of the inductor model are illustrated in section 3.3. The main novelty of this work is the adaptation of the inductor model to the true inductor parameters. The model adaptation consists of two steps: identification of the parameters (presented in section 3.4) and adjustment of the model (section 3.5). Section 3.6 then explains the modelling of the converter; the model is needed to design the current and voltage controllers, which is described in sections 3.7 and 3.8.

3.1 Design goals and considerations, general remarks

The new control scheme is developed with the aim of designing a method of control that does not require sensing any current. Current sensing in this context means obtaining a current signal by means of a magnetic field sensor or by measuring the voltage across a shunt resistor. In order to achieve the current-sensorless control, the current through the inductor is computed from the voltage across it. The performance of the devised control scheme should be insensitive to variations in the equivalent circuit parameters of the inductor. Both the abilities to shape the input current and to regulate the output voltage are not to be impacted if the inductance and equivalent series resistance of the inductor are not precisely known.

Using the inductor voltage as the signal to be measured as opposed to the current leads to further advantages. On the one hand, the inductor voltage is larger than the

output of a current sensor or the shunt resistor voltage usually would be. The measurement is therefore less sensitive to noise. On the other hand, the inductor voltage does not change much as a function of the load current [55]. The signal to noise ratio of the acquired signal thus profits even more at partial load.

The disadvantage with omitting the current sensing is that fault conditions that lead to overcurrent, such as overloads and short circuits, cannot be detected as reliably.

3.2 Computation of the inductor current

The inductor current as feedback signal for the current control loop is not available. The feedback control setup shown in Figure 2.3 can therefore not be used directly.

In the frequency range of interest, a real inductor can be modelled with sufficient accuracy through its inductance and its equivalent series resistance. Both equivalent circuit elements are connected in series as illustrated in Figure 3.1.

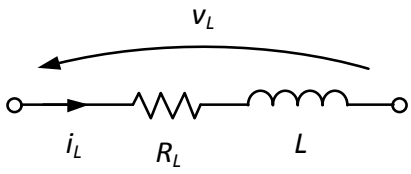


Figure 3.1: Inductor model

In this section, R_L and L are treated as known constants.

The terminal voltage and the current through the equivalent circuit Figure 3.1 are related via the equation

$$v_L = R_L i_L + L \frac{di_L}{dt} \tag{3.1}$$

Equation (3.1) can be transformed into the Laplace domain to become

$$v_L(s) = R_L\, i_L(s) + sL\, i_L(s) \tag{3.2}$$

and be expressed as a transfer function

$$G(s) = \frac{i_L(s)}{v_L(s)} = \frac{1}{sL + R_L} \tag{3.3}$$

This transfer function describes a first-order low-pass filter that imitates the behaviour of the network depicted in Figure 3.1. If v_L, L and R_L are known, the desired current i_L can be computed using the current-voltage relation from eq. (3.3).

In summary, the current-sensorless control method extends the conventional control method, Figure 3.2a, with an inductor model to compute the inductor current such that it needs not be measured (Figure 3.2b). An adaptive inductor model will be developed in section 3.5.

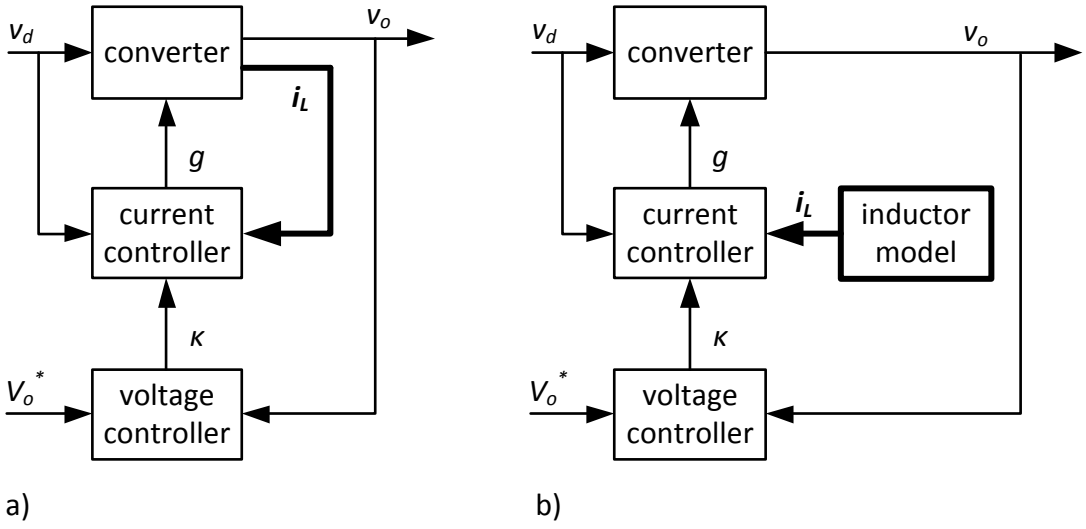


Figure 3.2: Conventional two-loop control scheme (a) and current-sensorless control scheme with computation of the current (b)

3.3 Effects of inductor model parameter mismatches

As pointed out in subsection 2.4.3, the true inductor parameters L and R_L can deviate from the model parameters due to material tolerances, core saturation,

changes in temperature and aging. When the true inductor parameters differ from the model parameters, the current through the inductor as response to the voltage across it will deviate from what the inductor model computes. The controller outputs the duty cycle based on the inaccurate computed current whereas the actual current obeys the hardware characteristics. Mismatches of the model parameters therefore result in distorted input currents. The waveforms of the input current for various cases have been investigated through simulation and are presented in this section.

Henceforth, the following terms and symbols will be used to refer to the different variants of the inductor parameters:

- The nominal inductance and ESR, L^{nom} and R_L^{nom}, is the theoretical value that the inductor is designed to or believed to have;
- the true inductance/ESR, L^{tr}/R_L^{tr}, is the value that the inductor actually has;
- the estimated inductance/ESR, L^{est}/R_L^{est}, is the value that the parameter identification scheme determines that the inductor has;
- the implemented inductance/ESR, L^{imp}/R_L^{imp}, is the value of which the inductor model makes use to compute the inductor current from the inductor voltage;
- the initially implemented inductance/ESR, L^0/R_L^0, is the value that the inductor model uses before the parameter adjustment is activated.

The results are plotted in Figure 3.3. The inductor has its nominal parameters of 8 mH and 0.6 Ω in all cases, whereas the implemented values vary from one half to twice the nominal values. One can distinguish two types of distortion of the input current. This behaviour can be understood from the inductor characteristic eq. (3.3) written in the form

$$\frac{i_L(s)}{v_L(s)} = \frac{1}{L}\,\frac{1}{s + \frac{R_L}{L}} \tag{3.4}$$

The waveshape of $i_L = |i_{in}|$ depends on the relation of the ratio $\frac{R_L^{imp}}{L^{imp}}$, the corner frequency of the model, to $\frac{R_L^{tr}}{L^{tr}}$, the corner frequency of the inductor:

1) $\frac{R_L^{imp}}{L^{imp}} < \frac{R_L^{tr}}{L^{tr}}$

When the model uses a corner frequency that is smaller than the true one, the input current reaches zero before the line voltage zero crossing and the converter remains in discontinuous conduction mode for the rest of the half-cycle. These cases are found in the plots below the diagonal in Figure 3.3. The peak inductor current increases because the output power remains unchanged.

2) $\frac{R_L^{imp}}{L^{imp}} > \frac{R_L^{tr}}{L^{tr}}$

Conversely, when the true corner frequency is smaller than the model assumes, the input current does not reach zero by the end of the line half-cycle. In that case, commutation from one pair of rectifier diodes onto the other one takes place the moment the line voltage changes its sign. This kind of distortion can be observed from the plots in Figure 3.3 above the diagonal.

3) $\frac{R_L^{imp}}{L^{imp}} = \frac{R_L^{tr}}{L^{tr}}$

In the case that the corner frequency of the model matches the corner frequency of the inductor, irrespective of the actual values of L^{imp} and R_L^{imp}, the input current is not distorted. These plots are the ones on the diagonal in Figure 3.3. Under this circumstance, the inductor model exhibits the proper frequency characteristic and outputs a signal that has the right waveshape, except that the amplitude is not correct (factor $\frac{1}{L}$ in eq. (3.4)). The amplitude of the input current, however, is regulated in a closed loop by the output voltage controller, which adjusts its output κ accordingly (Figure 2.4). Table 3.1 exemplifies this compensation mechanism for the cases from the diagonal of Figure 3.3:

Table 3.1: Model parameter mismatch when $\frac{R_L^{imp}}{L^{imp}} = \frac{R_L^{tr}}{L^{tr}}$

L^{imp} [mH]	R_L^{imp} [Ω]	$\frac{R_L^{imp}}{L^{imp}}$ [s^{-1}]	$\kappa \left[\frac{A}{V}\right]$	$\kappa \cdot L^{imp}$ [s]
4	0.3	75	0.028	$112 \cdot 10^{-6}$
6	0.45	75	0.019	$112 \cdot 10^{-6}$
8	0.6	75	0.014	$112 \cdot 10^{-6}$
12	0.9	75	0.0092	$111 \cdot 10^{-6}$
16	1.2	75	0.0069	$110 \cdot 10^{-6}$

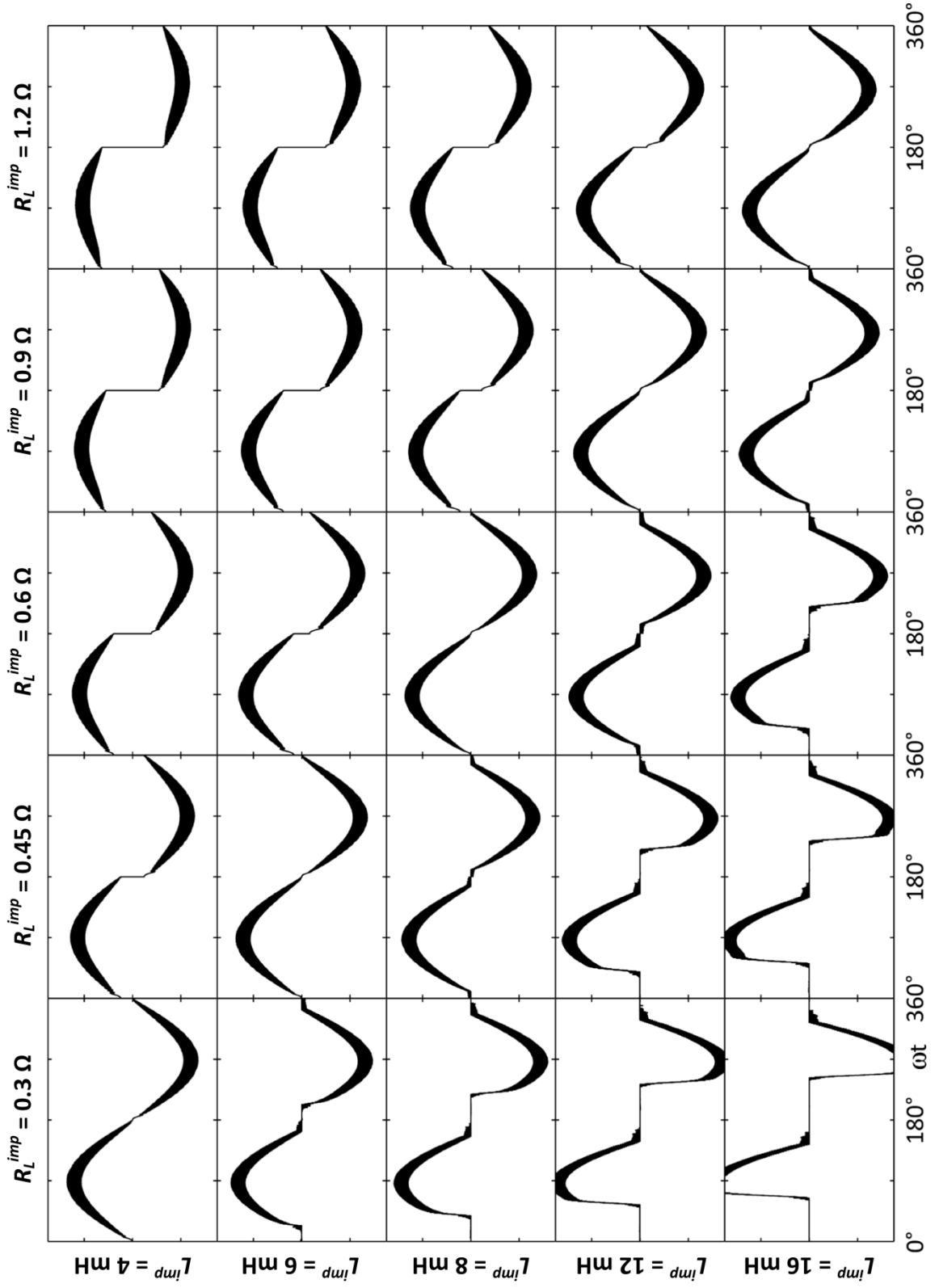


Figure 3.3: Input current waveforms for model mismatches; $L^{tr} = 8$ mH, $R_L^{tr} = 0.6\ \Omega$

3.4 Estimation of the inductor parameters

The calculation of the inductor current relies on the inductance and equivalent series resistance of the inductor. Because these parameters can vary, additional effort is put into estimating them. Computing estimates of the inductance and the equivalent series resistance is the main novelty of the present work. The estimation of the inductor parameters L and R_L is carried out continuously when the converter operates in steady-state. The results are then used to parametrize the inductor model described in section 3.2 to calculate the inductor current. In this section, the formulas that are used to determine L and R_L are derived.

To find the two values, the inductor voltage is integrated over time for one half and one full period respectively. Averaged over one switching cycle, the inductor voltage has a cosine-like waveform in each cycle of the rectified line voltage as drawn in Figure 3.4.

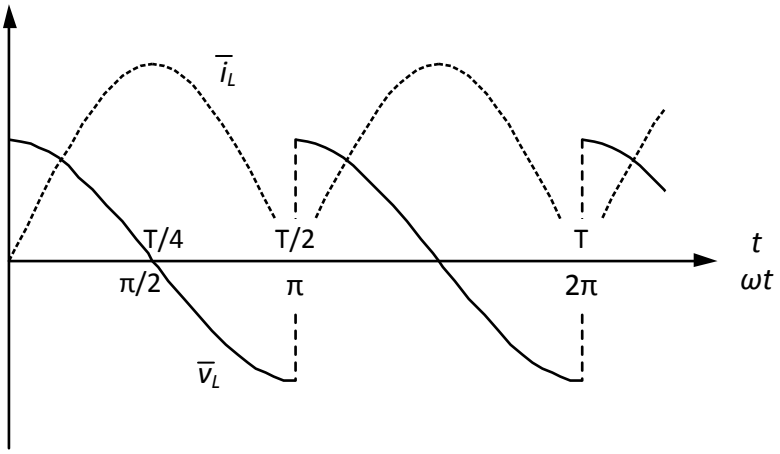


Figure 3.4: Inductor voltage and current waveforms, averaged over one switching cycle, when $R_L = 0$

Writing the current-voltage relationship of the inductor equivalent circuit eq. (3.1) in terms of the switching cycle averages to account for the conditions in the switch-mode converter yields

$$\overline{v}_L = R_L \bar{i}_L + L \frac{d\bar{i}_L}{dt} \tag{3.5}$$

The averaged inductor current $\bar{i}_L$ and its derivative can in the interval $\omega t \in [0, \pi)$ be expressed as

$$\begin{aligned} \bar{i}_L &= \hat{I}_L \sin \omega t = \hat{I}_L \sin \theta \\ \frac{d\bar{i}_L}{dt} &= \hat{I}_L \omega \cos \omega t = \hat{I}_L \omega \cos \theta \end{aligned} \tag{3.6}$$

Integration of the inductor voltage waveform in eq. (3.5) over time, where $t_{1,2} \in [0, \frac{T}{2})$, and inserting eq. (3.6) results in

$$\begin{aligned} \int_{t_1}^{t_2} \overline{v}_L \, dt &= \int_{t_1}^{t_2} \left[R_L \bar{i}_L + L \frac{d\bar{i}_L}{dt} \right] dt \\ &= R_L \hat{I}_L \int_{t_1}^{t_2} \sin \omega t \, dt + L \hat{I}_L \omega \int_{t_1}^{t_2} \cos \omega t \, dt \\ &= R_L \hat{I}_L \frac{1}{\omega} (\cos \omega t_1 - \cos \omega t_2) + L \hat{I}_L (\sin \omega t_2 - \sin \omega t_1) \\ &= \hat{I}_L \left[\frac{R_L}{\omega} (\cos \theta_1 - \cos \theta_2) + L (\sin \theta_2 - \sin \theta_1) \right] \end{aligned} \tag{3.7}$$

If the integration is carried out over one period of $\overline{v}_L$, i.e. one line half-cycle, then $t_1 = 0, \theta_1 = 0$ and $t_2 = \frac{T}{2}, \theta_2 = \pi$ and eq. (3.7) simplifies to

$$\int_0^{T/2} \overline{v}_L \, dt = \hat{I}_L \frac{R_L}{\omega} \cdot 2 \tag{3.8}$$

This expression can be solved to obtain the estimate of R_L, giving

$$R_L^{est} = \frac{\omega}{2\hat{I}_L} \int_0^{T/2} \overline{v}_L \, dt \tag{3.9}$$

Similarly, integrating $\overline{v}_L$ from 0 to $\frac{T}{4}$ such that $\theta_1 = 0$ and $\theta_2 = \frac{\pi}{2}$ in eq. (3.7) results in

$$\int_0^{T/4} \overline{v}_L \, dt = \hat{I}_L \left[\frac{R_L}{\omega} + L \right] \tag{3.10}$$

Solving for L yields

$$\begin{aligned} L^{est} &= \frac{1}{\hat{I}_L} \int_0^{T/4} \overline{v}_L \, dt - \frac{R_L^{est}}{\omega} \\ &= \frac{1}{\hat{I}_L} \left[\int_0^{T/4} \overline{v}_L \, dt - \frac{1}{2} \int_0^{T/2} \overline{v}_L \, dt \right] \end{aligned} \tag{3.11}$$

Eqs. (3.9) and (3.11) are the formulas that are used for the online estimation of R_L and L. At the end of the integration interval, the results of the integration are passed to a sample and hold element. The value is stored for further processing until it is overwritten by the subsequent integration result.

Because the inductor voltage is integrated over at least $\frac{T/4}{T_{sw}} = \frac{f_{sw}}{4f} = \frac{20\text{ kHz}}{4 \cdot 60\text{ Hz}} = 83$ switching cycles, it is expected that switching noise and other disturbances of higher frequencies have little effect on the computed values. Required variables that appear in the expressions are the averaged inductor voltage and the peak inductor current. The inductor voltage is also used by the inductor model. The peak inductor current has to be determined separately as described next.

It is possible to calculate the peak of the inductor current averaged over one switching cycle, $\hat{I}_L$, from the peak value of the second harmonic of the output voltage, $\hat{V}_{o(2)}$ [56]. Neglecting losses other than the ones occurring in the inductor ESR, p_{R_L}, the power balance of the converter is

$$p_{in} = p_o + p_{R_L} \tag{3.12}$$

Substituting

$$\begin{aligned} p_{in} &= \hat{V}_{in} \sin\theta \cdot \hat{I}_{in} \sin\theta \\ p_o &= v_o \bar{i}_D \approx V_o \bar{i}_D \end{aligned} \tag{3.13}$$

$$p_{R_L} = R_L \bar{i}_L^2 = R_L \hat{I}_L^2 \sin^2\theta$$

and noting that $\hat{V}_{in} = \hat{V}_d$, $\hat{I}_{in} = \hat{I}_L$ and $\sin^2\theta = \frac{1}{2}(1 - \cos 2\theta)$, (3.12) becomes

$$\frac{1}{2}\hat{V}_d \hat{I}_L (1 - \cos 2\theta) = V_o \bar{i}_D + \frac{1}{2} R_L \hat{I}_L^2 (1 - \cos 2\theta) \tag{3.14}$$

Equation (3.14) solved for $\bar{i}_D$ reads

$$\bar{i}_D = \frac{\hat{V}_d \hat{I}_L - R_L \hat{I}_L^2}{2V_o}(1 - \cos 2\theta) \tag{3.15}$$

The peak value of the 120 Hz component (second harmonic) of the diode current is $\hat{I}_{D(2)} = \frac{\hat{V}_d \hat{I}_L - R_L \hat{I}_L^2}{2V_o}$. The diode current is divided between the output capacitor and the load resistor. The current division ratio is the ratio of the admittances of load resistor and output capacitor. It is maximum at rated load and is given by

$$\frac{\frac{1}{R_o}}{\omega C} = \frac{\frac{1}{744\ \Omega}}{2\pi \cdot 120\ \text{Hz} \cdot 270\ \mu\text{F}} = 6.6 \cdot 10^{-3} \tag{3.16}$$

Since the share of $\bar{i}_{D(2)}$ that flows through the load resistor is very small, $\bar{i}_{D(2)}$ will be treated as if it flowed through the output capacitor entirely: $\hat{I}_{D(2)} \approx \hat{I}_{o(2)} = \hat{I}_C$. Under this assumption, the voltage-current relationship for the output capacitor $\hat{V}_{o(2)} = \frac{\hat{I}_C}{2\omega C}$ can be applied to obtain

$$\hat{V}_{o(2)} = \frac{\hat{V}_d \hat{I}_L - R_L \hat{I}_L^2}{4\omega C \cdot V_o} \tag{3.17}$$

whose feasible solution for $\hat{I}_L$ is

$$\hat{I}_L = \frac{\hat{V}_d}{2R_L} - \sqrt{\left(\frac{\hat{V}_d}{2R_L}\right)^2 - \frac{4\omega C \cdot V_o}{R_L}\hat{V}_{o(2)}} \tag{3.18}$$

The computation of $\hat{I}_L$ will have to be carried out during every switching cycle when the converter operates. Computation time is precious, and evaluating eq. (3.18) would put a heavy load on the processor. For this reason, R_L in eq. (3.17) is neglected, and then $\hat{I}_L$ can be found more straightforwardly as

$$\hat{I}_L = 4\omega C \, \frac{V_o}{\hat{V}_d} \cdot \hat{V}_{o(2)} \tag{3.19}$$

Unfavourable with this equation is the dependence of $\hat{I}_L$ on the output capacitance C. The capacitors that are used for this purpose are typically electrolytic capacitors. They have large tolerances in their capacitance, e.g. $\pm 20\%$ [57], and deteriorate further thoughout their lifespan as the electrolyte slowly evaporates. Dependence of the converter's current shaping capability on the exact capacitance value would be detrimental to the practicality of the developed control scheme.

If the inductor model is expressed in the form of eq. (3.4), the estimated model parameters appear as $\frac{1}{L^{est}}$ and $\frac{R_L^{est}}{L^{est}}$. As per eqs. (3.9) and (3.11), both the estimated inducance and inductor ESR are inversely proportional to the peak inductor current. Therefore, their quotient is unaffected by errors in $\hat{I}_L$. If the factor $\frac{1}{L^{est}}$ does not have its correct value because $\hat{I}_L$ is not accurately determined, this is compensated for by the output voltage controller as discussed in section 3.3. As a result, C need not have its nominal value for the current control to work. This was confirmed by computer simulations.

An alternative means of obtaining the peak inductor current is evaluating

$$\hat{I}_L = \kappa \cdot \hat{V}_d \tag{3.20}$$

that follows from the control objective eq. (2.4). However, simulation revealed that this way of computing the peak inductor current makes the parameter estimation, and in consequence the entire control of the converter, less stable. The chosen method is therefore the former one that uses eq. (3.19).

To estimate the inductor equivalent circuit parameters, the integral of the inductor voltage over one line half-cycle needs to be determined. The method is therefore inherently incapable of capturing changes in the inductance occurring within one line half-cycle. These are due to the nonlinearity and hysteresis of the core's $B(H)$ curve. In particular, when the core approaches saturation near the peak of the line voltage, the resulting decline in the inductance is not accounted for. The

equivalent series resistance meanwhile is mainly influenced by a varying temperature, and the inductor's thermal capacitance provides sufficient inertness to oppose quick changes thereof.

3.5 Inductor model parameter adaptation algorithm

The parameter identification is discontinuous with new results becoming available after every line half-cycle. Even when the converter operates in steady state, the simulation shows small fluctuations in the estimated inductor parameters from computation to computation. Those variances can be attributed to numerical errors. In the simulation, the variances are of less than 1%; they are expected to be larger in the actual converter where noise and the lower resolution of the variables in the digital controller are additional issues.

First-order low-pass filter elements are added after the computation of the inductor model parameters before they are passed on to the model. This step follows two goals. One is to feed the new values gradually to make the adaptation smooth rather than allowing abrupt changes. The filters turn the discontinuous estimated values into implemented ones that are continuous. Second, the low-pass filters reduce the dynamics of the parameter adaptation loops. Thus there is less interference between the parameter adaptation loops and the output voltage control loop because the coupling between the two is reduced. Suitable time constants for the filters were found to be 40 ms through simulation for both the adaptation of L and R_L.

The relationship between $\hat{I}_L$ and $\hat{V}_{o(2)}$ eq. (3.19) holds when the converter operates in steady state. During transients, e.g. after turn-on and load changes, the equation is not fulfilled. As a result, the parameter identification scheme returns false results that should not be used for the current computation by the inductor model. Parameter adjustment of the inductor model therefore only takes place when

the converter operates in steady state. Otherwise, the model continues to use the previous parameters until the transient has decayed.

3.6 Modelling of the converter

Transfer functions required for the design of both the current and the voltage controllers are derived using state-space averaging [1,58,59]. The state-space averaging approach is an established technique to obtain linearized models of switch-mode converters. Its underlying idea is to analyze the converter in all possible switch states (on/off) separately and then obtain an averaged model by calculating the weighted average of the individual configurations over one switching cycle depending on how long each state prevails. The resulting models are valid for frequencies much lower than the switching frequency.

To obtain concrete values e.g. for time constants and transfer functions, the parameters of the circuit components in the converter prototype are used. A detailed listing of these can be found in section 5.1.

The following assumptions about the converter are made to simplify the modelling:

- The converter operates in continuous conduction mode.
  It is designed to do so under rated conditions, but will enter discontinuous conduction mode near the zero crossings of the line voltage and at very light load.
- The inductor current can be expressed as a piecewise linear function.
  It is justified to make this assumption because the time constants of the inductor and of the output stage are much larger than the switching period. Under nominal conditions, the values are $\tau_L = \frac{L}{R_L} = \frac{8\ \text{mH}}{0.6\ \Omega} = 13\ \text{ms}$ for the inductor and $\tau_o = \frac{1}{R_o C} = \frac{1}{722\ \Omega \cdot 270\ \mu\text{F}} = 5.1\ \text{s}$ for the output; both are very large compared to $T_{sw} = 50\ \mu\text{s}$.

- The output voltage is constant.

  This condition implies that the ripple in the output voltage is negligibly small, which is ensured by a suitable choice of the capacitance (see chapter 5).
- Within each switching period, the input voltage to the boost converter v_d remains the same. At maximum, the change is

  $\Delta v_d = \left.\frac{dv_d}{dt}\right|_{max} \cdot T_{sw} = \sqrt{2} V_{in} \omega T_{sw} = \sqrt{2} \cdot 120\ \text{V} \cdot 2\pi \cdot 60\ \text{Hz} \cdot \frac{1}{20\ \text{kHz}} = 3.2\ \text{V}.$
- All circuit elements except the inductor are assumed ideal.

  The on resistance of the MOSFET is much smaller than the inductor ESR—to which it is connected in series when the transistor is turned on—and has little effect on the derived converter transfer functions in the frequency ranges that are relevant for the design of the controllers. As for the diode voltage drops, they only amount to a very small fraction of the input and output voltages and are neglected on these grounds.
- The snubber circuit for the transistor as well as auxiliary circuit elements like voltage dividers for the analog to digital conversion (see chapter 5) are disregarded.

As the converter is designed to operate in continuous conduction mode, either the transistor or the diode carries the inductor current and there are two possible switch configurations or states:

1) the transistor Q is on and the diode D is blocking,
2) the transistor Q is off and the diode D conducts.

While the converter is a nonlinear circuit because of the switching action, each state can be represented by a linear circuit. The effective circuit diagrams of the two states are shown in Figure 3.5:

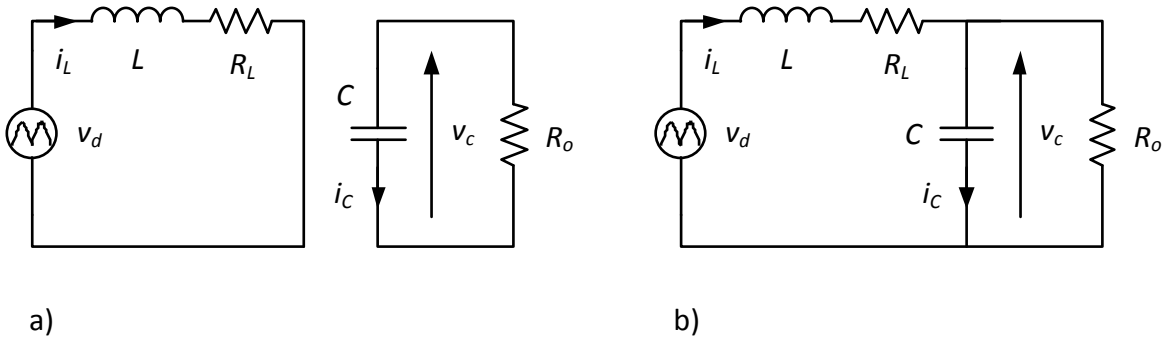


Figure 3.5: States of the converter: a) transistor on; b) transistor off

The states chosen for the mathematical description of the circuit are the current through the inductor and the voltage across the output capacitor. When Q is on, the describing equations as found with Kirchhoff's voltage law are

$$\begin{aligned} v_d &= R_L i_L + L\frac{di_L}{dt} \\ i_C &= C\frac{dv_C}{dt} = \frac{-v_C}{R_o} \end{aligned} \tag{3.21}$$

They can be written in state-space form as

$$\begin{aligned} \frac{di_L}{dt} &= \frac{v_d}{L} - R_L i_L \\ \frac{dv_C}{dt} &= -\frac{v_C}{R_o C} \end{aligned} \tag{3.22}$$

When Q is off and D conducts, then

$$\begin{aligned} R_L i_L + L\frac{di_L}{dt} &= v_d - v_C \\ C\frac{dv_C}{dt} = i_C &= i_L - \frac{v_C}{R_o} \end{aligned} \tag{3.23}$$

or

$$\frac{di_L}{dt} = \frac{v_d}{L} - \frac{v_C}{L} - \frac{R_L}{L} i_L$$
$$\frac{dv_C}{dt} = \frac{i_L}{C} - \frac{v_C}{R_o C} \tag{3.24}$$

Using matrix notation, the state equations from eq. (3.22) can be written in the standard state-space form

$$\frac{d\underline{x}}{dt} = \underline{A}_1 \underline{x} + \underline{b}_1 u$$
$$y = \underline{c}_1 \underline{x} \tag{3.25}$$

where $\underline{x}$ is the state vector $\begin{bmatrix} i_L \\ v_c \end{bmatrix}$, u is the input to the converter, that is v_d, and y is the output of the converter, which can be any linear combination of the two states. Note that $v_o = v_C$ as the capacitor ESR is neglected. The system matrix $\underline{A}_1$ and the input vector $\underline{b}_1$ are

$$\underline{A}_1 = \begin{bmatrix} -\frac{R_L}{L} & 0 \\ 0 & -\frac{1}{R_o C} \end{bmatrix}$$
$$\underline{b}_1 = \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \tag{3.26}$$

Doing likewise for the off state with eq. (3.24), one obtains

$$\underline{A}_2 = \begin{bmatrix} -\frac{R_L}{L} & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{R_o C} \end{bmatrix}$$
$$\underline{b}_2 = \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \tag{3.27}$$

To obtain the plant transfer functions required for the designs of the inductor current and output voltage controllers, $\bar{i}_L$ and v_o, the switching cycle averages as defined by eq. (2.11), are taken as the outputs of the converter respectively. The

overbar notation is omitted with v_o on the grounds that it hardly varies within a switching cycle.

$$\underline{c_1} = \underline{c_2} = \begin{cases} [1 \quad 0] & \text{to obtain } \dfrac{\tilde{\tilde{i}}_L(s)}{\tilde{d}(s)} \\ [0 \quad 1] & \text{to obtain } \dfrac{\tilde{v}_o(s)}{\tilde{d}(s)} \end{cases} \tag{3.28}$$

The general form $\underline{c}_{[i]}$ is used subsequently to refer to either the inductor current or output voltage output vector; $\underline{c}_{[i]}$ has only one non-zero element at the index i, which is 1.

The circuit in Figure 3.5a and eq. (3.26) are valid for the period dT_{sw}, while the circuit in Figure 3.5b and eq. (3.27) hold during $(1-d)T_{sw}$. Calculating the weighted average is done as follows:

$$\underline{A} = \frac{1}{T_{sw}} \left[\underline{A}_1 \cdot dT_{sw} + \underline{A}_2 \cdot (1-d)T_{sw} \right] \tag{3.29}$$

Doing the same operation on $\underline{b}$ and $\underline{c}$ results in

$$\begin{aligned} \underline{A} &= \underline{A}_1 d + \underline{A}_2 (1-d) \\ \underline{b} &= \underline{b}_1 d + \underline{b}_2 (1-d) \\ \underline{c} &= \underline{c}_1 d + \underline{c}_2 (1-d) \end{aligned} \tag{3.30}$$

Inserting the matrices from eqs. (3.26) and (3.27) into (3.30) gives

$$\begin{aligned} \underline{A} &= \begin{bmatrix} -\dfrac{R_L}{L} & -\dfrac{1-d}{L} \\ \dfrac{1-d}{C} & -\dfrac{1}{R_o C} \end{bmatrix} \\ \underline{b} &= \begin{bmatrix} \dfrac{1}{L} \\ 0 \end{bmatrix} \\ \underline{c} &= \begin{cases} [1 \quad 0] & \text{to obtain } \dfrac{\tilde{\tilde{i}}_L(s)}{\tilde{d}(s)} \\ [0 \quad 1] & \text{to obtain } \dfrac{\tilde{v}_o(s)}{\tilde{d}(s)} \end{cases} \end{aligned} \tag{3.31}$$

The obtained system description is one of a linear continuous system—the converter averaged over one switching cycle. One now represents the time-varying d in eq. (3.31) as the sum of one component D, which remains constant from switching cycle to switching cycle, and small perturbations $\tilde{d}$ superimposed on D, writing $d = D + \tilde{d}$. The perturbations in d will cause corresponding perturbations in the state vector such that $\underline{x} = \underline{X} + \underline{\tilde{x}}$, and allow to find the transfer functions necessary for the controller design. In general, the transfer function between any state variable $\overline{x}_{[i]}$ and the duty cycle d can be calculated using

$$G(s) = \frac{\tilde{\overline{x}}_{[i]}(s)}{\tilde{d}(s)} = \underline{c}_{[i]}\left(s\underline{I} - \underline{A}\right)^{-1}\left[\left(\underline{A}_1 - \underline{A}_2\right)\underline{X} + \left(\underline{b}_1 - \underline{b}_2\right)u\right] + \left(\underline{c}_1 - \underline{c}_2\right)\underline{X} \tag{3.32}$$

[59]. Recognizing that $\underline{b}_1 = \underline{b}_2$ and $\underline{c}_1 = \underline{c}_2$ from eqs. (3.26), (3.27) and (3.28) simplifies the general expression for the transfer function eq. (3.32) to

$$G(s) = \underline{c}_{[i]}\left(s\underline{I} - \underline{A}\right)^{-1} \cdot \left(\underline{A}_1 - \underline{A}_2\right)\underline{X} \tag{3.33}$$

The vector $\underline{X}$ is the DC value of the state vector and given by $\underline{X} = -\underline{A}^{-1}\underline{b}\, v_d$. Finding first the negative of the inverse of the system matrix $-\underline{A}^{-1}$ as

$$-\underline{A}^{-1} = \frac{1}{\frac{R_L}{R_o LC} + \frac{(1-D)^2}{LC}} \begin{bmatrix} \frac{1}{R_o C} & -\frac{1-D}{L} \\ \frac{1-D}{C} & \frac{R_L}{L} \end{bmatrix} \tag{3.34}$$

yields

$$\underline{X} = -\underline{A}^{-1}\underline{b}\, v_d = \frac{v_d}{\frac{R_L}{R_o LC} + \frac{(1-D)^2}{LC}} \begin{bmatrix} \frac{1}{R_o LC} \\ \frac{1-D}{LC} \end{bmatrix} = \frac{v_d}{\frac{R_L}{R_o} + (1-D)^2} \begin{bmatrix} \frac{1}{R_o} \\ 1-D \end{bmatrix} \tag{3.35}$$

The term $\left(s\underline{I} - \underline{A}\right)^{-1}$ equals

$$\left(s\underline{I} - \underline{A}\right)^{-1} = \frac{1}{\left(s + \frac{R_L}{L}\right)\left(s + \frac{1}{R_o C}\right) + \frac{(1-D)^2}{LC}} \begin{bmatrix} s + \frac{1}{R_o C} & -\frac{1-D}{L} \\ \frac{1-D}{C} & s + \frac{R_L}{L} \end{bmatrix} \tag{3.36}$$

3.7 Design of the inductor current controller

The control of the inductor current is realized as constant-frequency average current mode control. The switching frequency is kept constant at 20 kHz and the duty cycle determines the transistor on time. The controlled variable is the inductor current averaged over one switching cycle. The control loop configuration with the relevant signals can be seen from Figure 3.6:

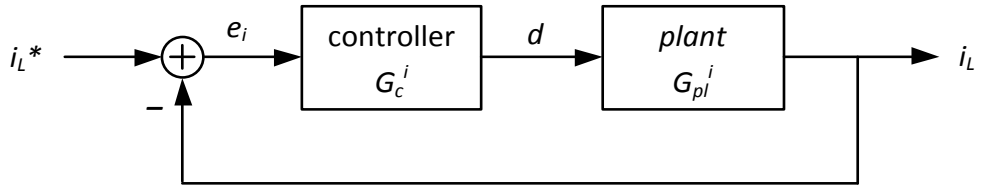


Figure 3.6: Block diagram of the inductor current control loop

3.7.1 Derivation of the plant transfer function

To find $\frac{\tilde{\hat{i}}_L(s)}{\tilde{d}(s)}$, $\underline{c}_{[1]} = [1 \quad 0]$ is plugged into the general form of the transfer function eq. (3.33) together with $\underline{X}$ from eq. (3.35) and $\left(s\underline{I} - \underline{A}\right)^{-1}$ from eq. (3.36). This produces

$$G_{pl}^i(s) = \frac{\tilde{\hat{i}}_L(s)}{\tilde{d}(s)} = \frac{v_d}{\frac{R_L}{R_o} + (1-D)^2} \frac{1-D}{L} \frac{s + \frac{2}{R_o C}}{\left(s + \frac{R_L}{L}\right)\left(s + \frac{1}{R_o C}\right) + \frac{(1-D)^2}{LC}} \tag{3.37}$$

This transfer function is dependent on the instantaneous values of v_d and D, which change throughout the line half-cycle. To obtain a linear model, the averages of both quantities over one line half-cycle are used in eq. (3.37). For v_d, this means $V_{d,av} = \frac{2\sqrt{2}}{\pi} V_{in}$. The average of $(1-D)$ over one line half-cycle is determined from Kirchhoff's voltage law in conjunction with the expression for the averaged switch voltage from eq. (2.12):

$$v_d = R_L \bar{i}_L + L\frac{d\bar{i}_L}{dt} + V_o(1-D) \tag{3.38}$$

Solved for $(1-D)$ and with $\bar{i}_L$ substituted according to eq. (2.4), eq. (3.38) reads

$$\begin{aligned} 1-D &= \frac{1}{V_o}\left[v_d - R_L\bar{i}_L - L\frac{d\bar{i}_L}{dt}\right] \\ &= \frac{1}{V_o}\left[v_d - \kappa\left(R_L v_d + L\frac{dv_d}{dt}\right)\right] \end{aligned} \tag{3.39}$$

The rectified voltage takes the form $v_d = \sqrt{2}V_{in}|\sin\theta|$, and in the interval $\theta \in [0, 180°)$, eq. (3.39) then becomes

$$1-D = \frac{\sqrt{2}V_{in}}{V_o}\left[\sin\theta - \kappa\left(R_L\sin\theta + \omega L\cos\theta\right)\right] \tag{3.40}$$

Now $(1-D)$ is averaged over one line half-cycle:

$$\begin{aligned} (1-D)_{av} &= \frac{1}{\pi}\int_0^{\pi}(1-D)\,d\theta = \frac{1}{\pi}\frac{\sqrt{2}V_{in}}{V_o}\left[2 - \kappa_{av}\cdot 2R_L\right] \\ &= \frac{2\sqrt{2}}{\pi}\frac{V_{in}}{V_o}\left[1 - R_L\kappa_{av}\right] \end{aligned} \tag{3.41}$$

The average of κ over one line half-cycle is calculated from the power balance

$$P_{in} = P_o + P_R \tag{3.42}$$

in which one can substitute $P_{in} = V_{in}I_{in} = V_{in}^2\,\kappa_{av}$ and $P_R = I_{in}^2R_L = V_{in}^2\kappa_{av}^2R_L$, thereby obtaining

$$V_{in}^2\,\kappa_{av} = P_o + V_{in}^2\kappa_{av}^2R_L \tag{3.43}$$

This is a quadratic equation in κ_{av} whose feasible solution is

$$\begin{aligned} \kappa_{av} &= \frac{1}{2R_L} - \frac{\sqrt{V_{in}^2 - 4P_oR_L}}{2V_{in}R_L} \\ &= \frac{1}{2\cdot 0.6\ \Omega} - \frac{\sqrt{(120\ \mathrm{V})^2 - 4\cdot 200\ \mathrm{W}\cdot 0.6\ \Omega}}{2\cdot 120\ \mathrm{V}\cdot 0.6\ \Omega} = 0.0140\ \frac{\mathrm{A}}{\mathrm{V}} \end{aligned} \tag{3.44}$$

This result can be plugged into eq. (3.41) to obtain $(1-D)_{av} = 0.282$ and then used in the plant transfer function eq. (3.37) together with the nominal circuit element parameters from section 5.1. The result is

$$
\begin{aligned}
G_{pl}^{i}(s) &= \frac{\tilde{\tilde{\imath}}_L(s)}{\tilde{d}(s)} \\
&= \frac{\left(\frac{2\sqrt{2}}{\pi}\cdot 120\right)\cdot 0.282}{\left(\frac{0.6}{722}+0.282^2\right)\cdot(8\cdot 10^{-3})} \\
&\quad\cdot \frac{s+\frac{2}{722\cdot(270\cdot 10^{-6})}}{\left(s+\frac{0.6}{8\cdot 10^{-3}}\right)\left(s+\frac{1}{722\cdot(270\cdot 10^{-6})}\right)+\frac{0.282^2}{(8\cdot 10^{-3})\cdot(270\cdot 10^{-6})}} \\
&= 47{,}400\,\frac{s+10.3}{(s+75.0)(s+5.13)+36{,}800}
\end{aligned}
\tag{3.45}
$$

A Bode plot of the plant transfer function can be found in Figure 3.8.

3.7.2 Controller design

A straightforward way to design a controller with a good performance is the k factor approach [60]. The k factor approach is a technique that allows obtaining controller parameters using only a few algebraic equations. Depending on the nature of the plant transfer function, one chooses one of three types of controllers. While type 1 is a pure integral controller with only the gain as a parameter, the type 2 and type 3 controllers are compensated ones with three parameters each.

The design of the controller is carried out in the frequency domain. Specifications to be chosen are the open-loop gain crossover frequency ω_c and phase margin φ_m. In general, it is desirable to have a high gain crossover frequency in order to get a fast response. On the other hand, there are inherent sources of disturbances in each control loop. In the current control loop, this is the ripple in the inductor current. The critical frequency here is the switching frequency. It is desirable that the magnitude of the open-loop transfer function at this frequency is as low as possible

to prevent the disturbances from propagating around the control loop and distorting the duty cycle. It must certainly be smaller than one. As for the phase margin, [60] recommends 60° as a good trade-off between fast transient response, i.e. short settling time, and stability (low overshoot, highly damped oscillations). In [1], the range of 45 to 60° is stated as a reasonable choice. The steps to design the controller with the k factor approach are as follows:

First, the desired gain crossover frequency ω_c has to be selected that is lower than the switching frequency. A value of $\omega_c = 20{,}000\ \mathrm{s}^{-1}$ that makes $f_c = \frac{f_{sw}}{2\pi}$ will be suitable [11].

Second, the desired phase margin φ_m is chosen to be 60°. The phase of the plant at the gain crossover frequency is $\varphi_{pl}(\omega_c) = \arg\{G_{pl}^{i}(j\omega_c)\} = \arg\left\{\frac{V_o}{j\omega_c \cdot L}\right\} = -90°$. A pure integral controller (type 1) would add another $-90°$ to the phase of the open-loop transfer function. For the control loop to have a margin of 60° away from the critical value of $-180°$, the compensator needs to provide a phase boost of $\varphi_b = 60°$. This can be accomplished by a type 2 controller. Its general form is

$$G_c(s) = K\ \frac{s + \omega_z}{s\left(s + \omega_p\right)} \tag{3.46}$$

Next, the factor k can be calculated as follows: $k = \tan\left(45° + \frac{\varphi_b}{2}\right) = \tan\left(45° + \frac{60°}{2}\right) = 3.73$. The factor k fixes the ratio of the compensator pole and zero via $\frac{\omega_p}{\omega_z} = k^2$. Furthermore, ω_z and ω_p are located such that their geometric mean is the gain crossover frequency, $\sqrt{\omega_z\ \omega_p} = \omega_c$. The resulting phase response of a type 2 controller is plotted in Figure 3.7:

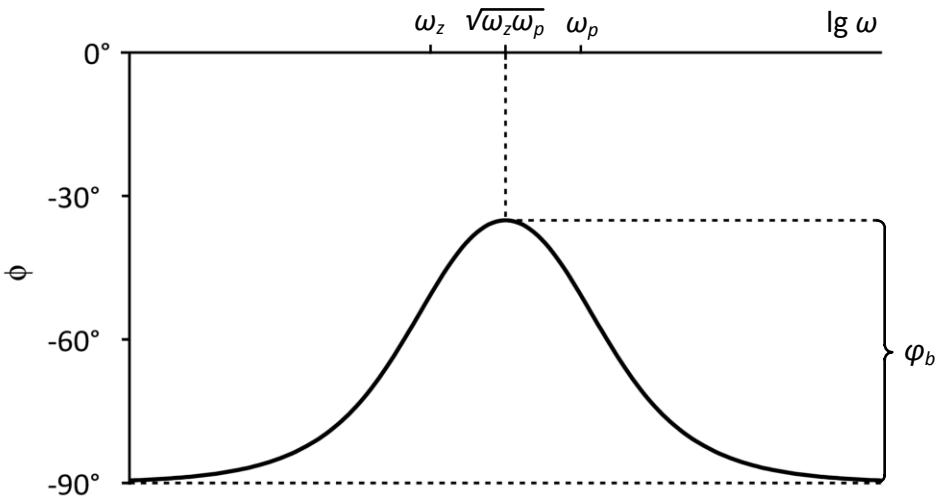


Figure 3.7: Phase response of a type 2 controller

Therefore,

$$\omega_z = \frac{\omega_c}{k} = \frac{20{,}000\ \text{s}^{-1}}{3.73} = 5{,}360\ \text{s}^{-1} \tag{3.47}$$

and

$$\omega_p = k \cdot \omega_c = 3.73 \cdot 20{,}000\ \text{s}^{-1} = 74{,}600\ \text{s}^{-1} \tag{3.48}$$

Finally, the controller gain K is obtained from the definition of the gain crossover frequency $|G_{OL}(j\omega_c)| = \left|G_{pl}^{i}(j\omega_c) \cdot G_c^{i}(j\omega_c)\right| = 1$:

$$K = \frac{1}{\left| G_{pl}^{i}(j\omega_c)\ \dfrac{j\omega_c + \omega_z}{j\omega_c\left(j\omega_c + \omega_p\right)} \right|} = 31{,}500 \tag{3.49}$$

Figure 3.8 shows the Bode plots of the plant, controller and open-loop transfer functions of the current control loop:

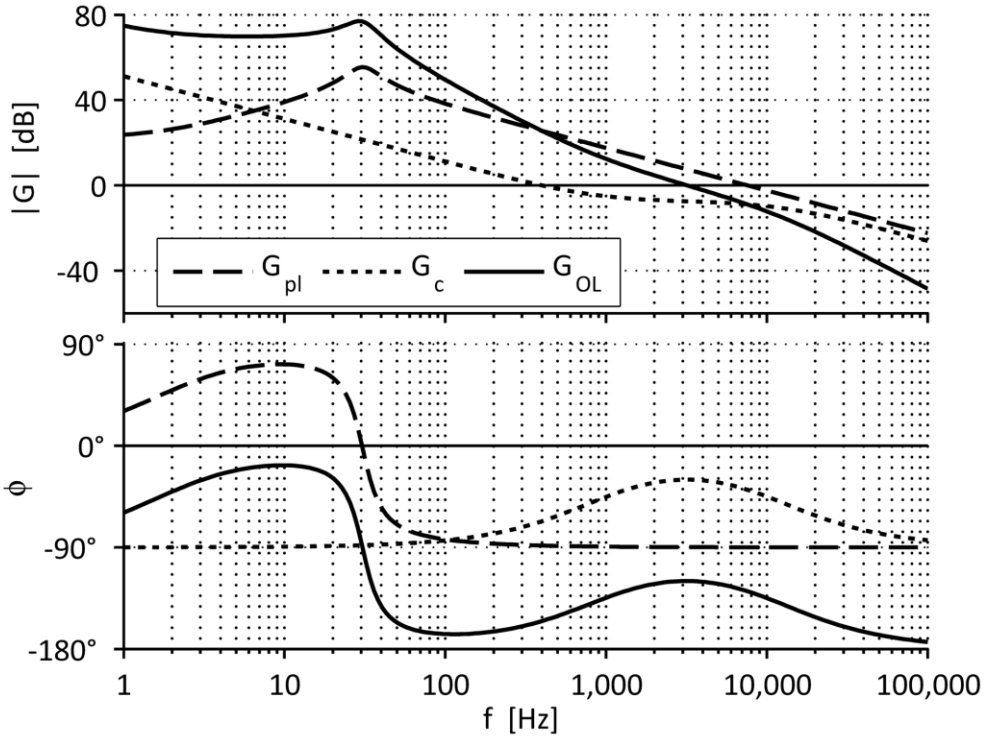


Figure 3.8: Bode plots of the current control loop

To confirm sufficient attenuation of the open-loop transfer function at the switching frequency, its magnitude is calculated: $|G_{OL}(j\omega_{sw})| = 0.081 = -22$ dB. The quality of the current waveshape can be seen from Figure 4.2.

3.8 Design of the output voltage controller

Figure 3.9 schematically depicts the interactions to control the converter output voltage:

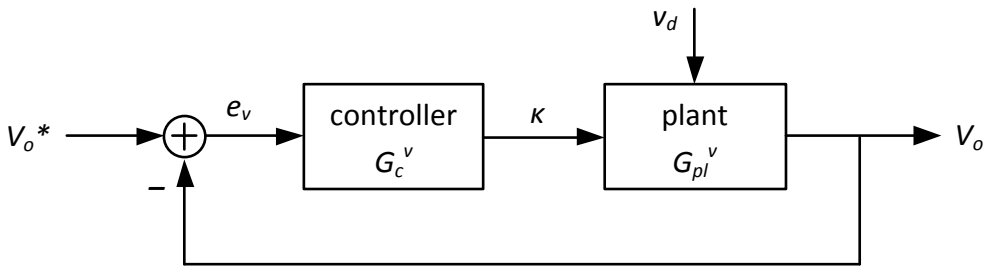


Figure 3.9: Block diagram of the output voltage control loop

3.8.1 Derivation of the plant transfer function

The state-space model of the converter in section 3.6 also serves to derive the required relation between $\tilde{v}_o$ and $\tilde{\kappa}$ via

$$\frac{\tilde{v}_o(s)}{\tilde{\kappa}(s)} = \frac{\tilde{v}_o(s)}{\tilde{d}(s)} \frac{\tilde{d}(s)}{\tilde{\bar{i}}_L(s)} \frac{\tilde{\bar{i}}_L(s)}{\tilde{\kappa}(s)} \tag{3.50}$$

To find $\frac{\tilde{v}_o(s)}{\tilde{d}(s)}$, $\underline{c}_{[2]} = [0 \quad 1]$ can be inserted into the general form of the converter transfer function (3.33):

$$\frac{\tilde{v}_o(s)}{\tilde{d}(s)} = -\frac{v_d}{R_o C} \frac{1}{\frac{R_L}{R_o} + (1-D)^2} \frac{s + \left[\frac{R_L}{L} - \frac{R_o}{L}(1-D)^2\right]}{\left(s + \frac{R_L}{L}\right)\left(s + \frac{1}{R_o C}\right)} \tag{3.51}$$

Then, in conjunction with the previously found $\frac{\tilde{\bar{i}}_L(s)}{\tilde{d}(s)}$ from (3.37),

$$\begin{aligned} \frac{\tilde{v}_o(s)}{\tilde{\bar{i}}_L(s)} &= \frac{\tilde{v}_o(s)}{\tilde{d}(s)} \frac{\tilde{d}(s)}{\tilde{\bar{i}}_L(s)} \\ &= -\frac{L}{R_o C(1-D)} \frac{s + \frac{1}{L}[R_L - R_o(1-D)^2]}{s + \frac{2}{R_o C}} \\ &= \frac{L}{R_o C(1-D)} \frac{-s + \frac{1}{L}[R_o(1-D)^2 - R_L]}{s + \frac{2}{R_o C}} \end{aligned} \tag{3.52}$$

The line half-cycle average of $(1-D)$ was calculated in subsection 3.7.1.

Because the current control loop has a much faster transient behaviour than the voltage control loop, as evident from the higher control loop bandwidth, one can assume that $\bar{i}_L = i_L^*$ in this context. Therefore, the ratio $\frac{\tilde{\bar{i}}_L(s)}{\tilde{\kappa}(s)}$ is simply v_d (eq. (2.4)). Eventually, the desired plant transfer function $G_{pl}^v(s) = \frac{\tilde{v}_o(s)}{\tilde{\kappa}(s)}$ can then be found from eq. (3.52), $\frac{\tilde{\bar{i}}_L(s)}{\tilde{\kappa}(s)} \approx V_{d,av}$ and the nominal circuit element parameters from section 5.1 as

$$
\begin{aligned}
G_{pl}^v(s) &= \frac{\tilde{v}_o(s)}{\tilde{\kappa}(s)} = \frac{\tilde{\bar{i}}_L(s)}{\tilde{\kappa}(s)} \cdot \frac{\tilde{v}_o(s)}{\tilde{\bar{i}}_L(s)} \\
&= \frac{2\sqrt{2}}{\pi} V_{in} \frac{L}{R_o C (1-D)_{av}} \frac{-s + \frac{1}{L}\left[R_o (1-D)_{av}{}^2 - R_L\right]}{s + \frac{2}{R_o C}} \\
&= \frac{\left(\frac{2\sqrt{2}}{\pi} \cdot 120\right) \cdot (8 \cdot 10^{-3})}{722 \cdot (270 \cdot 10^{-6}) \cdot 0.282} \frac{-s + \frac{722 \cdot 0.282^2 - 0.6}{8 \cdot 10^{-3}}}{s + \frac{2}{722 \cdot (270 \cdot 10^{-6})}} \\
&= 15.7 \frac{-s + 7100}{s + 10.3}
\end{aligned}
\tag{3.53}
$$

Refer to Figure 3.11 for a Bode plot of the plant transfer function.

3.8.2 Controller design

Like for the design of the current controller, the k factor approach is used to find a suitable voltage controller. The steps are analogous to the ones taken in section 3.7.2.

After passing through the full-bridge rectifier, the input current will be a pulsating direct current with a fundamental frequency of twice the line frequency. This results in an output voltage ripple of the same frequency. To ensure a substantial attenuation of this frequency in the calculation of the reference input current, the gain crossover frequency of the output voltage control loop must be significantly lower. The critical frequency for the open-loop transfer function of the voltage

control loop, at which it must be minimized, is 120 Hz. The gain crossover frequency should not be too small, on the other hand, to guarantee a reasonably fast transient response. For the purpose of evaluating the effect of the control loop bandwidth on the distortion of the input current in steady-state and on the transient response of the output voltage to load changes, two designs with gain crossover frequencies of 10 Hz and 20 Hz are worked out. The performance characteristics are then compared for both designs, and the decision for one is made as a trade-off between their effects on the converter. The phase margin is chosen to be 60°, which is achieved with a type 2 controller (eq. (3.46)) in both cases. The rest of the steps differs for the two gain crossover frequencies and is detailed in Table 3.2:

Table 3.2: Voltage controller design

$f_c = 10$ Hz	$\varphi_{pl}(\omega_c) = \arg\{G_{pl}^v(j\omega_c)\} = -81.2°$
	$\varphi_b = \varphi_m - (90° + \varphi_{pl}) = 60° - (90° - 81.2°) = 51.2°$
	$k = \tan\left(45° + \frac{\varphi_b}{2}\right) = \tan\left(45° + \frac{51.2°}{2}\right) = 2.84$
	$\omega_z = \frac{\omega_c}{k} = \frac{2\pi \cdot 10\text{ Hz}}{2.84} = 22.1\text{ s}^{-1}$
	$\omega_p = k \cdot \omega_c = 2.84 \cdot 2\pi \cdot 10\text{ Hz} = 179\text{ s}^{-1}$
	$K = \frac{1}{\left\lvert G_{pl}^v(j\omega_c)\ \frac{j\omega_c + \omega_z}{j\omega_c(j\omega_c + \omega_p)} \right\rvert} = 0.102$
	$\lvert G_{OL}(j \cdot 2\pi \cdot 120\text{ Hz}) \rvert = 0.020 = -34\text{ dB}$

$f_c = 20$ Hz

$$\varphi_{pl}(\omega_c) = \arg\{G_{pl}^{v}(j\omega_c)\} = -86.3°$$

$$\varphi_b = \varphi_m - (90° + \varphi_{pl}) = 60° - (90° - 86.3°) = 56.3°$$

$$k = \tan\left(45° + \frac{\varphi_b}{2}\right) = \tan\left(45° + \frac{56.3°}{2}\right) = 3.31$$

$$\omega_z = \frac{\omega_c}{k} = \frac{2\pi \cdot 20\ \text{Hz}}{3.31} = 38.0\ \text{s}^{-1}$$

$$\omega_p = k \cdot \omega_c = 3.31 \cdot 2\pi \cdot 20\ \text{Hz} = 416\ \text{s}^{-1}$$

$$K = \frac{1}{\left| G_{pl}^{v}(j\omega_c)\ \frac{j\omega_c + \omega_z}{j\omega_c(j\omega_c + \omega_p)} \right|} = 0.469$$

$$|G_{OL}(j \cdot 2\pi \cdot 120\ \text{Hz})| = 0.081 = -22\ \text{dB}$$

The comparison in Table 3.3 shows that with a gain crossover frequency of 20 Hz, the signal κ contains a considerable 120 Hz ripple (peak-to-peak value κ_{pp}), whereas the ripple is noticeably more attenuated when $f_c = 10$ Hz. It distorts the reference inductor current waveform, and in consequence the input current. This effect is evident from the magnitude of the third harmonic of the input current $I_{in(3)}$ and its total harmonic distortion (THD) and is shown in Figure 3.10. The effect is even stronger at half load.

By contrast, the transient response of the output voltage following a change in load suffers from the smaller bandwidth. The reaction of the controller to a deviation of the output voltage from its command is slower. For a step change in load from half load to full load, the undershoot Δv_o^- of the output voltage is listed in Table 3.3 as an example, as is the overshoot Δv_o^+ for a step change from full load to half load. Circumstances are similar when the load decreases, in which case the output voltage initially rises.

Table 3.3: Effects of the voltage control loop bandwidth

	$\frac{\kappa_{pp}}{\kappa_{av}}$	$I_{in(3)}$	THD$\{i_{in}\}$	$\frac{\Delta v_o^-}{V_o^*}$	$\frac{\Delta v_o^+}{V_o^*}$
$f_c = 10$ Hz	5%	27 mA	2.3%	3.4%	3.2%
$f_c = 20$ Hz	21%	95 mA	5.8%	2.1%	1.7%

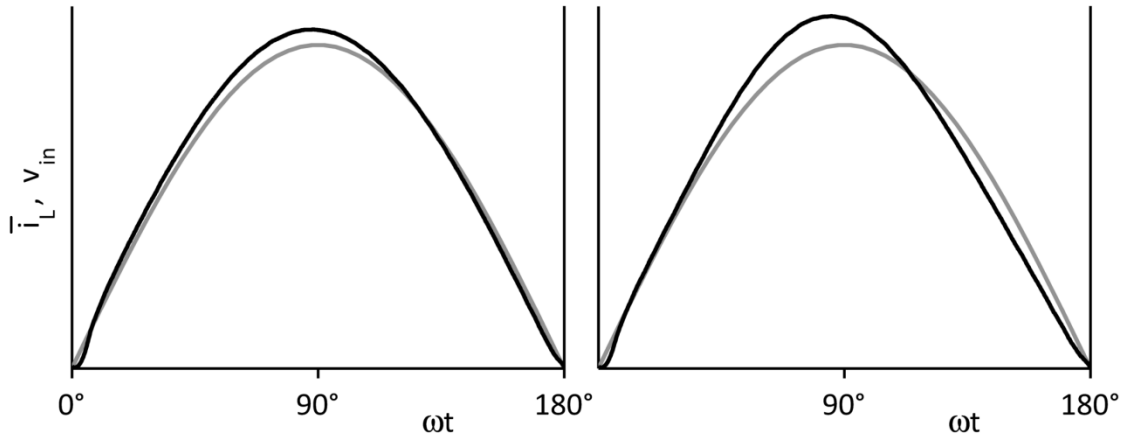


Figure 3.10: Input voltage (grey) and averaged inductor current (black) for a voltage control loop bandwidth of 10 Hz (left) and 20 Hz (right)

The priority is given to the precise shaping of the input current. Hence, the controller that makes the control loop bandwidth 10 Hz is going to be used. Bode plots of the plant, controller and open-loop transfer functions are depicted in Figure 3.11.

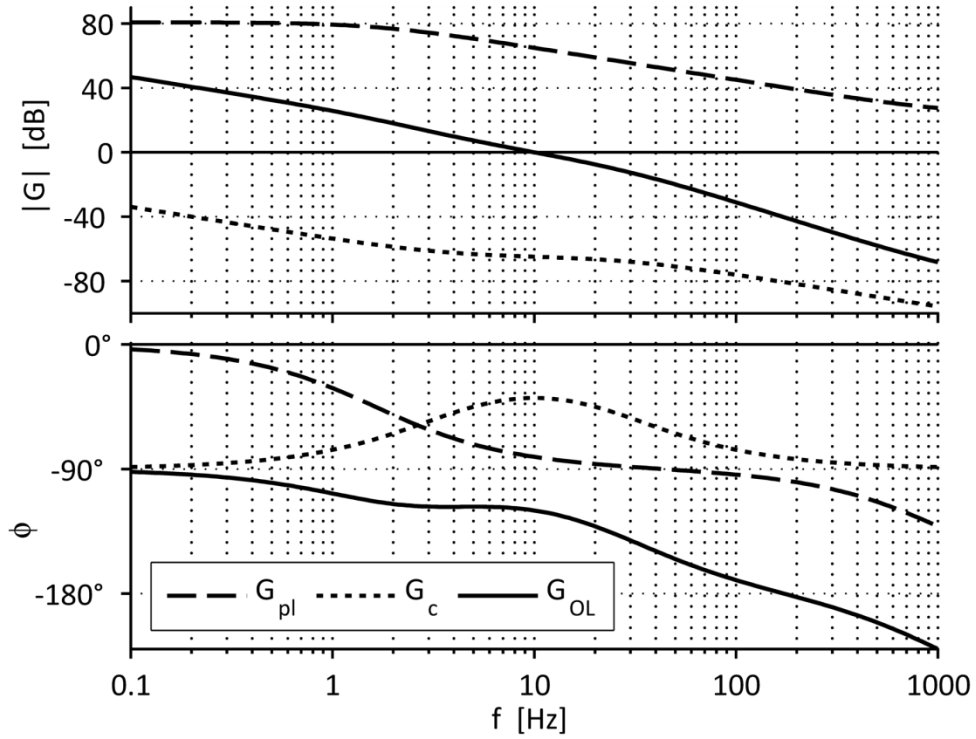


Figure 3.11: Bode plots of the voltage control loop

3.8.3 Additional considerations

Under normal operating conditions, the converter output voltage never deviates very much from its reference value. See the plots in Figure 4.6 and Figure 4.7 for responses to disturbances in the form of abrupt load changes. After turn-on, by contrast, the voltage error is initially very large. Because the voltage controller has integral action, the error is integrated, which leads to an overshoot in the output voltage. As a simple means to prevent controller windup, the voltage error is limited. A suitable bound was found via simulation. Figure 3.12 shows simulation results of the output voltage after turn-on with the voltage error limited to various values. It is assumed that, when the control action commences, the output capacitor is already charged to the peak input voltage of $\sqrt{2} \cdot 120$ V (see section 5.1).

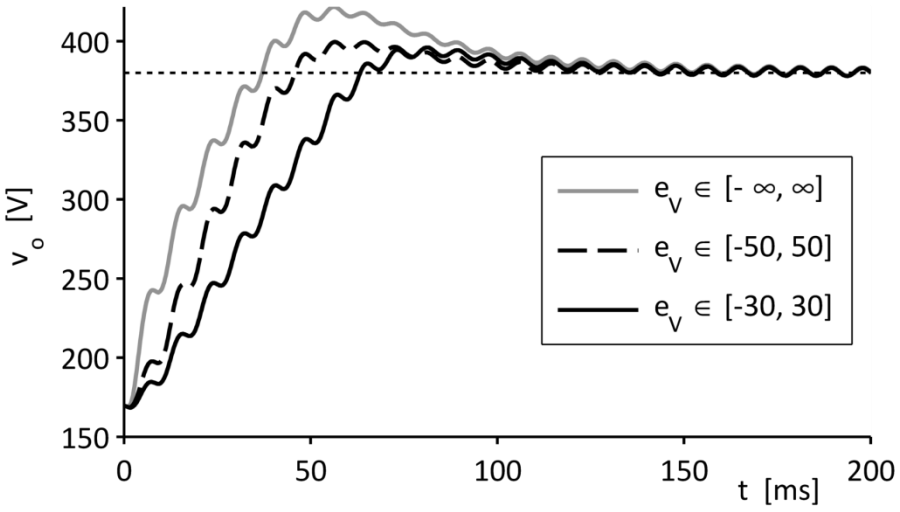


Figure 3.12: Simulated output voltage during turn-on

By limiting the voltage error to ± 30 V, the overshoot in the output voltage is reduced from 42 V to 16 V. The highest occurring input current peak is 6.2 A.

In addition, the controller output κ is limited to the interval $[0.0001, 0.024]$. According to (3.44), for rated conditions $\kappa = 0.014 \frac{\mathrm{A}}{\mathrm{V}}$, while at 10% load $\kappa = 0.0014 \frac{\mathrm{A}}{\mathrm{V}}$. The above stated interval includes generous margins to allow for the voltage control, but protects against too large currents through the upper bound and ensures that κ is strictly positive through the lower one. The latter is necessary for the computed reference current to be a feasible value; the bound may be reached when the load suddenly decreases substantially.

3.9 Overview

Figure 3.13 shows all the units required for the control action of the converter along their interfaces:

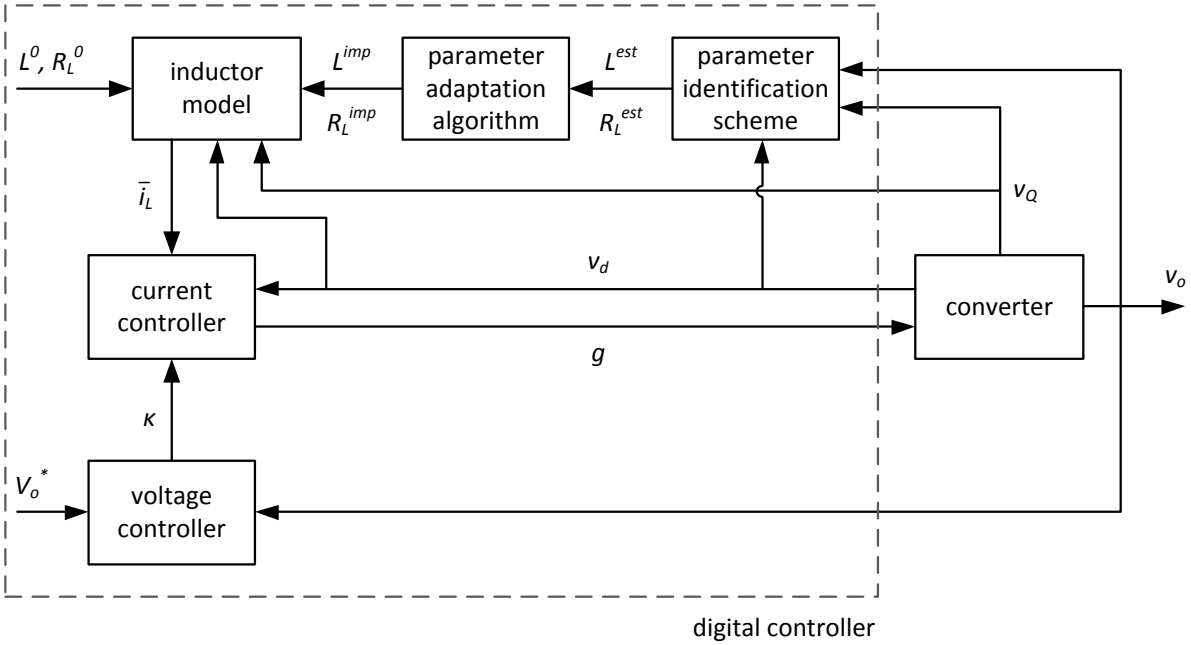


Figure 3.13: Block diagram of the implemented control configuration

4 Simulation of the converter

Simulation played an important role throughout the development process. It was used to investigate the behaviour of the converter, to test the controllers, to devise the inductor model parameter identification scheme and to study the effects of the model parameter adaptation.

Simulations have already been used as a tool for investigation and as a design aid in chapter 3. In this chapter, the main simulation results are presented. First, the converter was simulated without the inductor model adaptation in steady-state operation (section 4.2) and at load transients (section 4.3). Then the inductor model was added and the performance was tested under a variety of conditions in section 4.4.

4.1 Simulation settings

The simulations were carried out with the software Simulink [61]. The blocks for the converter itself (such as the input voltage source, the power semiconductors, the inductor and the load resistor) and some of the control blocks are from Simulink's SimPowerSystems toolbox [62]. For the simulation to run reasonably fast, the algorithm to converge and the results to be sufficiently accurate, it is important that proper simulation settings are used.

Simulating switch-mode circuits involves computing numerical solutions for highly nonlinear and stiff problems. A stiff problem has time constants that differ from one another by many orders of magnitude. In power electronics, such circumstances arise because of the presence of switches. In the state-space description of the model, this leads to system matrices with eigenvalues in a very large range. Solving these is computationally challenging. Implicit solvers are stable enough to deal with such problems. The mathematical description is such that the

unknown appears on both sides of the equation and a set of nonlinear equations needs to be solved at each time step. This approach is computationally expensive, but makes implicit solvers more stable than explicit ones. [63]

The employed solver ode23tb is an implicit variable-step solver for stiff problems from the Runge-Kutta family. It implements the trapezoidal rule (TR) to integrate up to an intermediate point between the current time step t_k and the successive time step t_{k+1}; then it uses a second-order backwards differentiation formula (BDF2) to advance the solution from the intermediate time to the new time step t_{k+1}. TR and BDF2 alike are very stable methods. [63,64]

In order to accurately represent the switching behaviour of the converter, the time steps taken by the solver must be much smaller than the switching period of $T_{sw} = \frac{1}{f_{sw}} = \frac{1}{20\ \text{kHz}} = 50\ \mu\text{s}$. It was found that the following solver parameters are suitable for simulating the converter and a good compromise between reliable and accurate results and a short computation time:

Table 4.1: Solver settings

solver	ode23tb		
maximum step size	0.1 µs	relative tolerance	10^{-4}
minimum step size	0.05 µs	absolute tolerance	automatic
initial step size	automatic	solver reset method	fast
number of consecutive minimum steps	1	solver Jacobian method	automatic

More stringent settings extend the simulation time, but do not produce results of significantly higher accuracy. Setting larger step sizes or tolerances, on the other hand, leads to noticeably different results.

To ensure good convergence and reliable results, it is also crucial that the model is discretized. The sample time was set to the maximum time step of the solver, 0.1 µs. Discretizing the circuit also shortens the simulation time considerably.

4.2 Steady-state operation

To confirm the proper performance of the converter including input current and voltage control, the circuit was first simulated in steady-state operation with a non-adaptive inductor model. See Figure 4.1 for the corresponding Simulink model.

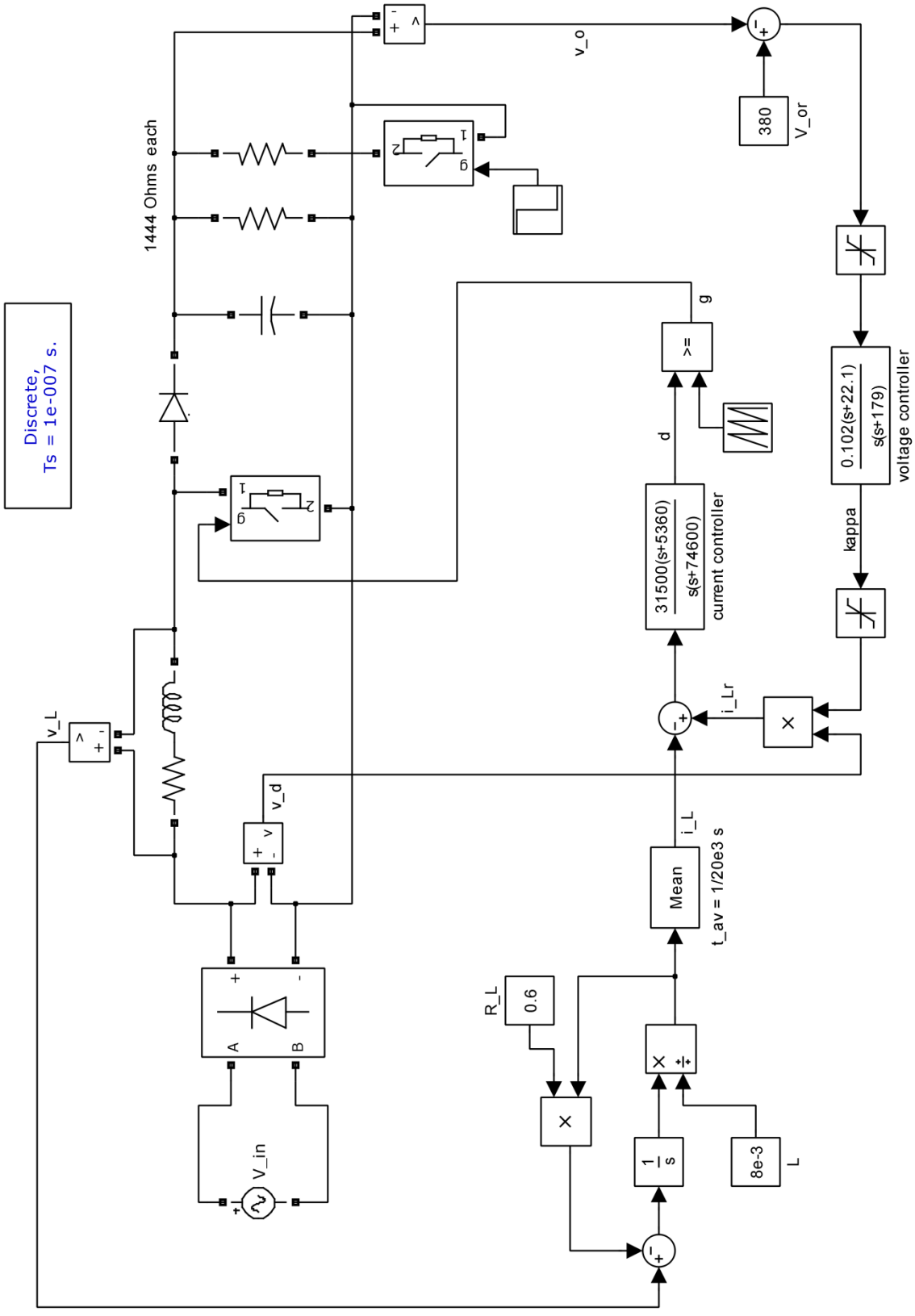


Figure 4.1: Simulation schematic of the circuit without model adaptation

Figure 4.2 shows the simulation results for the input current under rated conditions. As evident from the waveforms, the input current, apart from exhibiting the switching ripple, follows the sinusoidal shape of the input voltage very precisely.

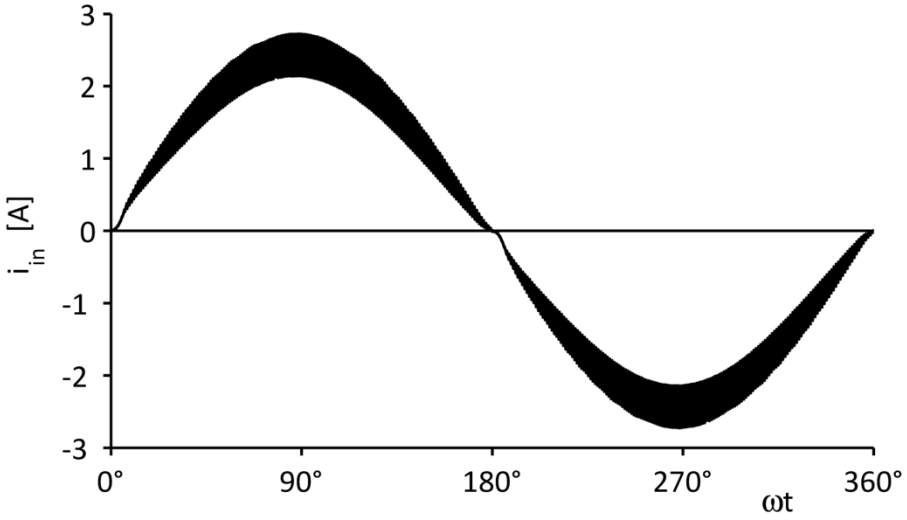


Figure 4.2: Simulated input current waveform at rated load

When the load current has only half its nominal value, the input current will also be smaller. The current ripple, on the other hand, remains the same, as it is determined by the rectified input voltage, output voltage and inductance. This is evident from Figure 4.3.

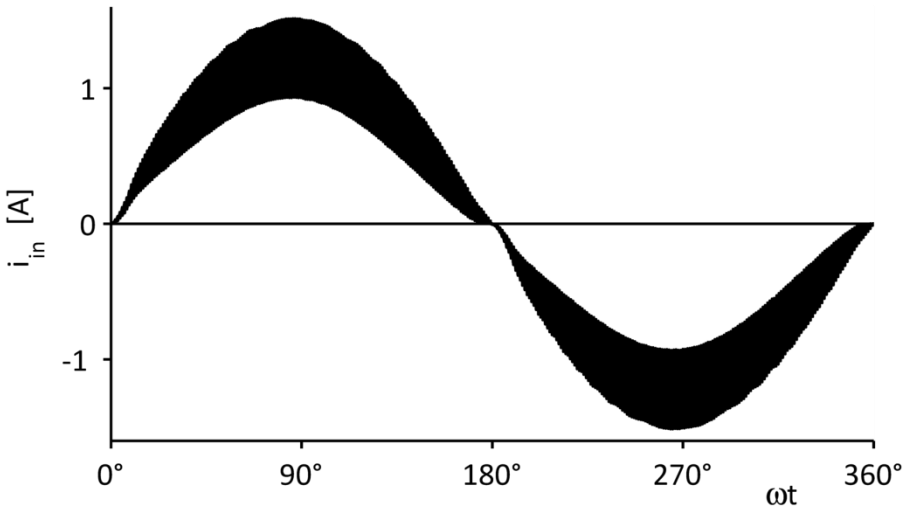


Figure 4.3: Simulated input current waveform at half load

The RMS values of the first harmonics were determined for four different load levels by the FFT Analysis Tool from 1 s of data sampled at intervals of 1 µs. The results are listed in Table 4.2 and visualized in Figure 4.4. Even multiples of 60 Hz do not occur. The calculation of the THD, also given in Table 4.2, takes all frequency components up to 499,920 Hz into account.

Table 4.2: RMS values of the low-order harmonics of the input current in mA
and its total harmonic distortion (simulated)
for different output powers (100%, 75%, 50% and 25% of rated load)

P_o	$I_{in(1)}$	$I_{in(3)}$	$I_{in(5)}$	$I_{in(7)}$	$I_{in(9)}$	$I_{in(11)}$	THD
200 W	1699	27.5	10.7	9.8	8.8	7.9	2.3%
150 W	1274	23.1	10.5	9.4	8.2	7.2	2.7%
100 W	852	19.1	10.6	9.1	7.7	6.2	3.5%
50 W	431	17.1	11.4	8.7	6.2	4.2	6.4%

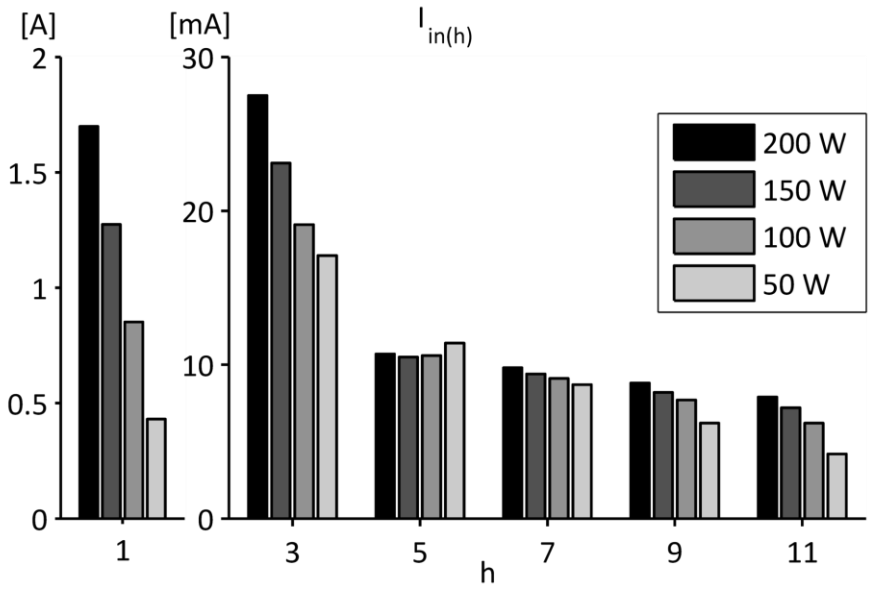


Figure 4.4: RMS values of the low-order harmonics of the input current for different output powers (100%, 75%, 50% and 25% of rated load)

Meanwhile, the output voltage controller manages to regulate the average value of the converter output voltage to its reference value of 380 V, as can be observed from Figure 4.5. The waveform shows a small ripple ($V_{o,pp} = 5.3$ V) as a result of the pulsating input power.

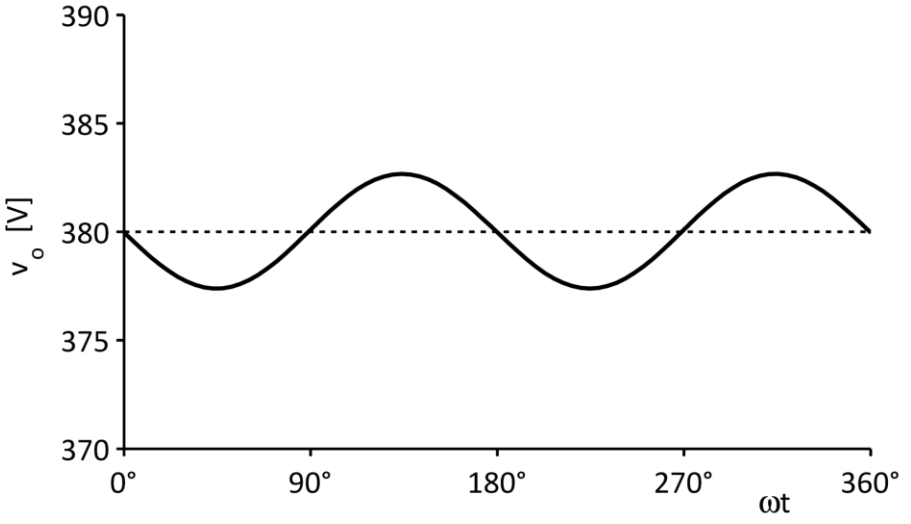


Figure 4.5: Simulated output voltage waveform at rated load

To conclude, the controllers fulfil their tasks effectively when the converter operates in steady state. The effects of disturbances are dealt with in the following section.

4.3 Load change transients

The simulation results in this section illustrate the response of the converter to sudden changes in the output current, which occur when the load changes. When the load resistance changes abruptly from its rated value of 722 Ω to 1444 Ω, the converter responds as shown in Figure 4.6.

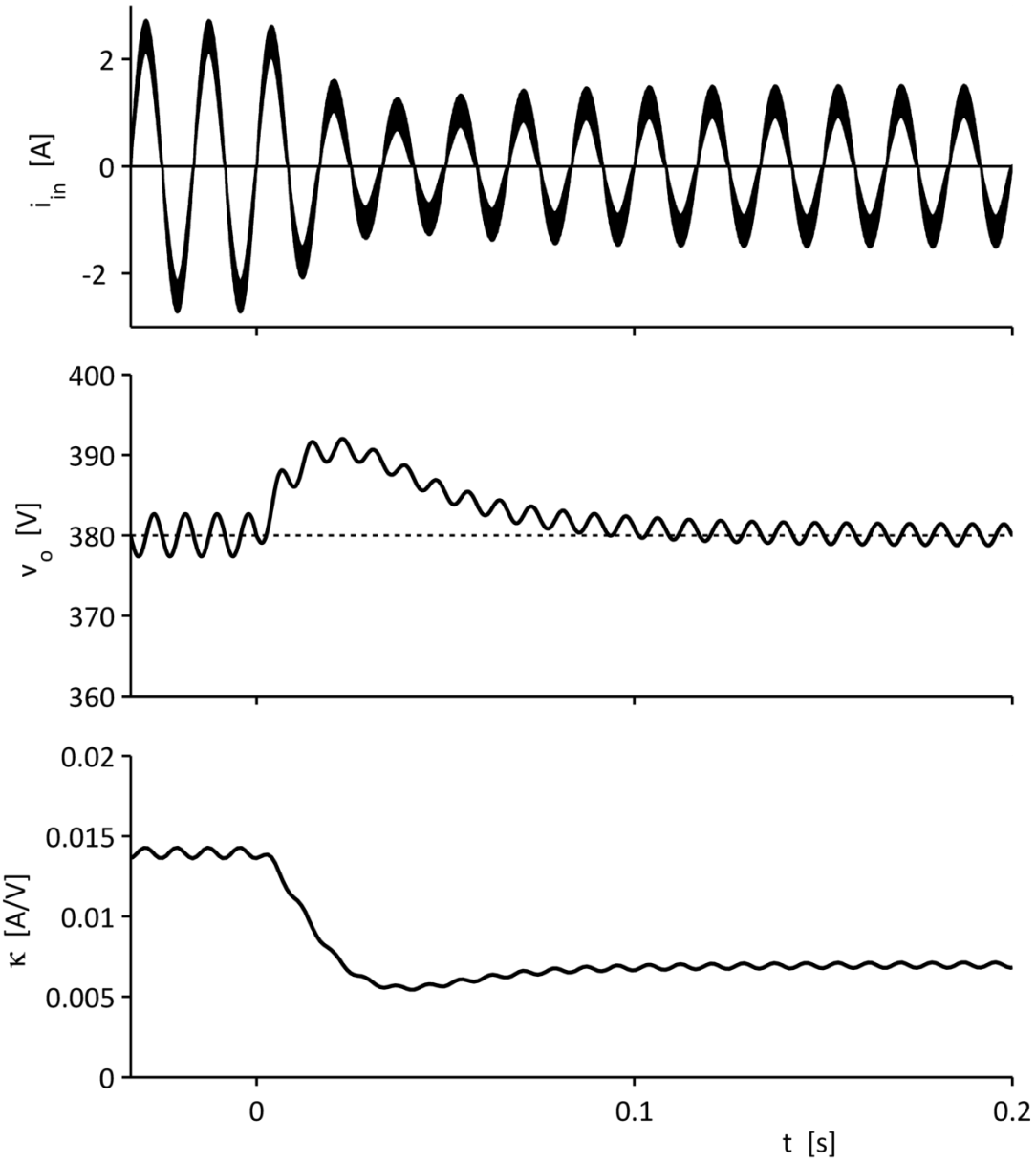


Figure 4.6: Simulated converter waveforms for a load step change from full load to half load at $t = 0$

The converter has reached steady state before the load changes at $t = 0$. As the load current suddenly drops, less power is consumed at the output, causing the output voltage to rise. This increase is noticed by the voltage controller, which in response decreases κ to regulate the output voltage to its reference value. A lower κ in turn leads to a smaller magnitude of the input current.

Circumstances are likewise when a step change occurs in the load from 1444 Ω to 722 Ω and the load current rises from half its rated value to its rated value. The voltage controller increases its output κ, thus demanding more current from the mains to transfer into the load. The waveforms of the most important signals are given in Figure 4.7:

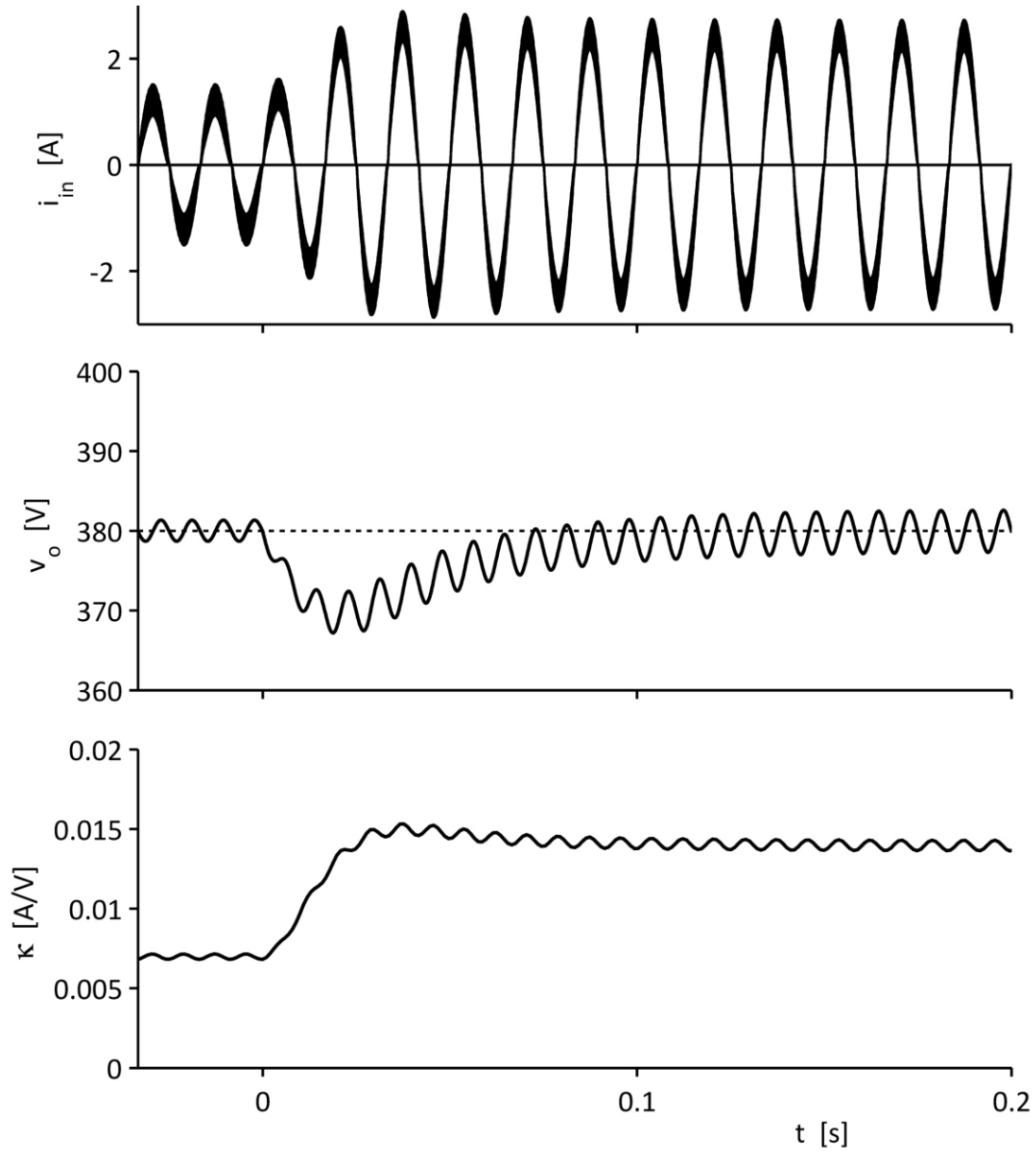


Figure 4.7: Simulated converter waveforms for a load step change from half load to full load at $t = 0$

The behaviour after turn-on has already been investigated in subsection 3.8.3 in connection with the voltage controller design. See Figure 3.12 for the turn-on transient.

4.4 Input voltage change transients

Aside from changes in the load, fluctuations in the input voltage to the converter can present disturbances to the control. The standards in North America allow tolerances of $\pm 5\%$ around the nominal line voltage of 120 V, that is the line voltage can change from 114 to 126 V. In the simulation, the converter has been subjected to abrupt changes in the line voltage RMS value from $1.1 \cdot 120\ \mathrm{V} = 132\ \mathrm{V}$ to $0.9 \cdot 120\ \mathrm{V} = 108\ \mathrm{V}$ and reverse. Figure 4.8 shows the reaction of the converter:

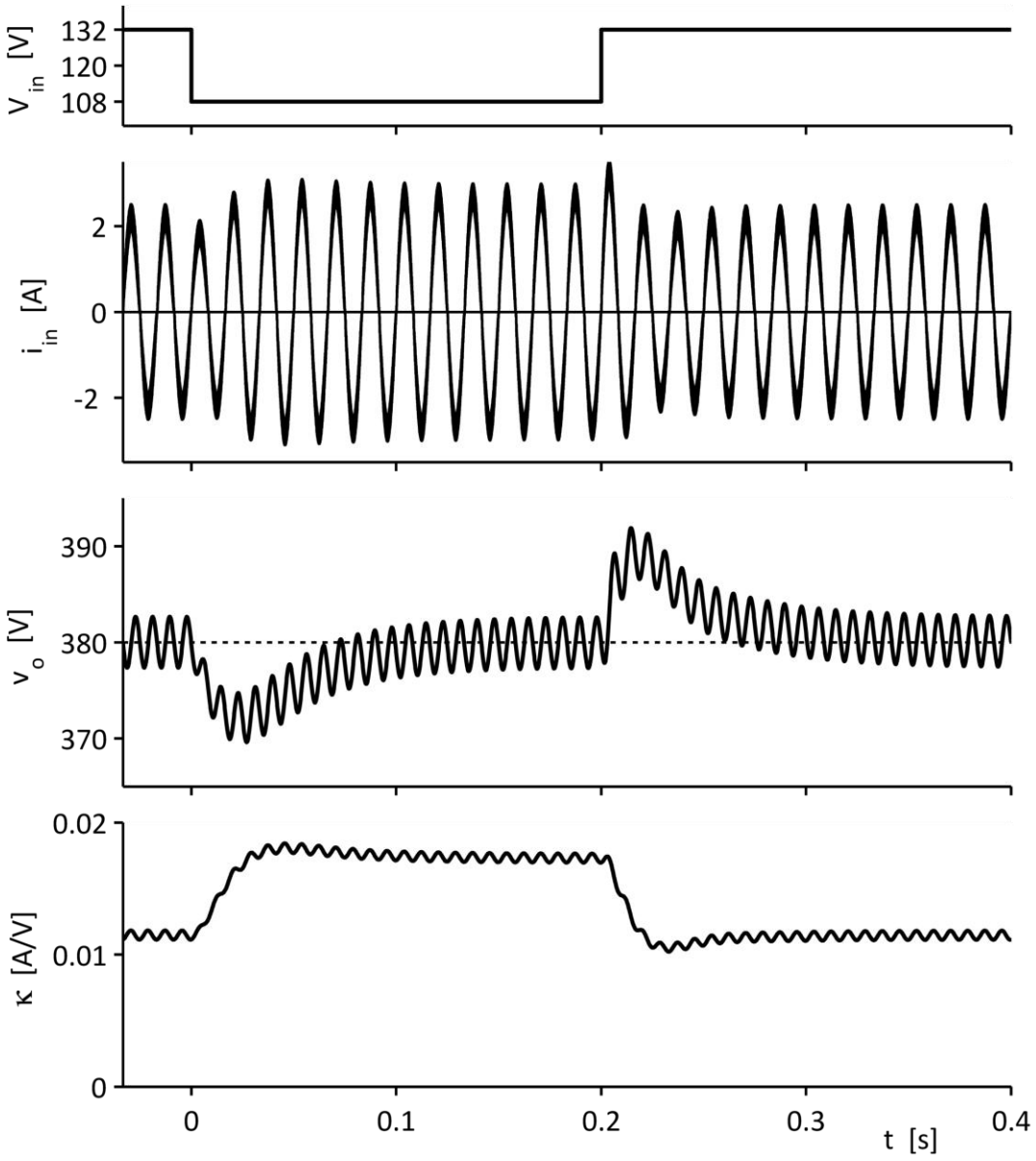


Figure 4.8: Simulated converter waveforms for step changes in the input voltage at $t = 0$ and $t = 0.2$ s

Upon the input voltage drop at $t = 0$, the output voltage controller strives to regulate the output voltage back to 380 V by raising its output κ and thereby commanding more current to be drawn at the input. The opposite occurs when the input voltage is suddenly decreased at $t = 0.2$ s. The control can easily handle these disturbances; both the level of the output voltage and the shape of the input current are unaffected once the converter reaches steady-state operation.

4.5 Adaptation of the inductor model parameters

In this section, simulation results are presented that verify the feasibility of the parameter estimation and adaptation. The schematic from Figure 4.1 was expanded to include the estimation and feedback parts described in sections 3.4 and 3.5. Figure 4.9 shows the full circuit that was used for the simulations. The subsystems therein are detailed in Figure 4.10 and Figure 4.11. The circuit in Figure 4.10 implements eqs. (3.9), (3.11) and (3.19), calculating the peak inductor current and the estimates of the inductor model parameters. The converter is first allowed to reach steady state with the initially implemented model parameters before, at $t = 0.5$ s, the model adaptation is activated (Figure 4.11).

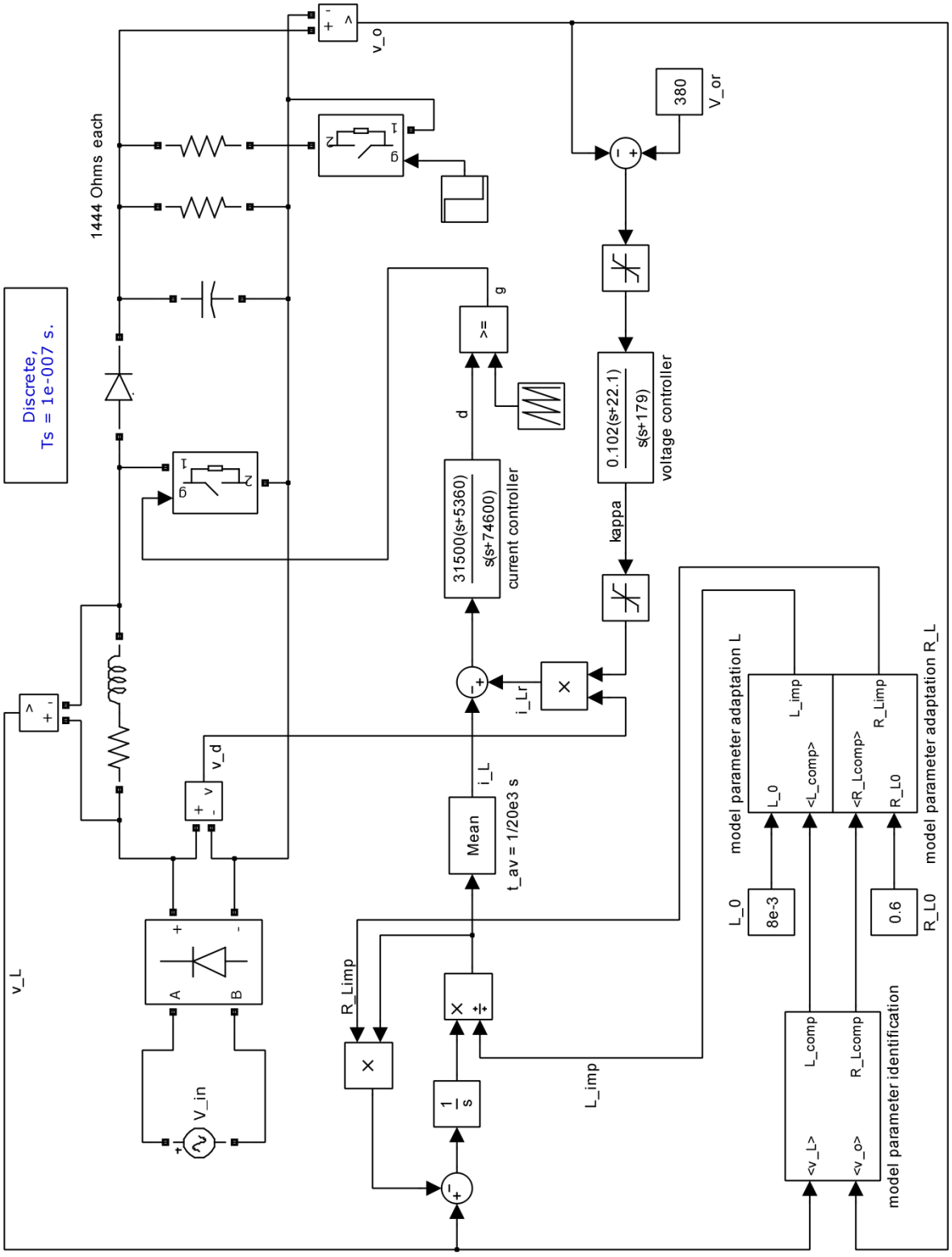


Figure 4.9: Simulation schematic of the circuit with model adaptation

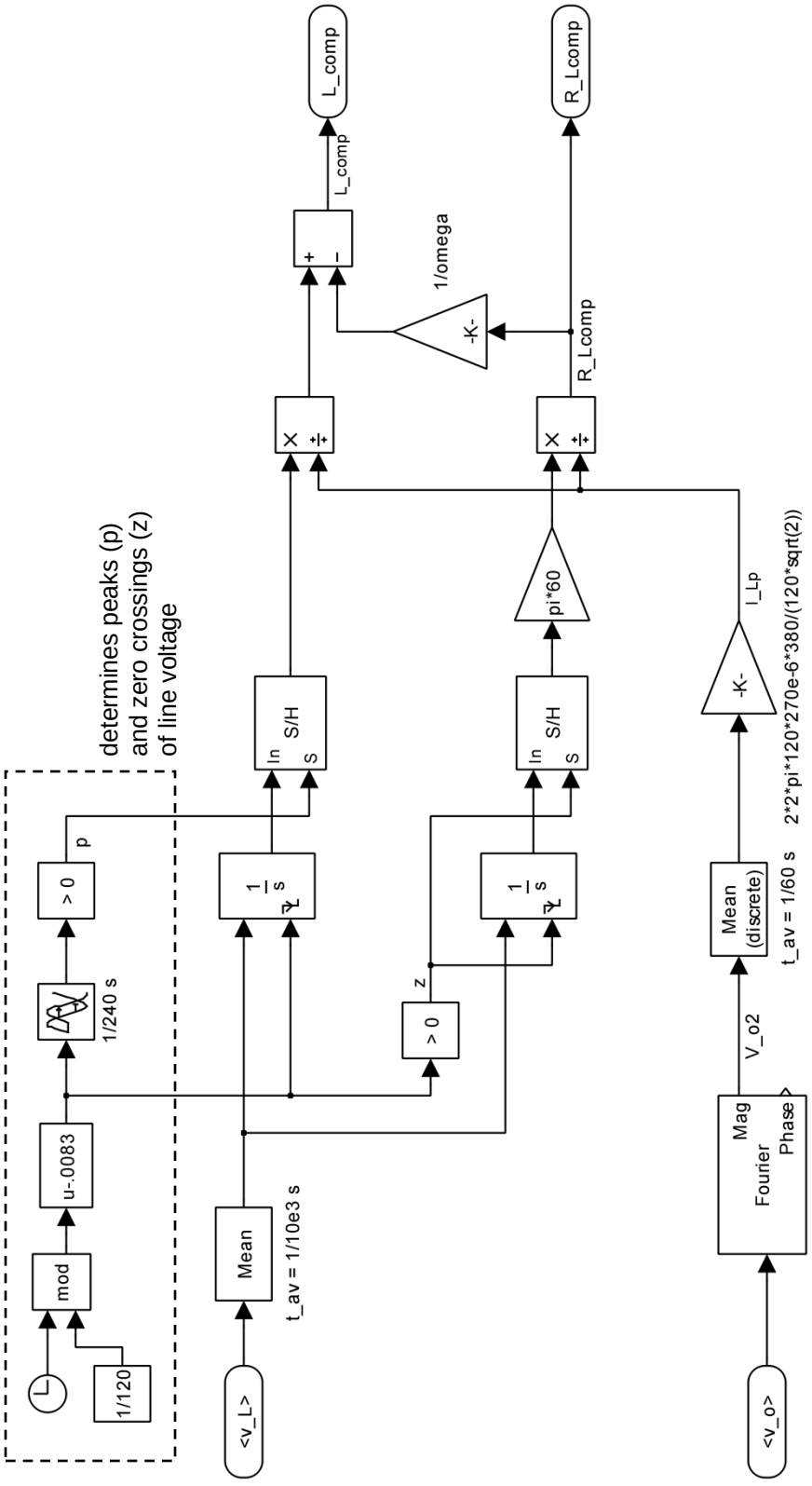


Figure 4.10: Subsystem with model parameter identification scheme

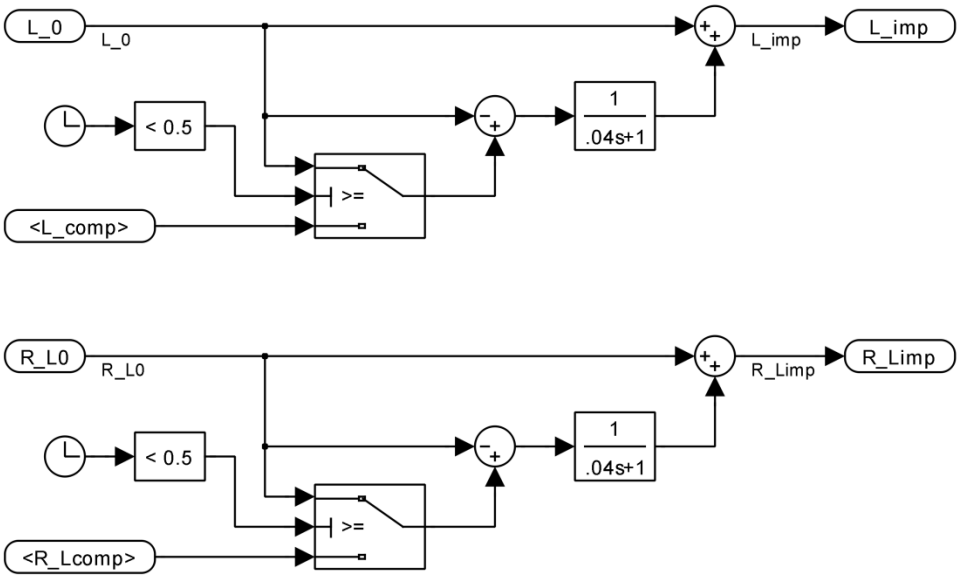


Figure 4.11: Subsystems with model parameter adaptation for L and R_L

In the following, the adaptation is investigated for various cases of parameter mismatches where L^0 and R_L^0 differ from L^{tr} and R_L^{tr}. The initial values for both parameters are given values in the range between one half and twice the true values. Figure 4.12 and Figure 4.13 illustrate the adaptation for some exemplary conditions. The plots show L^{imp} and R_L^{imp} with respect to time for different initial values. The converter has reached steady state when the adaptation is turned on at $t = 0$. The implemented parameters then asymptotically approach the estimated values. After the transients have decayed, L^{imp} fluctuates between 8.02 and 8.03 mH and R_L^{imp} reaches a value of 0.59 Ω irrespective of the initial values. These results are remarkably close to $L^{tr} = 8$ mH and $R_L^{tr} = 0.6$ Ω.

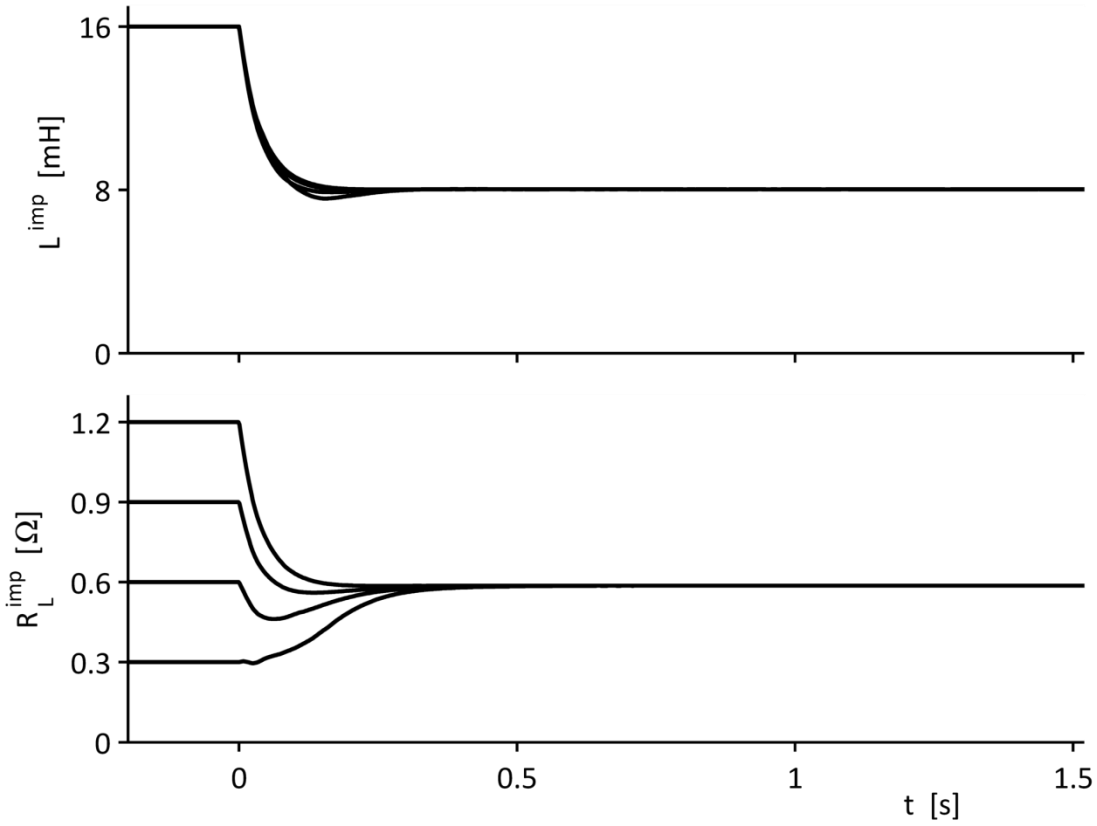


Figure 4.12: Inductor model parameter adaptation for $L^0 = 16$ mH and various values of R_L^0

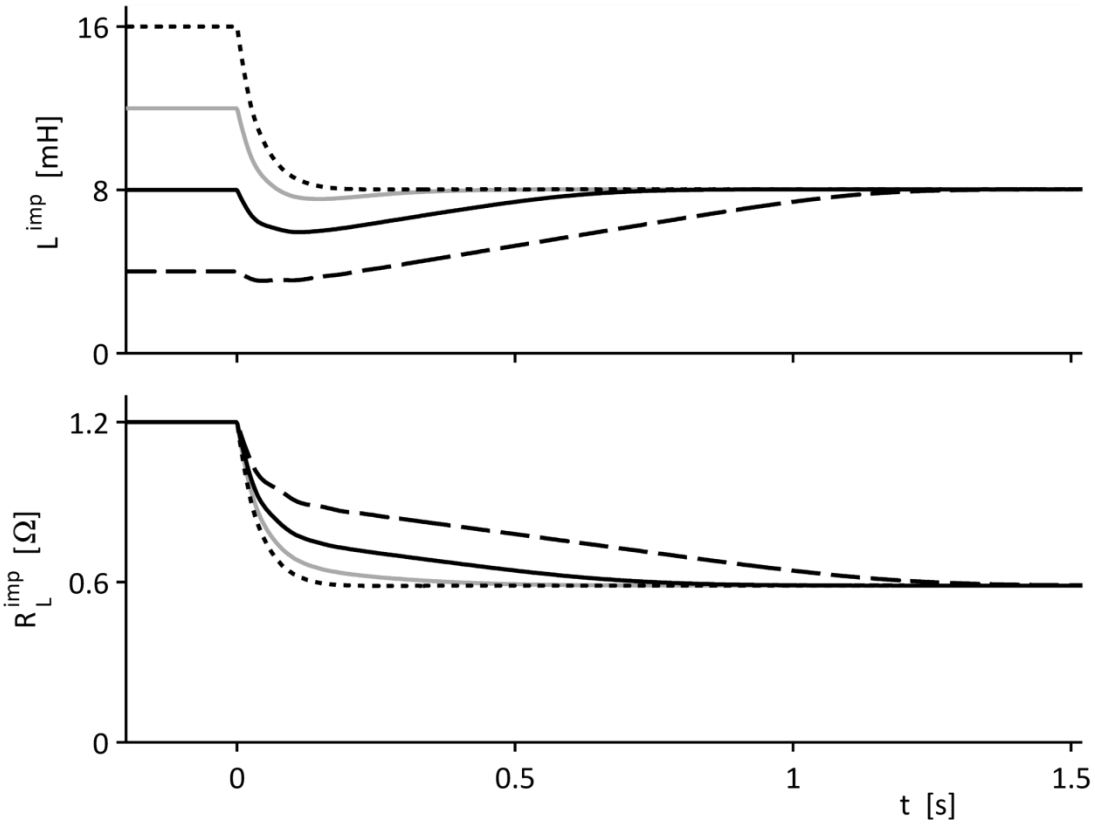


Figure 4.13: Inductor model parameter adaptation for $R_L^0 = 1.2\ \Omega$ and various values of L^0

Figure 4.12 and Figure 4.13 indicate that the model adaptation reaches steady-state faster the larger the initial values are. Shown in Figure 4.14, the course of the model adaptation can also be visualized in the L-R_L plane with time as parameter:

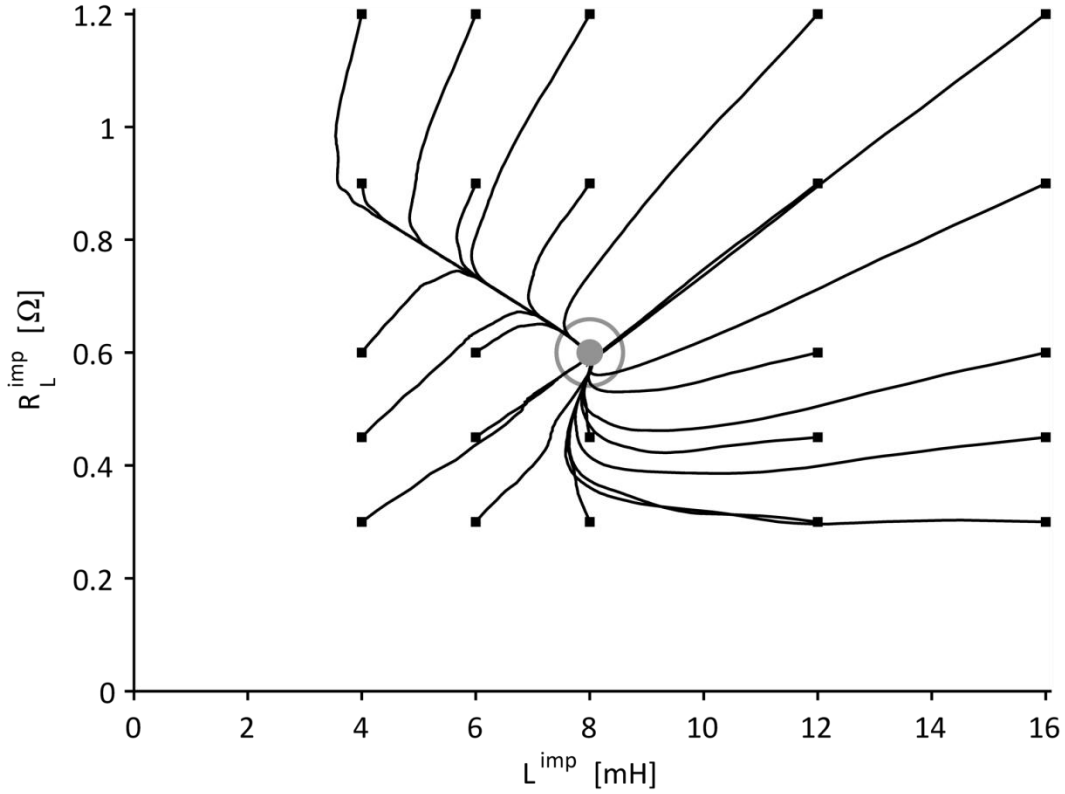


Figure 4.14: Trajectories of the implemented model parameters in the L-R_L plane for various pairs of initial values (squares)

It is apparent that the true parameter pair of 8 mH and 0.6 Ω is an attractor which all tested initial parameter pairs approach in the course of the closed-loop model adaptation. From that, it can be concluded that the model adaptation is asymptotically stable in the investigated range and has an equilibrium in very close vicinity to the true parameters. Extending the time in Figure 4.14 on a separate axis, the implemented model parameters form trajectories in the L-R_L-t space:

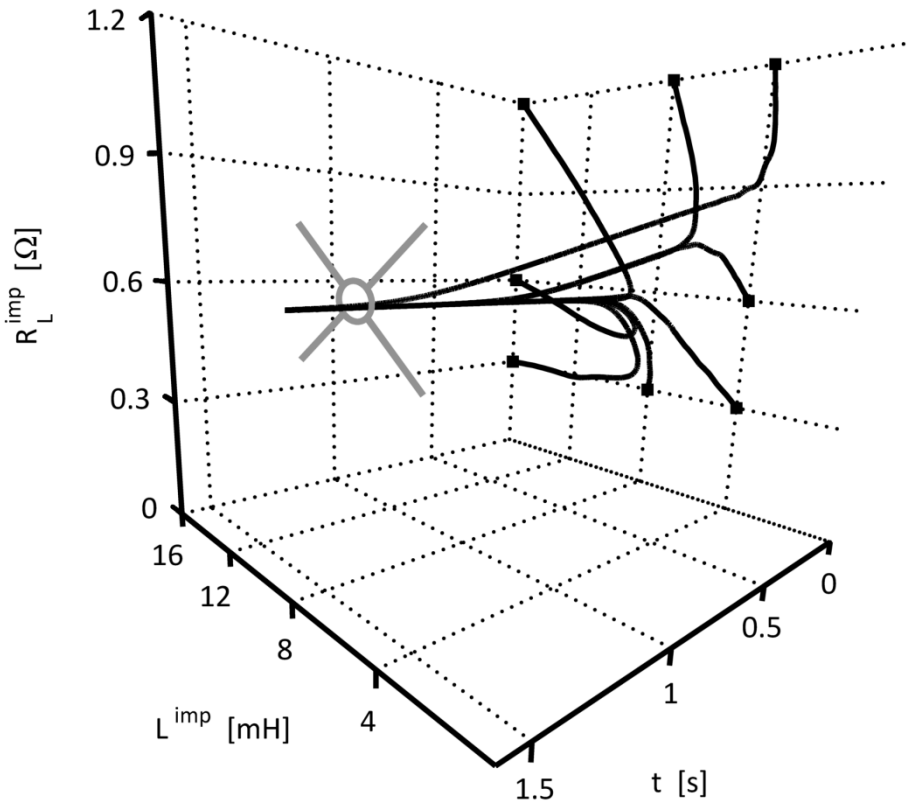


Figure 4.15: Trajectories of the implemented model parameters in the L-R_L-t space for various pairs of initial values (squares)

5 Hardware implementation

After designing the control circuit and verifying the operation via computer simulations, a prototype of a switch-mode rectifier was used to test the developed control method on hardware. Section 5.1 provides information on the converter circuit. The control is realized through a digital controller; details on the controller are given in section 5.2 and the controller implementation is explained in section 5.4.

5.1 Converter circuit

An existing converter that had been designed and built by a former student is reused to examine the performance of the new control scheme. Design of the prototype and selection of the circuit components are documented in [65]. Switching frequency, input and output voltage and power rating for this work match the original design specifications. The relatively low switching frequency of 20 kHz avoids strong electromagnetic interference and keeps the influence of the circuit elements' parasitic effects (e.g. the equivalent series resistance of the output capacitor, reverse recovery charge of the diode D) low. The inductance had been chosen so that the converter operates in continuous conduction mode at the rated load and the peak-to-peak input/inductor current ripple under this condition is 25 % of the peak current. Similarly, the bulk output capacitance C_1 had been set by the goal to achieve a peak-to-peak output voltage ripple of less than 1.5 % of the nominal output voltage. The design equations for both parameters had been taken from [1]. The dimensioning of the two main circuit elements L and C_1 is detailed in Appendix A.

The circuit contains a currents sense resistor R_c. It is located in the negative DC bus, the voltage across it indicating the negative of the inductor current. During regular operation in which the inductor current is not sensed, the resistor can be

short-circuited. It is included in the circuit to be able to test the current and voltage controllers without the inductor current being computed, as well as to gather reference data for later comparison with the computation of the inductor current active.

In addition to the main hardware components of the converter already mentioned in section 2.1, the circuit includes the following ones:

- diode D_b in parallel to the inductor L and the diode D. After the supply voltage to the converter is turned on, the output capacitor is rapidly charged to the peak of the rectified line voltage. The charging current is provided through a path bypassing the inductor through the surge bypass diode D_b to avoid saturating the magnetic core. Once the output voltage is greater than the peak of the rectified line voltage, D_b is reverse biased and has no effect on the circuit behaviour.
- output capacitors C_2 and C_3. The bulk capacitor C_1 is an aluminum electrolytic capacitor with a relatively high equivalent series resistance of 215 mΩ at 20 kHz [57]. To alleviate the adverse effects of the ESR on the output voltage ripple and the transient behaviour of the converter [14], two ceramic capacitors of smaller capacitances with lower ESRs were added in parallel to C_1.
- current sense resistor R_c, which is required to sense the inductor current when the circuit operates in test mode without computing the current. It can be bridged with jumper J_1 during regular operation.
- resistor R_{LL} in series with the inductor. If not bridged via jumper J_2, this resistor serves to test the model adaptation with a higher apparent equivalent series resistance.
- input filter capacitor C_f on the utility side of the rectifier. It reduces the switching noise propagating into the utility grid.
- fuse F directly at the input of the converter
- light emitting diode D_L indicating the presence of voltage at the converter output. Together with its series resistors R_7...R_9, it also serves as a minimum load for the converter.

Table 5.1 provides a summary of the hardware components of the converter:

Table 5.1: Hardware components of the converter

circuit element	type
bridge rectifier B	single phase diode bridge rectifier GBU806 $I_{(AV)} = 8$ A
inductor L	nominal inductance 8 mH core: toroidal iron powder core wire: enamelled copper wire for details see Appendix C
transistor Q	power MOSFET International Rectifier, IRF840A $V_{DSS} = 500$ V, $I_D = 8.0$ A mounted on heat sink
snubber capacitor C_s	metallized polyester film capacitor Kemet, R76PD1220SE00J $C_s = 2.2$ nF, rated voltage 630 V
snubber diode D_s	fast recovery rectifier diode Fairchild Semiconductor, EGP10J $V_{RRM} = 600$ V, $I_{F\ (AV)} = 3$ A $t_{rr} = 75$ ns ($I_F = 0.5$ A, $I_{RRM} = 1$ A)
snubber resistor R_S	cemented wirewound resistor Vishay, AC07000002001JAC00 $R_S = 2$ kΩ rated power 7 W

diode D	stealth diode Fairchild Semiconductor, ISL9R460PF2 $V_{RRM} = 600$ V, $I_{F\ (AV)} = 4$ A $t_{rr} = 17$ ns ($I_F = 1$ A, $V_R = 30$ V, $T_c = 25$ °C) mounted on heat sink
output capacitor C_1	aluminum electrolytic capacitor Panasonic, ECO-S2WB271DA $C_1 = 270$ µF, rated voltage 450 V
output capacitor C_2	ceramic capacitor AVX, SV09AC105KAR $C_2 = 1$ µF, rated voltage 1000 V
output capacitor C_3	ceramic capacitor TDK Corporation, FK26X7R2J103K $C_3 = 10$ nF, rated voltage 630 V
current sense resistor R_c	wirewound resistor with low-inductive Ayrton-Perry winding Vishay, MRA-05R5000FE12 $R_c = 0.5$ Ω, rated power 5 W
diode D_b	general purpose rectifier diode Fairchild Semiconductor, 1N5406 $V_{RRM} = 600$ V, $I_{F\ (AV)} = 3$ A

resistors $R_1 \ldots R_8$	rated power ¼ W unless otherwise noted $R_1 = 470$ kΩ $R_2 = 7.5$ kΩ $R_3 = 470$ kΩ $R_4 = 2.7$ kΩ $R_5 = 820$ kΩ, rated power ½ W $R_6 = 2.7$ kΩ $R_7 = 820$ Ω, rated power ½ W $R_8 = 15$ kΩ, rated power 25 W Caddock Electronics, MP925-15.0K-1% mounted on heat sink $R_9 = 820$ Ω, rated power ½ W
trimmer potentiometers $P_1 \ldots P_3$	$R_1 = R_2 = R_3 = 2$ kΩ rated power ¼ W
resistor R_{LL}	Ohmite, 15FR250E $R_{LL} = 0.25$ Ω, rated power 5 W
fuse F	fast-acting glass tube fuse, rated current 5 A
switch S	two-pole two-position miniature rocker NKK Switches, M2022TZW13-JB
light emitting diode D_L	5 mm red LED

The complete circuit diagram of the built prototype can be seen in Figure 5.1:

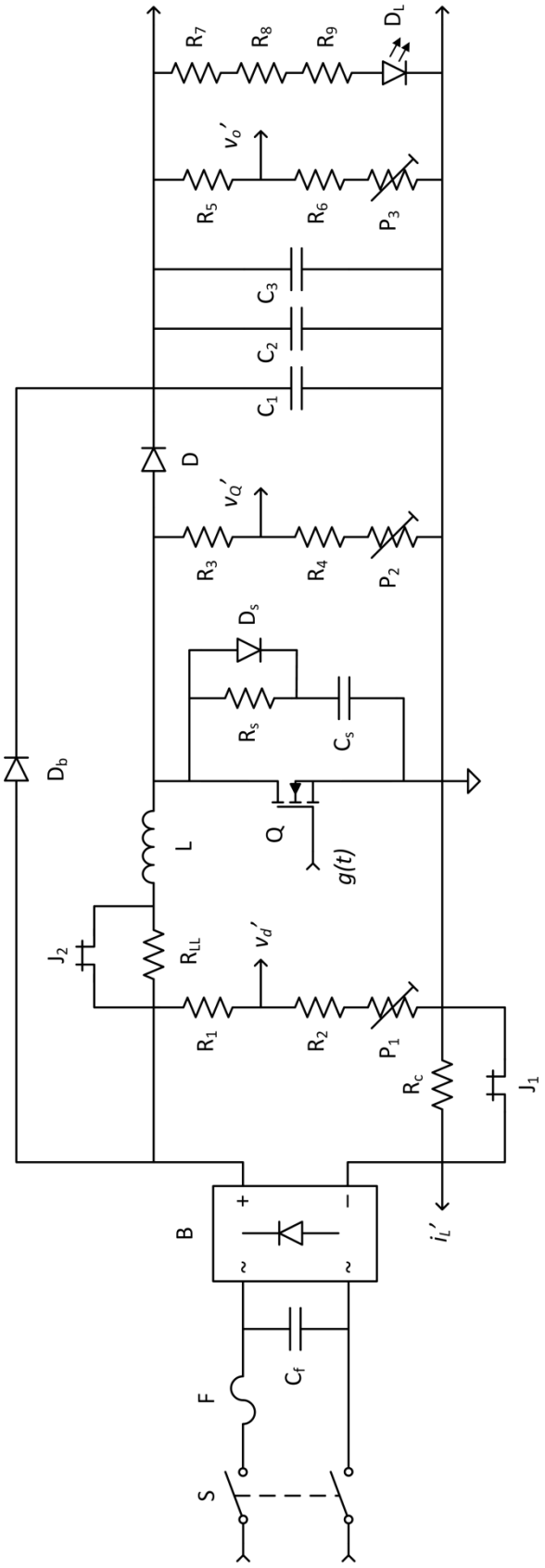


Figure 5.1: Schematic of the converter as implemented in hardware

All circuit components were soldered onto a single-sided prototyping stripboard in through-hole technology. The power semiconductors are passively cooled with heat sinks.

The operational amplifiers used for signal conditioning and the gate driver (detailed in subsections 5.5.1 and 5.5.2) require supply voltages of ±5 V and 15 V respectively. These are produced by a separate linear power supply on another circuit board.

While creating the circuit board layout, basic considerations were given to reducing electromagnetic interference. Generally, it was attempted to reduce the length of conductors by placing connected circuit elements close to one another. The components at the converter input, namely the switch, fuse, filter capacitor C_f, rectifier B and current sense resistor R_c, are lumped together in one corner of the circuit board. Adjacent to them are the parts that are subjected to the MOSFET drain-source voltage with its steep edges (inductor, transistor, its snubber circuit and diode D). As far as possible away from those is the circuitry that produces the low-level signals for the analog-to-digital converter to process (see subsection 5.5.1), which are particularly sensitive to interference. Return paths are kept close to the corresponding forward paths, and conductors mostly cross perpendicularly.

Figure 5.2 is a photo of the converter circuit board and the digital controller board with their interfaces labelled:

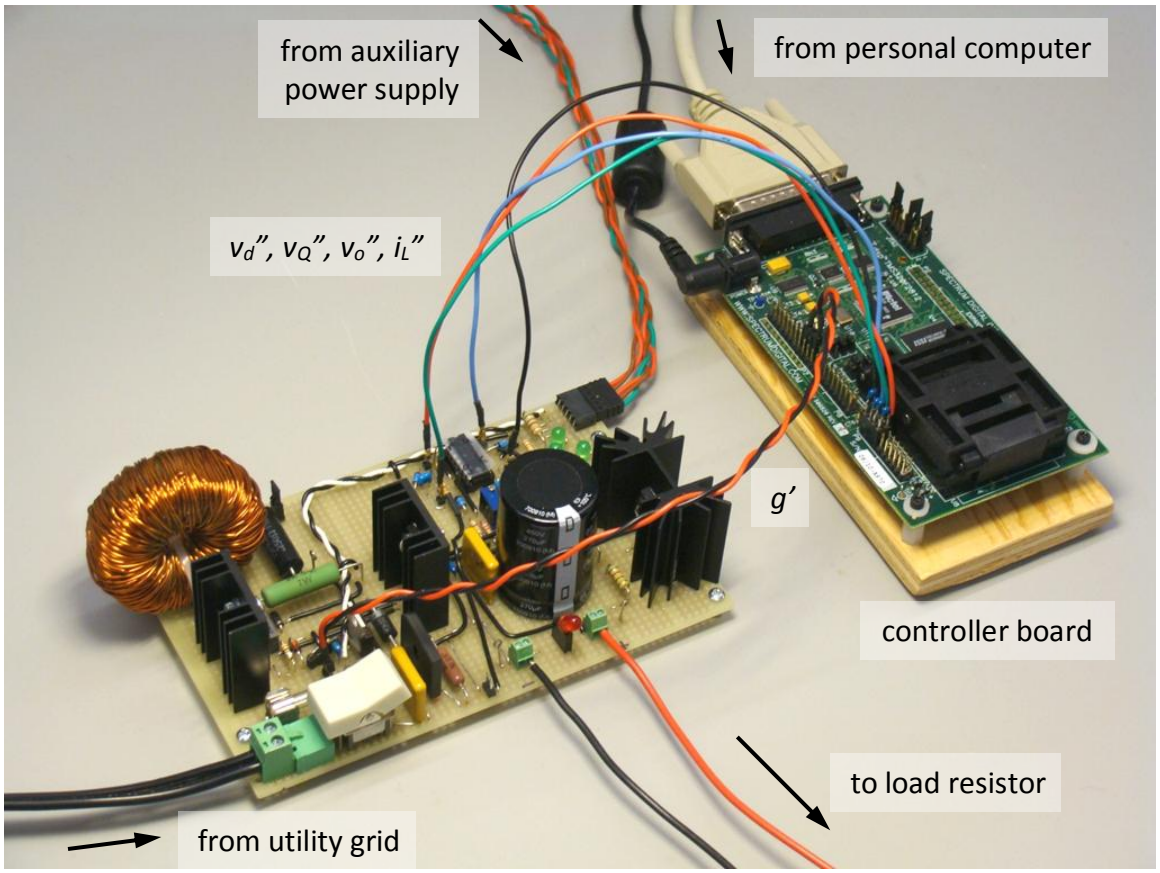


Figure 5.2: Converter circuit and controller board

5.2 Digital controller

A suitable controller was available from a previous project. It is a digital signal controller TMS320F2812 by Texas Instruments [66] embedded on an evaluation board eZdsp F2812 by Spectrum Digital [67,68]. The evaluation board is a printed circuit board that provides necessary hardware for the digital controller and allows using a variety of its capabilities. The board includes, for example, part of the power supply circuitry, an oscillator that generates the clock signal, a serial interface to establish a connection to a personal computer, memory chips, as well as connectors for the digital and analog interfaces.

The TMS320F2812 features a 32-bit fixed-point processor [69] that runs at a clock frequency of 150 MHz. It furthermore offers a number of functional modules and control peripherals, of which two are of interest for this application:

- analog-to-digital converter (ADC):
  The signal processor includes a 12-bit ADC that, by means of multiplexing, is capable of converting 16 voltage signals sequentially. The input voltages may assume values from 0 to 3 V. The signals that need to be acquired are v_d, v_Q, v_o and, for testing, i_L.
- event manager:
  The event manager bundles several timers and compare units. These functions are for instance required to produce pulse-width modulated output signals such as the gating signal for the transistor.

The eZdsp board possesses six populated jumpers on the component side of the board. They were set as follows:

Table 5.2: Jumper settings on the controller board

jumper	position	meaning
JP1	2-3	select microcomputer mode Microcomputer mode needs to be selected to access the boot read-only memory that contains look-up tables for required mathematical operations and functions.
JP9	1-2	enable phase-locked loop The phase-locked loop is needed to obtain a processor clock frequency of 150 MHz from the oscillator frequency of 30 MHz.
JP7	2-3	boot mode: jump to H0 SARAM, memory location 3F8000
JP8	2-3	
JP11	1-2	
JP12	2-3	

5.3 Discretization of transfer functions

Frequency dependencies of the inductor model (section 3.2) as well as the converter and its controllers (sections 3.7 and 3.8) have hitherto been described as continuous elements in the s domain. For the digital controller to deal with time varying quantities, they must be handled in discrete-time form.

Several methods exist for finding an equivalent discrete-time transfer function for a given continuous-time one [70]. The various possibilities arise from different ways of approximating the response between the sampling times. One of them is the bilinear transformation method. It approximates the integral of a signal via the trapezoidal method. Compared to other methods, the bilinear transformation is a compromise between accuracy in the transformation and mathematical complexity. This method will be used in the following.

Two types of transfer functions need to be transformed into the z domain:

1) the first-order low-pass filter that is used for the inductor model (section 3.2) and the inductor model parameter adaptation (section 3.5) and
2) the controller transfer functions (sections 3.7 and 3.8).

Consider a general transfer function G of a first-order low-pass filter that has an input X and an output signal Y

$$G(s) = \frac{Y(s)}{X(s)} = \frac{1}{1 + s\tau} = \frac{1}{1 + \frac{s}{\omega_0}} = \frac{\omega_0}{s + \omega_0} \tag{5.1}$$

with the time constant τ and the cutoff frequency ω_0. The bilinear transformation is carried out by substituting s in eq. (5.1) with $2f_s \frac{1-z^{-1}}{1+z^{-1}}$ to obtain the transfer function of the equivalent discrete system in the z domain:

$$G(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 + 2\tau f_s \frac{1 - z^{-1}}{1 + z^{-1}}} = \frac{1}{1 - 2\tau f_s} \frac{z^{-1} + 1}{z^{-1} + \frac{1 + 2\tau f_s}{1 - 2\tau f_s}} \tag{5.2}$$

The z domain transfer function in eq. (5.2) depends on the sampling frequency f_s. It can be written as a difference equation by multiplying with the denominators of both sides of the equation:

$$z^{-1}\,X(z) + X(z) = (1 - 2\tau f_s)\,z^{-1}\,Y(z) + (1 + 2\tau f_s)\,Y(z) \tag{5.3}$$

Doing the inverse z-transform of eq. (5.3), one obtains

$$x_{k-1} + x_k = (1 - 2\tau f_s)\,y_{k-1} + (1 + 2\tau f_s)\,y_k \tag{5.4}$$

and after solving for y_k:

$$\begin{aligned} y_k &= \frac{x_k + x_{k-1} + (2\tau f_s - 1)\,y_{k-1}}{2\tau f_s + 1} \\ &= \frac{2\tau f_s - 1}{2\tau f_s + 1}\,y_{k-1} + \frac{1}{2\tau f_s + 1}\,(x_k + x_{k-1}) \end{aligned} \tag{5.5}$$

The difference equation of the inductor model can also be expressed in terms of L and R_L:

$$i_L(k) = \frac{2Lf_s - R_L}{2Lf_s + R_L}\,i_L(k-1) + \frac{1}{2Lf_s + R_L}\,[v_L(k) + v_L(k-1)] \tag{5.6}$$

The found discrete-time transfer function $G(z)$ from eq. (5.2) was compared to the results from the bilinear transformation method with frequency prewarping and the matched pole-zero mapping method. Differences between the three are only marginal. This can be attributed to the sampling frequency being several orders of magnitude higher than the cutoff frequencies of interest.

The same steps are carried out with the controller transfer functions that are of the form

$$G_c(s) = \frac{Y(s)}{X(s)} = K\,\frac{s + \omega_z}{s(s + \omega_p)} \tag{5.7}$$

Substituting $s = 2f_s\,\frac{1-z^{-1}}{1+z^{-1}}$ yields

$$G_c(z) = \frac{Y(z)}{X(z)} = \frac{K}{2f_s}\,\frac{1+z^{-1}}{1-z^{-1}}\,\frac{2f_s\dfrac{1-z^{-1}}{1+z^{-1}}+\omega_z}{2f_s\dfrac{1-z^{-1}}{1+z^{-1}}+\omega_p} \tag{5.8}$$

and the corresponding difference equation

$$\begin{aligned}&\left(2f_s+\omega_p\right)Y(z) - 4f_s\,z^{-1}\,Y(z) + \left(2f_s-\omega_p\right)z^{-2}\,Y(z)\\ &= \frac{K}{2f_s}\left[\,(\omega_z+2f_s)\,X(z) + 2\omega_z\,z^{-1}\,X(z) + (\omega_z-2f_s)\,z^{-2}\,X(z)\,\right]\end{aligned} \tag{5.9}$$

Transforming eq. (5.9) into the time domain, one obtains

$$\begin{aligned}&\left(2f_s+\omega_p\right)\cdot y_k - 4f_s\cdot y_{k-1} + \left(2f_s-\omega_p\right)\cdot y_{k-2}\\ &\qquad = \frac{K}{2f_s}\left[\,(\omega_z+2f_s)\cdot x_k + 2\omega_z\cdot x_{k-1} + (\omega_z-2f_s)\cdot x_{k-2}\,\right]\end{aligned} \tag{5.10}$$

and after solving for y_k:

$$\begin{aligned}y_k = {}&\frac{K}{2f_s}\,\frac{\omega_z+2f_s}{\omega_p+2f_s}\cdot x_k + \frac{K}{2f_s}\,\frac{2\omega_z}{\omega_p+2f_s}\cdot x_{k-1} + \frac{K}{2f_s}\,\frac{\omega_z-2f_s}{\omega_p+2f_s}\cdot x_{k-2}\\ &+ \frac{4f_s}{\omega_p+2f_s}\cdot y_{k-1} + \frac{\omega_p-2f_s}{\omega_p+2f_s}\cdot y_{k-2}\end{aligned} \tag{5.11}$$

Fitting eq. (5.11) to the general form of a second-order system

$$y_k = b_0\,x_k + b_1\,x_{k-1} + b_2\,x_{k-2} + a_1\,y_{k-1} + a_2\,y_{k-2} \tag{5.12}$$

reveals that the controller coefficients are

$$\begin{aligned}b_0 &= \frac{K}{2f_s}\,\frac{\omega_z+2f_s}{\omega_p+2f_s}\\ b_1 &= \frac{K}{2f_s}\,\frac{2\omega_z}{\omega_p+2f_s}\\ b_2 &= \frac{K}{2f_s}\,\frac{\omega_z-2f_s}{\omega_p+2f_s} = b_1 - b_0\\ a_1 &= \frac{4f_s}{\omega_p+2f_s}\\ a_2 &= \frac{\omega_p-2f_s}{\omega_p+2f_s} = 1 - a_1\end{aligned} \tag{5.13}$$

5.4 Control program

The digital controller is programmed in the language C with the software Code Composer Studio [71]. Code Composer Studio is a development environment for Texas Instruments embedded processors. The source code that was written for this project is attached in Appendix B. In the following, the operating principle of the control program and the source code in the file *main.c* will be described.

5.4.1 Operating principle

After the initialization upon program start, the program enters an infinite loop and hits a conditional statement. Its condition is at first not fulfilled so that no further commands are executed. Meanwhile, in the background, the analog-to-digital converter samples the analog input signals from the converter. Once 40 samples of each signal have been collected, the condition in the infinite loop of the main routine becomes true. Now all computations that are needed during the operation of the converter—as described below—are carried out. When they are completed, the processor idles again in the infinite loop. The ADC constantly operates in the background and issues interrupts every time a conversion sequence is complete. When another 40 samples have been converted, the computation routine in the main loop is entered once again. Effectively, the computations determining the duty cycle of the current switching cycle use the sampled values that were acquired during the previous switching cycle. A detailed description of the source code is given in subsection 5.4.4.

5.4.2 Timing considerations

The highest processor clock frequency at which the controller can operate is 150 MHz. This frequency is selected. Peripherals use their own clocking signals, which are obtained from the CPU clock through frequency dividers (clock prescalers). Table 5.3 lists all clock signals that are used in this application:

Table 5.3: Digital controller clocking

clock signal	frequency	use
SYSCLKOUT	150 MHz	processor clock
HSPCLK	150 MHz	high-speed peripheral clock used by event managers (timers)
ADCCLK	25 MHz	ADC clock

While the sampled signals v_d, v_o and i_L are continuous, the MOSFET drain-source voltage v_Q ideally exhibits discontinuities when the transistor is turned on or off. To obtain accurate values of its switching cycle average for the computation of i_L, v_Q must be sampled often. With the limited number of computation cycles available during one switching cycle in mind, it was decided to take 40 samples per switching cycle (`NSPSC`=40). Then the uncertainty in $\overline{v}_Q$ is $\frac{1}{40} = 2.5\%$ and the sampling frequency would be $f_s = 40 \cdot 20\,\text{kHz} = 800\,\text{kHz}$. This frequency is generated by timer 1. The relationship between the desired frequency f_s and the corresponding period register value N is

$$N = \frac{f_{clk}}{f_s} - 1 \tag{5.14}$$

where f_{clk} is the high-speed peripheral clock frequency (150 MHz) [72]. Therefore, to generate a frequency of 800 kHz, N would have to be $\frac{f_{clk}}{f_s} - 1 = \frac{150\,\text{MHz}}{800\,\text{kHz}} - 1 = 186.5$. The register value however must be a 16-bit integer; 187 is selected instead. The resulting sampling frequency is $f_s = \frac{f_{clk}}{N+1} = \frac{150\,\text{MHz}}{187+1} = 797.872\ldots\,\text{kHz}$. To maintain the integer ratio of sampling frequency to switching frequency, the latter needs to be adjusted to $f_{sw} = \frac{f_s}{40} = 19.9468\ldots\,\text{kHz}$. This frequency is generated by timer 2 with its period register set to $N = \frac{f_{clk}}{f_s/40} - 1 = 40 \cdot (187 + 1) - 1 = 7519$.

The processor has to compute 19,947 duty cycles per second. With the processor operating at a frequency of 150 MHz, one switching cycle lasts $\frac{f_{clk}}{f_{sw}} = 7520$ processor

clock cycles. As long as the converter operates in continuous conduction mode at the extrema of the line voltage, which is the case except at very small input currents, the minimum duty cycle is always greater than 0.5. Then about $7520 \cdot 0.5 = 3760$ clock cycles are available for the computations until the compare register of timer 2, that generates the gating signal, has to be updated. The interrupt service routine triggered by the ADC takes about 40 clock cycles to complete, such that roughly $3760 - 20 \cdot 40 = 2960$ cycles remain to do all the computations that process the sampling values and compute the duty cycle.

5.4.3 IQmath library

The TMS320F2812 is a fixed-point processor. Operations with float and double variables are possible, but use much computation power. The IQmath library allows using floating-point numbers at higher execution speeds than standard float variables. The concept is to store floating-point numbers as 32-bit fixed-point numbers with a certain available range and numerical precision. The conversion from rational numbers to IQ numbers is done by multiplying the floating-point number with a factor such that the product extends over the 32-bit integer range. All operations carried out with IQ numbers are then integer operations. [73]

When using IQ numbers, operations other than addition and subtraction must use special functions that are provided with the IQ library. The most commonly used one is the one for multiplications, `_IQmpy`. It takes two factors as arguments, necessitating the syntax `product=_IQmpy(factor1,factor2)`. Additionally, the controller memory contains look-up tables for mathematical operations and functions in the IQ domain such as division and the sine function.

The global IQ format, the standard format for IQ variables, is chosen as IQ22. The IQ22 format can represent values in the range $[-512, 512)$ with a resolution of 0.000,000,238. This range and accuracy are suitable for most variables. Values that exceed the range or are small enough to require a higher precision are shifted right or left by a number of bits, for example the sum of 10 samples of v_d in the line

voltage zero crossing detection algorithm (subsection 5.4.5). Higher precision is needed in the computation of κ because the bandwidth of the output voltage control loop is very small with respect to the switching frequency. The voltage controller coefficients and κ are therefore represented in the IQ30 format that provides the highest resolution (10^{-9} for the range $[-2,2)$).

5.4.4 Description of the source code

Apart from the contents of the file *main.c* that are listed in Appendix B, the program relies on files that are provided by Texas Instruments for this device [73,74]. The source code contains detailed comments for every command or functional entity.

The variables `v_d`, `v_o`, `v_L` and `i_L` that are used throughout the program refer to the switching cycle averages of the respective quantities. All values of $\overline{v}_d$ during each line half-cycle are stored in the array `v_dLHC`. The elements of this vector are needed for computing the peak input voltage, as well as for the averaging of $\overline{v}_d$ to determine the line voltage zero crossing instants as described in subsection 5.4.5.

The integral of any time-dependent signal over time is approximated by numerical integration via the rectangle method. This method turns the integration of the continuous-time signal into a summation of the samples of the discrete-time signal in the interval between the sampling instants that correspond to the limits of integration: $\int_{t_1}^{t_2} v(t)\,dt \approx \frac{1}{f_{sw}} \sum_{k_1}^{k_2} v_k$.

The program file *main.c* contains three functions:

1) `main`, the function at which the program execution starts,
2) `adc_isr`, the interrupt service routine triggered when the ADC has finished a conversion sequence, and
3) `shutdown` that turns the converter off.

Details of the three functions are given below. Figure 5.3 visualizes the operation sequence of the control program in form of a flowchart:

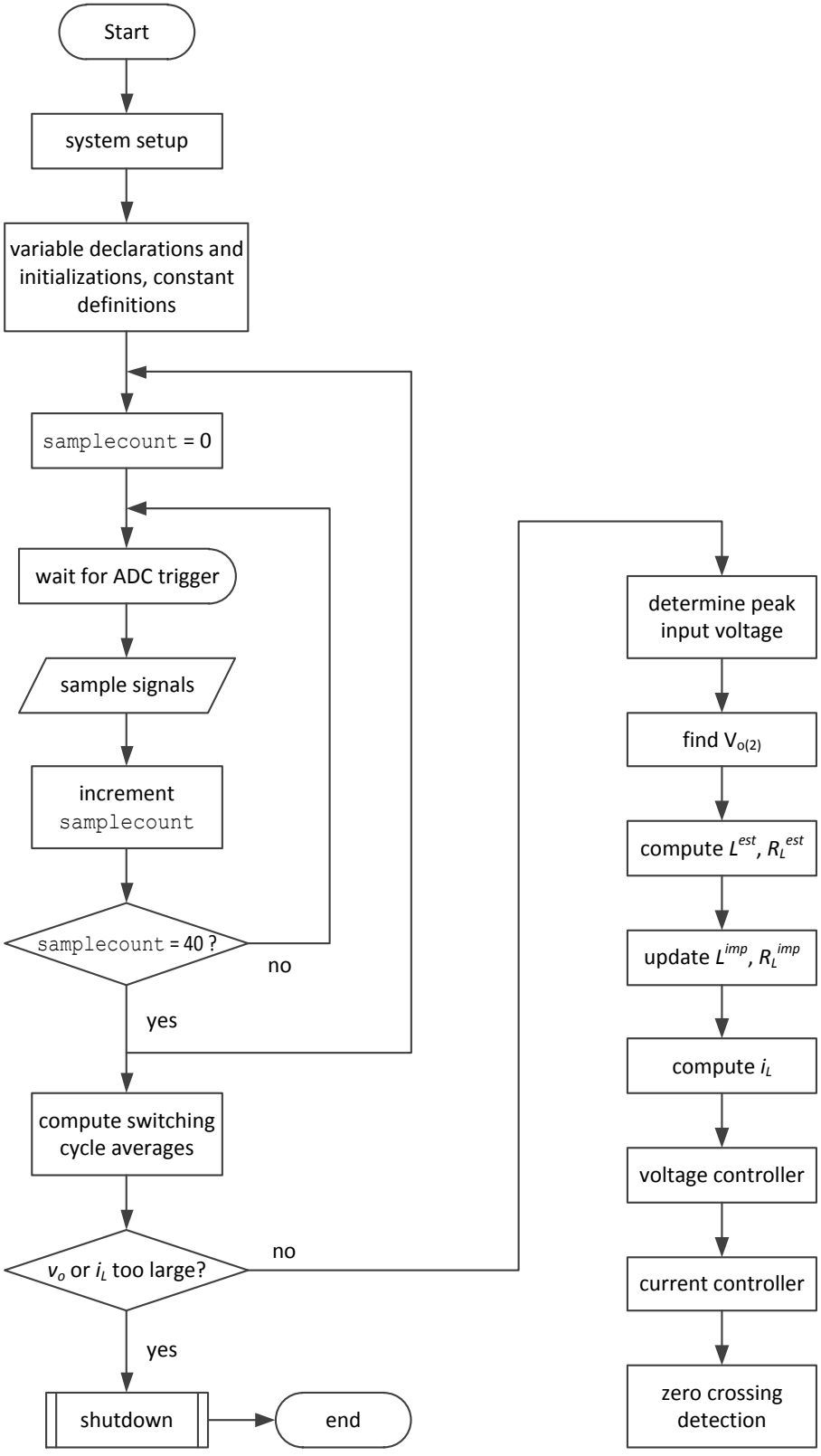


Figure 5.3: Operation sequence of the controller program

First, a number of hardware and system settings are made for system parameters and peripherals, including:

- disabling the watchdog timer, specifying the phase-locked loop multiplier for the processor clock, setting the frequency for and enabling peripheral clocks,
- configuring all general purpose input/output pins as digital outputs and setting them to low, except the general purpose timer 1 and 2 compare outputs that are set as peripheral outputs,
- setting the general purpose timer 1 and 2 period registers, initializing their compare and counter registers, setting timer 1 to start an ADC conversion sequence upon period match,
- powering up the analog-to-digital converter, setting the ADC clock frequency divider, specifying the input channels to be converted,
- enabling interrupts and defining `adc_isr` as the ADC end-of-sequence interrupt service routine.

Next, the variables are declared and initialized. All variables are created as static variables. This is done because automatic (non-static) variables are stored on the stack, and the stack can only hold up to 1024 words (1 word = 16 bits). When the variables are declared with storage class static, they are allocated for the entire duration of the program execution and reside in a different memory section that has more space available [75,76]. Values that do not change during program execution and that do not have to be computed are defined as symbolic constants, e.g. the switching frequency and reference output voltage. Mathematical terms that do not change while the converter operates, but that do require computation, are computed before the converter operation starts to reduce the number of calculations that need to be carried out while the converter operates.

After that, the program enters an infinite loop with a conditional statement and keeps converting the input signals in the background until the samples from one switching cycle have been captured. This incident is recognized by testing the variable `samplecount` to equal the number of samples per switching cycle. `samplecount` is incremented in every call of the ADC end-of-sequence interrupt

service routine and reset when the captured samples are processed. To obtain the switching cycle averages of the sensed voltages and the current from the sums of the conversion result register values `v_dsum`, `v_osum`, `v_Qsum` and `i_Lsum`, the sums are multiplied by individual factors. These factors take into account

1) the division of the sums by the number of samples per switching cycle, completing the switching cycle averaging after the summation done in the interrupt service routine,
2) the right shifting of the conversion result register values. The 12-bit values are left-justified in the 16-bit registers, that is, the conversion results occupy bits four to fifteen of the registers after the ADC stores them there. To become valid data, they need to be right-aligned like all other variables.
3) the mapping from the shifted conversion result values to the voltages/current occurring in the converter.

The mathematical relationships for the latter two steps were determined experimentally (ADC calibration, see Appendix C).

Next, the program tests whether the output voltage or the inductor current is too large. If either is the case, the function `shutdown` is called and the gating signal is immediately forced to low, the error flag pin (see subsection 5.5.3) is set to high, all interrupts are disabled and the program returns from the main function. This turns off the converter and effectively terminates the control program. Powering up the converter again requires resetting the controller. The two conditions constitute faults and cannot occur if the converter operates correctly. Testing for them is done for the sake of preventing hardware damage should an error occur in the computation of the switching cycle.

The peak of the input voltage is needed to compute the reference value of the inductor current, the threshold value for the line voltage zero crossing detection algorithm (subsection 5.4.5) and the estimates of inductance and equivalent series resistance. To obtain the peak value, four values of v_d around $t = \frac{T}{4}$ are added up and divided by four. The point $t = \frac{T}{4}$ is recognized by testing whether the line half-

cycle index `LHCcount` is within the range [`NSCPLHC2`-2, `NSCPLHC2`+1]. This averaging makes the determined peak value less susceptible to noise.

Subsequently, the controller determines the peak value of the output voltage ripple by the technique described in subsection 5.4.6.

Then the inductance and equivalent series resistance estimates are calculated as per the formulas given in section 3.4. Updating the implemented inductor parameters is done by low-pass filtering their errors; refer to section 3.5 for full details of the model adaptation. A discrete-time low-pass filter algorithm was derived in section 5.3. Equation (5.5) is implemented in the form $\texttt{L\_corr[0]} = \texttt{fL1} \cdot \texttt{L\_corr[1]} + \texttt{fL2} \cdot (\texttt{deltaL[0]} + \texttt{deltaL[1]})$ in the case of L, and likewise for R_L. The factors `fL1` and `fL2` are computed during the program initialization.

With the inductor model now updated, it can be used to compute the inductor current (eq. (5.6)).

Afterwards, the voltage controller computes a new value of κ using the discrete-time controller equation from section 5.3. With regard to the output voltage regulation, the control frequency of 20 kHz is much larger than the control loop bandwidth of 10 Hz. Consequently, the changes in κ from one switching cycle to the next one are very small and lie in the order of the resolution of the IQ22 format. This makes it necessary to choose a different number format (IQ30) for κ as well as the controller coefficients as already mentioned in subsection 5.4.3. Furthermore, the controller coefficients `b0v`, `b1v` and `b2v`, that are multiplied with the output voltage errors and have magnitudes of as low as $2 \cdot 10^{-9}$, are shifted leftwards by 16 bits (multiplied by 2^{16}) for a sufficiently accurate representation in the processor. Compensation is done via right-shifting the voltage errors by 8 bits and right-shifting the sum of the three products of voltage errors and controller coefficients by another 8 bits.

The next step is computing the duty cycle for the current switching period. As reference current, the expression

$$i_L^* = \kappa \cdot \hat{V}_d \cdot \sin\theta \tag{5.15}$$

is used during each line half-cycle. It is implemented in the form

$$i_L^* = \texttt{V\_dp} \cdot \texttt{sine}[\texttt{LHCcount}] \cdot \texttt{kappa}[0] \tag{5.16}$$

where `sine` is a look-up table whose elements are computed during the program initialization such that it takes `LHCcount` directly as its index. Once the new duty cycle is computed with eq. (5.12), the processor updates the compare register of timer 2 by multiplying the period register value with d.

Finally, the controller decides if the line voltage crossed zero during the previous switching-cycle. Refer to subsection 5.4.5 for a description of the principle behind the zero crossing detection. If the program decides that a zero crossing occurred, a number of actions are taken on this significant occasion. For instance, `LHCcount` is reset to zero, as is the computed inductor current to avoid integrator drift.

`adc_isr` is the end-of-sequence interrupt service routine of the analog-to-digital converter. It is called when the ADC has finished a conversion sequence, interrupting the program flow of the main function. It adds the conversion result register values of the four sampled signals to their corresponding sums `v_dsum`, `v_osum`, `v_Qsum` and `i_Lsum`. All further processing is done in the main function after all samples from the current switching cycle are captured. Finally, `samplecount` is incremented by one.

5.4.5 Line voltage zero crossing detection

An important program module is the detection of the line voltage zero crossings. The point where $\theta = 0$ serves as a reference point for the computation of the reference inductor current and is used to determine the peak of the input voltage as well as the peak-to-peak output voltage ripple.

The signal that is sampled is the full-wave rectified line voltage. Thus, it inherently does not cross zero. Further, the sampled signal may actually reach zero at the line

voltage zero crossing, or may not reach zero and have a minimum that is larger than zero. Nonlinearity in the ADC can also cause the sampled signal to assume zero at several subsequent sampling instants around the line voltage zero crossing. Lastly, the line voltage is distorted and contaminated by switching noise from the converter and other noise coming from the grid. All these factors complicate the reliable and accurate identification of the time when the line voltage reaches zero.

As the rectified line voltage approaches zero during the second half of its cycle, a moving average is computed over a certain fraction of the period. The average lags the signal in time, and therefore assumes a positive value when the rectified line voltage ideally touches zero (Figure 5.4). The averaging eliminates all high frequency noise and also alleviates errors in the signal acquisition at low voltages. Comparing the moving average to a threshold value is more precise indicator than testing the rectified line voltage for zero.

If the average is computed as

$$\bar{\bar{v}}_d(t) = \frac{1}{t_{av}} \int_{t-t_{av}}^{t} v_d(t')\, dt' \tag{5.17}$$

for the rectified line voltage having an waveform described by $v_d = \hat{V}_d\, |\sin \omega t|$, then the value that the average assumes at the line voltage zero crossing is

$$\bar{\bar{v}}_d\left(\frac{T}{2}\right) = \frac{\hat{V}_d}{t_{av}} \int_{T/2-t_{av}}^{T/2} \sin \omega t'\, dt' = \frac{\hat{V}_d}{\omega t_{av}} \left[1 + \cos\left(\pi - \frac{t_{av}}{T/2}\pi\right)\right] \tag{5.18}$$

The average is computed over 10 switching cycles, that is $t_{av} = \frac{10}{f_{sw}} \approx \frac{10}{20\text{ kHz}} =$ 0.5 ms. The threshold value $\bar{\bar{v}}_d\left(\frac{T}{2}\right)$ can now be computed during the operation as a function of the previously detected $\hat{V}_d$ only. Under nominal conditions where $V_{in} = 120$ V, $\bar{\bar{v}}_d\left(\frac{T}{2}\right) = 15.9$ V.

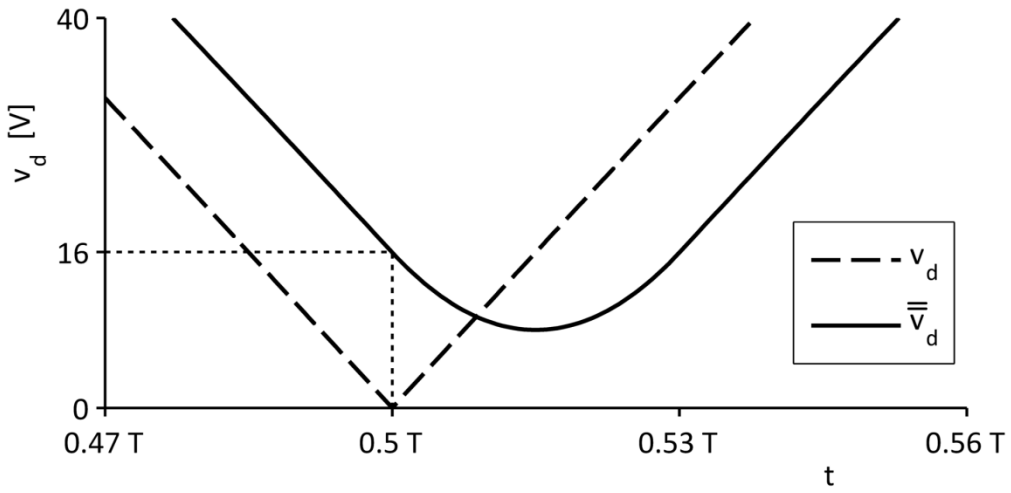


Figure 5.4: Technique to detect the line voltage zero crossing

It was found that with the algorithm implemented as described, the line voltage zero crossing is signalled slightly too late. This might be due to a significant content of low-order harmonics in the line voltage that flatten the peak and cause a distortion in the downward slope. To compensate, the factor by which $\hat{V}_d$ is multiplied to obtain the threshold value was adjusted slightly such that the line voltage zero crossing as observed on the oscilloscope coincides with the zero crossing detection flag that the controller outputs (section 5.5.3). While the term $\frac{1}{\omega t_{av}}\left[1+\cos\left(\pi-\frac{t_{av}}{T/2}\pi\right)\right]$ in eq. (5.18) equals 0.0940, a factor of 0.10 determines the zero crossing of the line voltage with a higher precision. Using this factor, the uncertainty in detecting the zero crossing during the rising slope of the line voltage is $(-110 \ldots -40)$ µs with respect to the actual zero crossing, and during the falling slope, it is $(-20 \ldots 40)$ µs.

5.4.6 Computation of the peak output voltage ripple

To determine the peak inductor current that is required for the estimation of L and R_L, one must find the amplitude of the 120 Hz component of the output voltage, i.e. the peak of the output voltage ripple $\hat{V}_{o(2)}$ (eq. (3.19)). Though switching noise is well filtered by the output capacitors (please see section 5.1), it turned out that the

acquired output voltage signal still exhibits fluctuations. They are attributed to errors of the analog-to-digital conversion which do not affect the output voltage control because of their relatively low magnitude, but which become significant when looking at the much smaller output voltage ripple. In an attempt to obtain $\hat{V}_{o(2)}$ as precisely as possible despite the perturbations in the sensed v_o, $\hat{V}_{o(2)}$ can be computed from the integral of $v_o - V_o$ over one quarter line cycle.

The output voltage ripple has a sinusoidal waveform with a frequency of 120 Hz $= 2\omega$ as depicted in Figure 5.5. The describing equation is $v_o(t) - V_o = -\hat{V}_{o(2)} \sin 2\omega t$. Because $\int_0^{T/4} \sin 2\omega t \, dt = \frac{1}{2\omega}[\cos 0 - \cos \pi] = \frac{1}{\omega}$,

$$\int_0^{T/4} (v_o - V_o)\, dt = -\frac{1}{\omega}\, \hat{V}_{o(2)} \tag{5.19}$$

and

$$\int_{T/4}^{T/2} (v_o - V_o)\, dt = \frac{1}{\omega}\, \hat{V}_{o(2)} \tag{5.20}$$

Subtracting eq. (5.19) from eq. (5.20) allows to calculate $\hat{V}_{o(2)}$:

$$\hat{V}_{o(2)} = \frac{\omega}{2}\left[\int_{T/4}^{T/2} (v_o - V_o)\, dt - \int_0^{T/4} (v_o - V_o)\, dt\right] \tag{5.21}$$

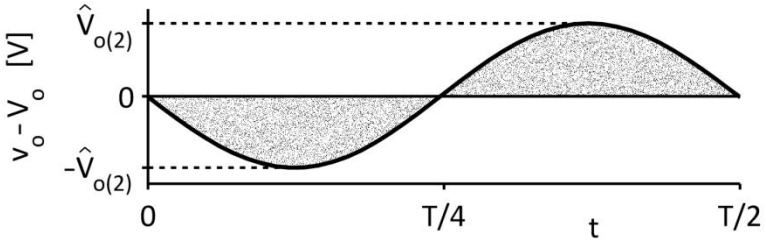


Figure 5.5: Output voltage ripple

Equation (5.21) can as well be used with V_o^* instead of V_o. This way, it is not necessary to find V_o. If, because of a small regulation error, the actual output voltage

differs slightly from its reference, both integrals in eq. (5.21) are equally affected and the subtraction eliminates the impact on the result.

5.5 Interfaces of the digital controller

Figure 5.6 shows the hardware interfaces of the digital controller:

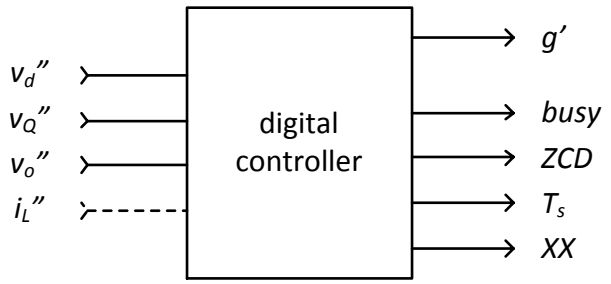


Figure 5.6: Hardware interfaces of the digital controller

The signals that are fed to the digital controller are the output voltages of the voltage dividers connected to the rectified line voltage v_d, the transistor voltage v_Q and the output voltage v_o. When the circuit operates in the test mode where the inductor current is sensed, the voltage across the sense resistor is acquired as well. All these are analog signals that are digitized by the analog to digital converter after being conditioned as explained in subsection 5.5.1.

What the digital controller outputs is most importantly the gating signal g'. Before being passed to the MOSFET gate, it still requires amplification; see subsection 5.5.2. Secondly, some test signals are given to binary outputs that are available on the DSP board as detailed in subsection 5.5.3.

5.5.1 Analog inputs

It is recommended in [77] that buffer circuits are used to drive the analog inputs. Such a buffer consists of an operational amplifier that is in this case configured as a

voltage follower. The selection of a suitable operational amplifier was made with the following criteria in mind:

– slew rate.

  The controller samples the drain-source voltage of the MOSFET which is a discontinuous waveform. At turn-on, it rapidly falls from the output voltage to a value close to zero. The operational amplifier must be able to track the full swing in the corresponding $v_Q{}'$ within at most one switching period. Because $v_Q{}'$ can be at the convertible maximum of 3 V, the necessary absolute minimum slew rate is

$$\mathrm{SR} > \frac{3\ \mathrm{V}}{T_s} = \frac{3\ \mathrm{V}}{1.25\ \mu\mathrm{s}} = 2.4\ \frac{\mathrm{V}}{\mu\mathrm{s}} \tag{5.22}$$

– gain-bandwidth product.

  The drain-source voltage has a fundamental frequency of 20 kHz but is very rich in harmonics as is characteristic for waveforms with high slopes. With unity gain, a gain-bandwidth product of not less than 1 MHz is desirable.

– package type.

  The operational amplifier should come in a package suitable for through-hole mounting. Because up to four signals are acquired, a package containing four amplifiers would be ideal.

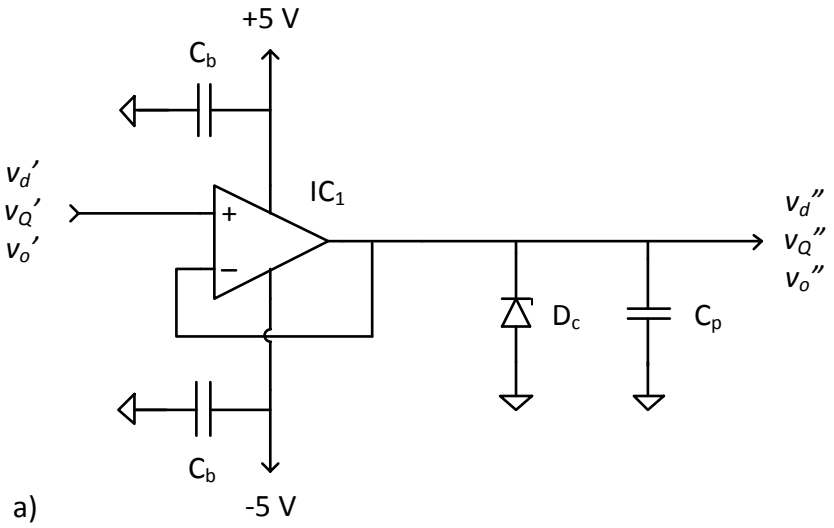
An integrated operational amplifier that meets these requirements is MC33079PG from ON Semiconductor [78]. The circuit diagram of the buffer circuits for the signals $v_d{}'$, $v_Q{}'$ and $v_o{}'$ is depicted in Figure 5.7a. Because the voltage across the inductor current sense resistor is negative, the operational amplifier is set up as an inverting amplifier, see Figure 5.7b. If the maximum ADC input voltage is desired to correspond to an inductor current of roughly 3 A, the gain has to be $\frac{3\ \mathrm{V}}{3\ \mathrm{A}\cdot 0.5\ \Omega} = 2 = -\frac{R_{11}}{R_{10}}$.

Because input voltages in excess of 3 V may interfere with the proper operation of the ADC, reverse biased Zener diodes are connected across the inputs. It was

experimentally found that Zener diodes with a breakdown voltage of 3.9 V limit the amplifier output voltage to 3.4 V.

The capacitor C_p serves to stabilize the signal during sampling [77] and to filter high frequency noise. It is connected on the DSP board, as close as possible to the ADC pins.

All components in the signal conditioning circuits are listed in Table 5.4.



a)

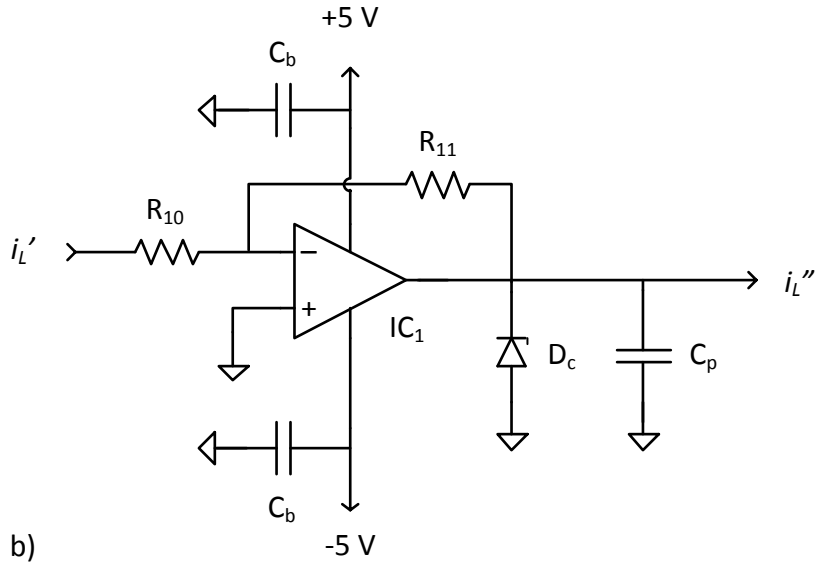


Figure 5.7: Signal conditioning circuits for v_d', v_Q' and v_o' (a) and for i_L' (b)

Table 5.4: Circuit components for the signal conditioning circuits

integrated circuit IC_1	operational amplifier, quad version ON Semiconductor, MC33079PG supply voltage ± 5 V
resistors R_{10}, R_{11}	$R_{10} = 2.4$ kΩ $R_{11} = 4.3$ kΩ
Zener diodes D_z	STMicroelectronics, 1N4730A $V_z = 3.9$ V, $P = 1$ W
capacitor C_p	ceramic capacitor TDK Corporation, FK28C0G1H220J $C_p = 22$ pF
bypass capacitor C_b	ceramic capacitor $C_b = 1$ µF

An auxiliary linear power supply provides the supply voltages of ± 5 V to the operational amplifier integrated circuit. The power supply is isolated from the utility supply and accommodated on a separate circuit board.

A calibration of the voltage dividers and analog-to-digital converters was carried out such that the voltages and the current can be computed from the conversion results accurately despite tolerances in the resistors and inaccuracies of the ADCs. The procedure consisted of finding the relationships between the conversion result register values and the quantities to be measured, expressed through a linear function. Refer to Appendix C for the details.

5.5.2 Gate drive circuit

The digital outputs of the signal controller provide too little voltage and current to be able to drive the power MOSFET [66]. A gate driver with a separate power supply

is required to interface the digital output of the controller with the gate of the transistor. The most important selection parameter is the driver's average output current. To get an estimate, the relation $I_G = Q_G f_{sw}$ was used. The voltage that is employed to drive the transistor is 15 V, at which the gate charge Q_G is found from the MOSFET data sheet [79] to be $Q_G = 4.7$ nC. Then $I_G = 4.7\text{ nC} \cdot 20\text{ kHz} = 94\ \mu\text{A}$. A suitable MOSFET gate driver is the MCP1407 from Microchip Technology [80]. It provides a continuous output current of 1.3 A and comes in a space-saving 5-pin TO-220 package.

In the MOSFET data sheet, gate resistances of 9.1 Ω and 25 Ω were used to specify the transient characteristics. A value that lies in this range, 20 Ω, was chosen for R_G. A resistor with a low power rating of ⅛ W was selected. It makes the resistor act as an overcurrent protection device (protecting the gate driver) if the MOSFET experiences a failure that causes a short circuit between all terminals.

A 100 kΩ resistor was connected between gate and source to prevent charge from accumulating on the gate (R_{GS}). As a further means of protection, a Zener diode D_{GS} with a breakdown voltage of 24 V prevents excess gate-source voltages (max. $V_{GS} = 30$ V).

Figure 5.8 shows the circuit diagram of the gate drive circuit. The circuit components are listed in Table 5.5.

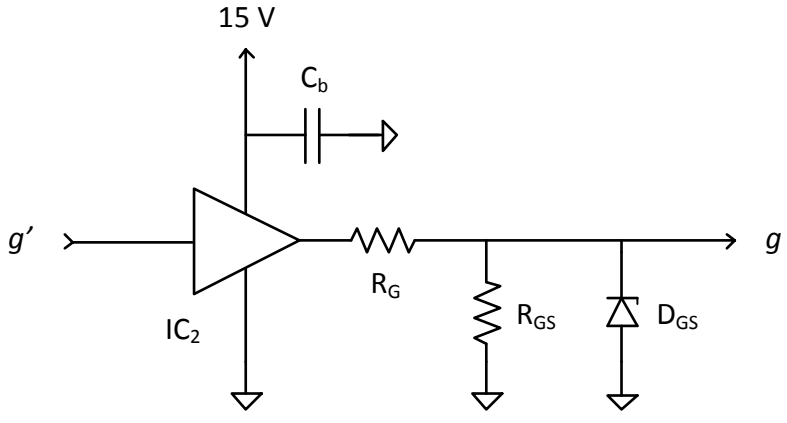


Figure 5.8: Gate driver circuit

Table 5.5: Circuit components for the gate driver circuit

integrated circuit IC_2	MOSFET gate driver Microchip Technology, MCP1407 supply voltage 15 V
gate resistor R_G	$R_G = 20\ \Omega$, rated power ⅛ W
gate-source resistor R_{GS}	$R_{GS} = 100\ \text{k}\Omega$, rated power ¼ W
gate-source Zener diode D_{GS}	Fairchild Semiconductor, BZX85C24 $V_z = 24$ V, $P = 1$ W
bypass capacitor C_b	ceramic capacitor $C_b = 1\ \mu\text{F}$

The gate driver supply voltage is provided by the same auxiliary power supply that also powers the operational amplifiers.

5.5.3 Binary outputs

Some of the binary outputs of the controller were programmed to output useful test signals. A description is given in Table 5.6:

Table 5.6: Controller output signals

pin	signal	use
3	GPIO F4	zero crossing detection (ZCD) A short pulse indicates that the software has detected the line voltage zero crossing.
11	GPIO A2	shutdown (XX) The operation of the converter was terminated because the output voltage or the inductor current was too high.
15	T1PWM	sampling start (T_s) Its rising edge marks the start of the analog to digital conversion sequence.
16	T2PWM	gating signal g'
35	GPIO B5	busy flag ($busy$) indicates that the computation following the acquisition of 40 samples is taking place

All output signals are high-active. Ground pins are 19, 20, 39 and 40.

6 Experimental results

Chapter 6 presents the experimental validation of the proposed scheme. The control scheme that was developed in chapter 3 and investigated through simulation in chapter 4 is now tested with the converter from chapter 5. Results are shown that demonstrate the functioning of the current and voltage control loops, for the inductor model identification and for its adaptation.

6.1 Test setup

Care was taken that the converter circuit is galvanically isolated from the utility supply. This was done to minimize the danger of receiving an electric shock when accidentally touching a bare part of the prototype. An isolation transformer (115 V/115 V, 500 VA) was used to connect the circuit to the utility grid. The controller board power supply provides galvanic isolation, as does the auxiliary power supply for the operational amplifiers and the gate driver. To prevent grounding via the oscilloscope, only differential voltage probes were used to measure voltages.

In addition to the isolation transformer, a variable autotransformer (115 V/0-140 V, 700 VA) was used to connect the converter to the utility outlet. This way, the input voltage could be varied, adjusted to 120 V, and applied gradually upon startup as a safety measure. Figure 6.1 illustrates the arrangement:

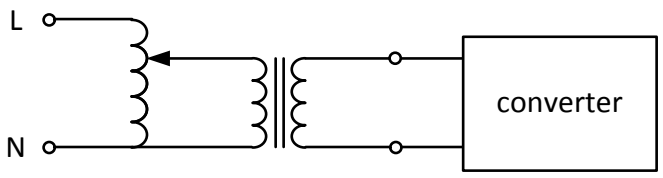


Figure 6.1: Connection of the converter to the utility grid

The load resistance comprised of several variable resistance modules in which parallel-connected discrete resistors could be switched on and off.

6.2 Measurement equipment

The following measurement equipment was used for testing the converter:

- oscilloscope: digital four-channel oscilloscope Tektronix TDS 224,
- voltage probes: high voltage differential probes Tektronix P5200, bandwidth: 100 kHz,
- current probe: Tektronix A622, bandwidth: 100 kHz,
- multimeter to measure V_{in}: digital multimeter Philips PM 2519,
- multimeter to measure V_o: digital multimeter Fluke 8010A,
- multimeter to measure I_o: digital multimeter Fluke 8050A,
- data acquisition module to store measurements in the computer: National Instruments USB-6259 BNC, 16 16-bit analog inputs, max. 1,250,000 samples per s.

6.3 Test procedure

First, the converter's steady-state operation was tested with the inductor current being acquired from the sensing resistor R_C to see whether the inductor current and output voltage regulation work properly. Then the current control capability was investigated when the inductor current was computed with the non-adaptive inductor model. In the next step, waveforms at load transients were captured. Thereafter, the model identification was tested and eventually, the model adaptation was turned on.

6.4 Steady-state operation

When the inductor current is measured via the sensing resistor, the resulting input current waveforms for rated load and half the rated load are as shown in Figure 6.2:

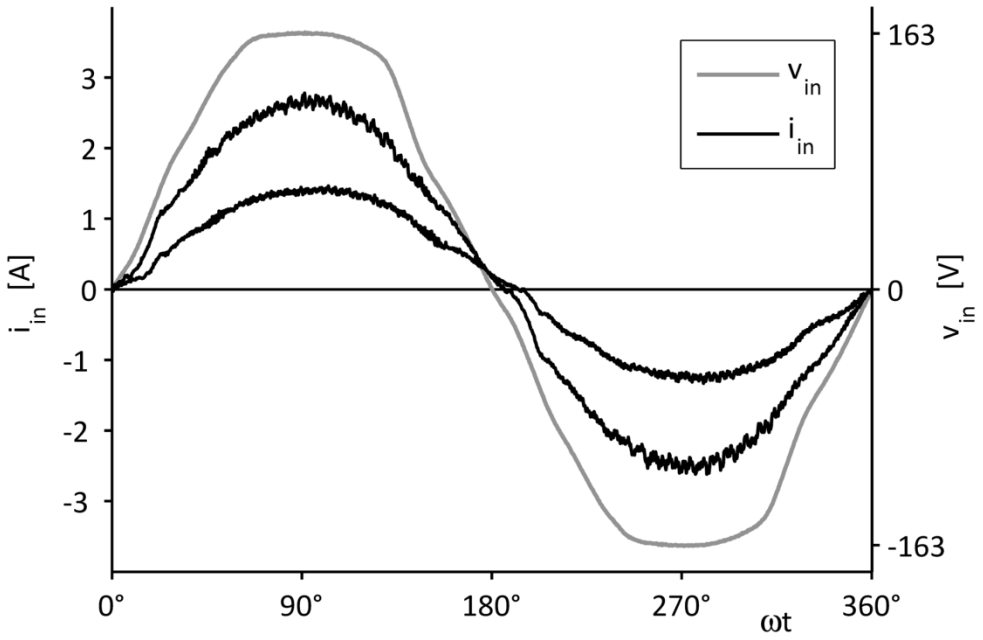


Figure 6.2: Measured input voltage and input current waveforms at rated load and half load with the inductor current being measured

Very remarkable is the strong distortion of the line voltage. Firstly, it is rather flat where the peaks should occur. With the RMS value being regulated to 120 V, the peak value should be 170 V, not only 163 V as were measured. Secondly, the upward and downward slopes do not exhibit the natural curvature of a sine wave. The input current is uninfluenced by the line voltage distortion. The reason is that the reference current is computed from a sine look-up table and only synchronized with the line voltage in time at the zero crossings.

It was noted that after turn-on, the output voltage is usually a few volts larger than 380 V, e.g. 4…6 V. Within a few minutes of operation, the output voltage drops by a few volts and then stabilizes slightly below 380 V, e.g. at 378 V. Such behaviour

might be attributable to the resistors of the voltage divider R_5, R_6, P_3 heating up. Other possible causes are the signal conditioning circuit and the analog-to-digital converter.

6.5 Computation of the inductor current

From the results presented so far, it can be concluded that the inductor current and output voltage control function satisfactorily during steady-state operation. Next, the quality of the computed inductor current was investigated. The computed current was captured from the digital controller along with the measured current for one line half-cycle while the measured current was still used to compute the control error. Ideally, both should be identical. When L^{imp} and R_L^{imp} have their nominal values of 8 mH and 0.6 Ω, a noticeable discrepancy in amplitude occurs. As it already appeared in section 6.4, the true inductance is greater than 8 mH. Different values were tried to make the computed current as close as possible to the measured current. Two examples of good agreement are depicted in Figure 6.3:

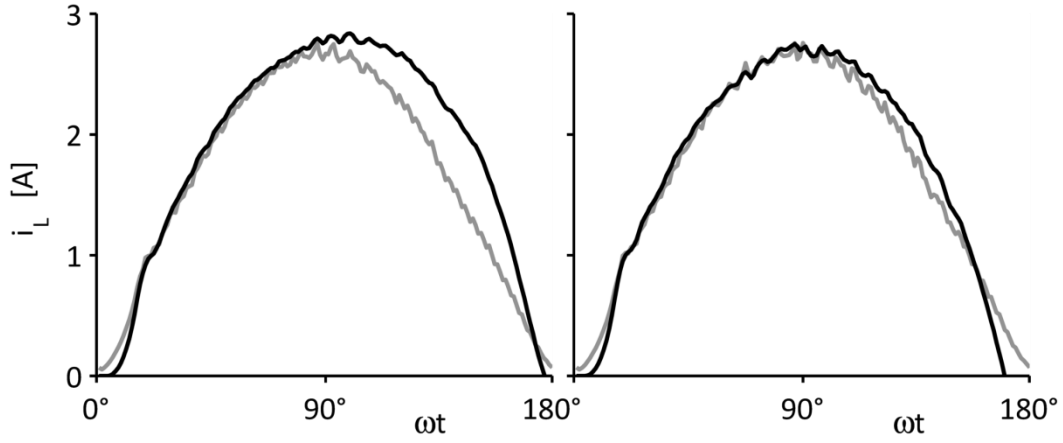


Figure 6.3: Measured (grey) and computed (black) inductor current when the measured current is used for the current control, inductor 1, rated load ($P_o = 202$ W), left: $L^{imp} = 11$ mH, $R_L^{imp} = 0.6\ \Omega$, right: $L^{imp} = 10.5$ mH, $R_L^{imp} = 1\ \Omega$

In the left set of waveforms in Figure 6.3, the computed current matches the measured current well in the first half of the line half-cycle. There is a distinct difference during the second half in which the computed current does not decrease quickly enough first, while the two converge again towards the end of the line half-cycle. The two curves in the right diagram show a better agreement overall, but the computed current drops too far at the end of the line half-cycle, reaching negative values.

When the computed current is now used by the current controller to compute the control error and the measured current is only recorded for comparison, the situations with the same sets of inductor model parameters as above are as follows:

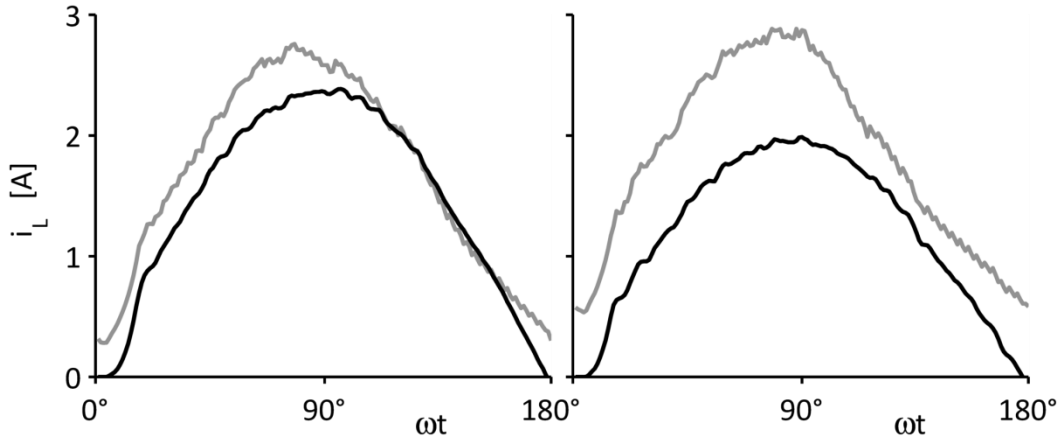


Figure 6.4: Measured (grey) and computed (black) inductor current when the computed current is used for the current control, inductor 1, rated load ($P_o = 202$ W), left: $L^{imp} = 11$ mH, $R_L^{imp} = 0.6\ \Omega$, right: $L^{imp} = 10.5$ mH, $R_L^{imp} = 1\ \Omega$

The shape of the measured inductor current is reasonably close to sinusoidal with both inductor models during the first half of the line half-cycle. In the second half, it first decreases too quickly, but then flattens out towards the end and does not reach zero at the end of the line half-cycle. This is more the case in the right plot than in the left. The fact that the inductor current does not reach zero at the end of the line half-cycle manifests itself in the input current jumping from a positive value to the corresponding negative value when the input voltage changes its sign.

Meanwhile, the computed current is overall very close in shape to a sine wave. This indicates that the current controller still performs well. The controller—not knowing the true inductor current—successfully regulates the computed current to follow its sinusoidal reference.

The waveshapes in Figure 6.4 can also be explained from the ones in Figure 6.3. When, for example, the computed inductor current decreases too slowly after reaching its peak, then the current controller adjusts the duty cycle such that the computed current falls more rapidly. But if the true inductor current previously had the right shape, then it will now drop too quickly, which can be observed in Figure 6.4.

Figure 6.5 shows the input current when the computed inductor current is used for the current control:

Figure 6.5: Measured input current waveform at rated load ($P_o = 202$ W) with inductor 1 when the computed current is used for the current control; $L^{imp} = 11$ mH, $R_L^{imp} = 0.6\ \Omega$

Striking are the two phases in the second half of each line half-cycle. The input current first approaches zero too quickly during the period θ_1, then changes its slope rather abruptly and flattens out during θ_2. Such behaviour can only occur if the inductor is nonlinear, that is, if the core has a nonlinear relationship between

magnetic field strength H and magnetic flux density B. All ferromagnetic materials are naturally nonlinear in that they exhibit saturation and hysteresis, effects that make the permeability dependent on the field strength and on the previous excitation. Consequently, the core behaves differently for small and large currents and for rising and falling currents. Figure 6.6 displays the hysteresis curve of the used core material:

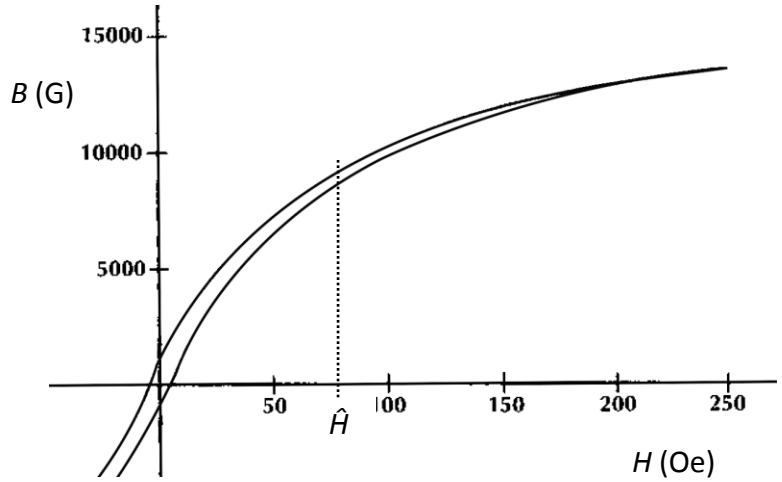


Figure 6.6: Hysteresis curve of the core material -40, used for inductor 1 [49]

The peak current in Figure 6.5 is 2.9 A. At this current, the mean magnetic field strength in the core, computed using the core's mean magnetic path length l_m, is

$$\widehat{H} = \frac{N \cdot \hat{I}_L}{l_m} = \frac{236 \cdot 2.9\ \text{A}}{11.2\ \text{cm}} = 6111\ \frac{\text{A}}{\text{m}} = 77\ \text{Oe} \tag{6.1}$$

From the $B(H)$ curve in Figure 6.6 it becomes evident that the core has a different characteristic at this field strength than at lower excitations. The permeability at $\widehat{H}$ is only 47% of the initial permeability occurring at small currents [49]. Hysteresis curves are determined with currents of alternating signs that excite the material in both orientations. In the present application, however, the inductor only experiences positive currents. As a result, even with the $B(H)$ curve known from the data sheet, it is difficult to predict how the inductance varies as a function of the current.

The linear inductor model that the control scheme uses takes neither saturation nor hysteresis into account. For this reason, the computed inductor current can be more accurate during some intervals of the line half-cycle, and less at others.

To circumvent this problem, the hitherto used inductor (henceforth referred to as inductor 1) was replaced with one that was designed to have a characteristic as linear as possible (inductor 2). The important difference between them is the core material. Comparing the hysteresis curve of material -2 (Figure 6.7) with the one of material -40 (Figure 6.6), one can see that the relationship B versus H is much more linear for the material -2. The linearity in the characteristic comes at the expense of a lower permeability. To achieve the same nominal inductance, a larger core and more turns are required. Details on both inductors can be found in Appendix D.

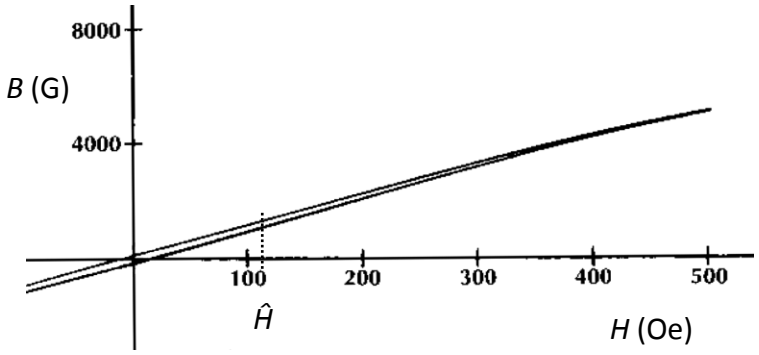


Figure 6.7: Hysteresis curve of the core material -2, used for inductor 2 [49]

With the new design, the field strength in the core is now

$$\widehat{H} = \frac{N \cdot \hat{I}_L}{l_m} = \frac{592 \cdot 2.9\ \text{A}}{19.8\ \text{cm}} = 8671\ \frac{\text{A}}{\text{m}} = 109\ \text{Oe} \tag{6.2}$$

Over this range, the permeability only drops by 2%.

Contrasting again the measured and the computed inductor current, it can now be observed from Figure 6.8 that both are practically identical:

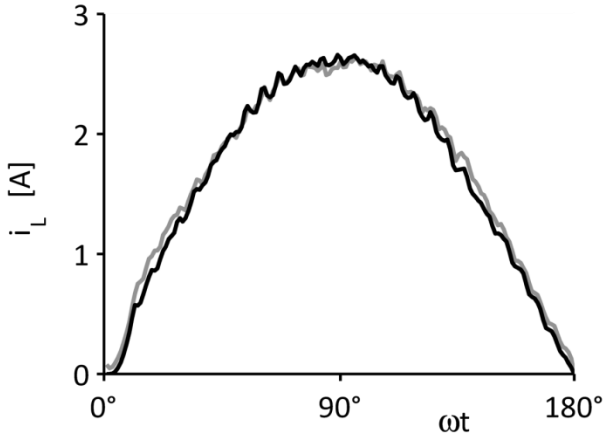


Figure 6.8: Measured (grey) and computed (black) inductor current when the computed current is used for the current control; inductor 2; rated load ($P_o = 204$ W), $L^{imp} = 6.5$ mH, $R_L^{imp} = 1\ \Omega$

The model parameters that produced this result were 6.5 mH and 1 Ω. Shown in Figure 6.9 is one cycle of the input current:

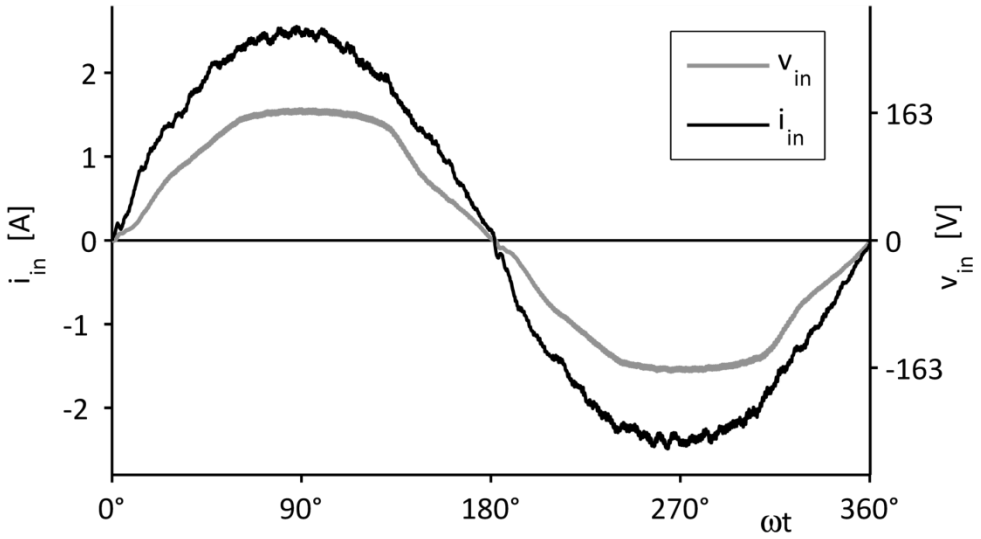


Figure 6.9: Measured input voltage and input current waveforms at rated load ($P_o = 204$ W) with inductor 2 when the computed current is used for the current control; $L^{imp} = 6.5$ mH, $R_L^{imp} = 1\ \Omega$

Its waveshape is now very close to sinusoidal and does not exhibit the distortion coming from inductor 1's nonlinearity, which is present in the waveform in Figure 6.5, any more. All further experiments were conducted with inductor 2.

The converter efficiency was determined at different loads and is listed in Table 6.1. The numbers do not take the power fed to the auxiliary power supply and the power for the digital controller board into account.

Table 6.1: Converter efficiency at various loads

P_o	198 W	150 W	100 W	50 W
η	0.94	0.92	0.91	0.86

6.6 Transients

The transient response of the voltage control loop was tested with a step change in the load resistance, analogously to the simulation (section 4.3). A load change was achieved by short-circuiting some of the series-connected resistance modules, or removing the short circuit, respectively. Output powers of 202 W and 93 W resulted from this operation. First, the experiment was done with the current controller using the current measured with R_c. The input current and output voltage waveforms are given in Figure 6.10 and Figure 6.11:

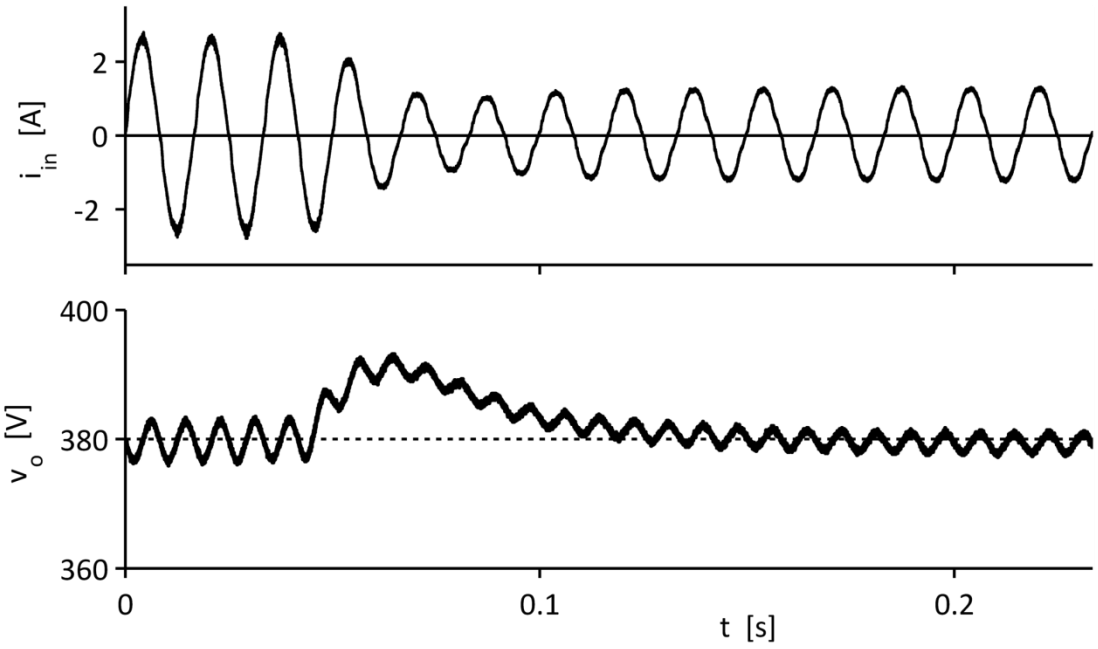


Figure 6.10: Measured input current and output voltage waveforms for a load step change from full load ($P_o = 202$ W) to half load ($P_o = 93$ W) when the measured current is used for the current control

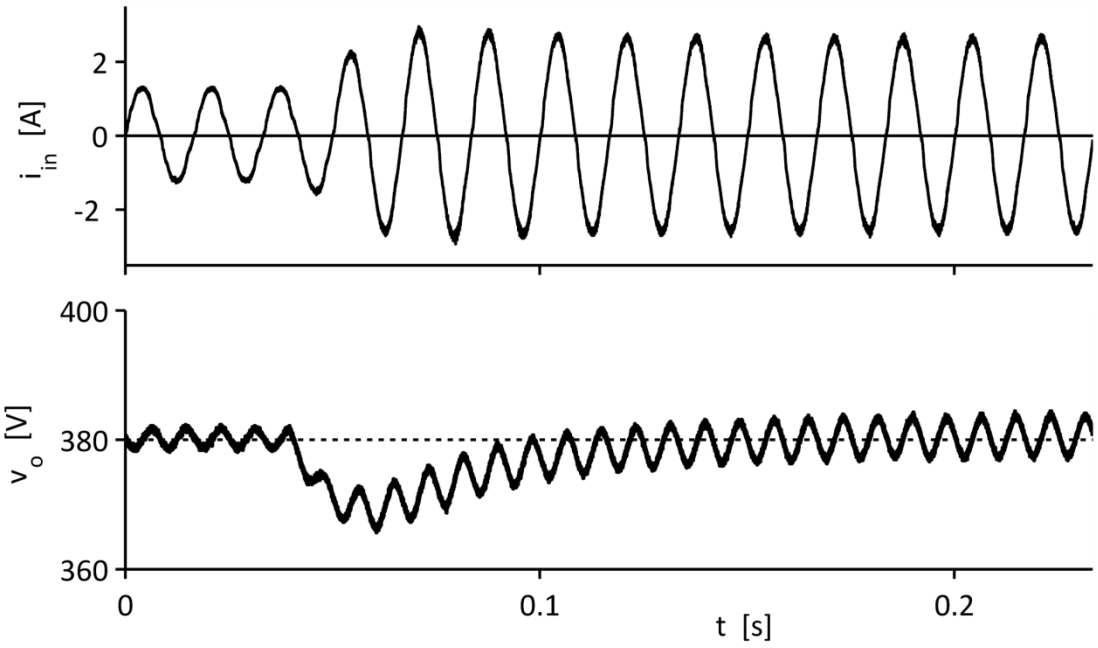


Figure 6.11: Measured input current and output voltage waveforms for a load step change from half load ($P_o = 93$ W) to full load ($P_o = 202$ W) when the measured current is used for the current control

The transient response of v_o is in very good agreement with the simulated results (Figure 4.6 and Figure 4.7). The input current retains a good waveshape during the transients.

When the current controller uses the computed current to calculate the control error, the waveforms in Figure 6.12 and Figure 6.13 result:

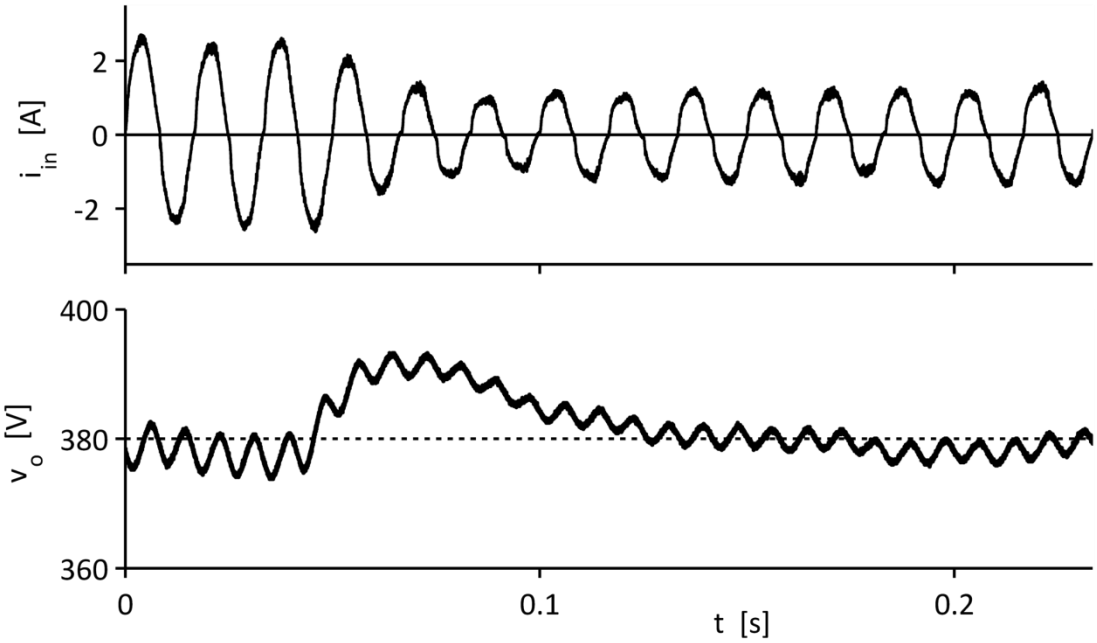


Figure 6.12: Measured input current and output voltage waveforms for a load step change from full load ($P_o = 202$ W) to half load ($P_o = 93$ W) when the computed current is used for the current control

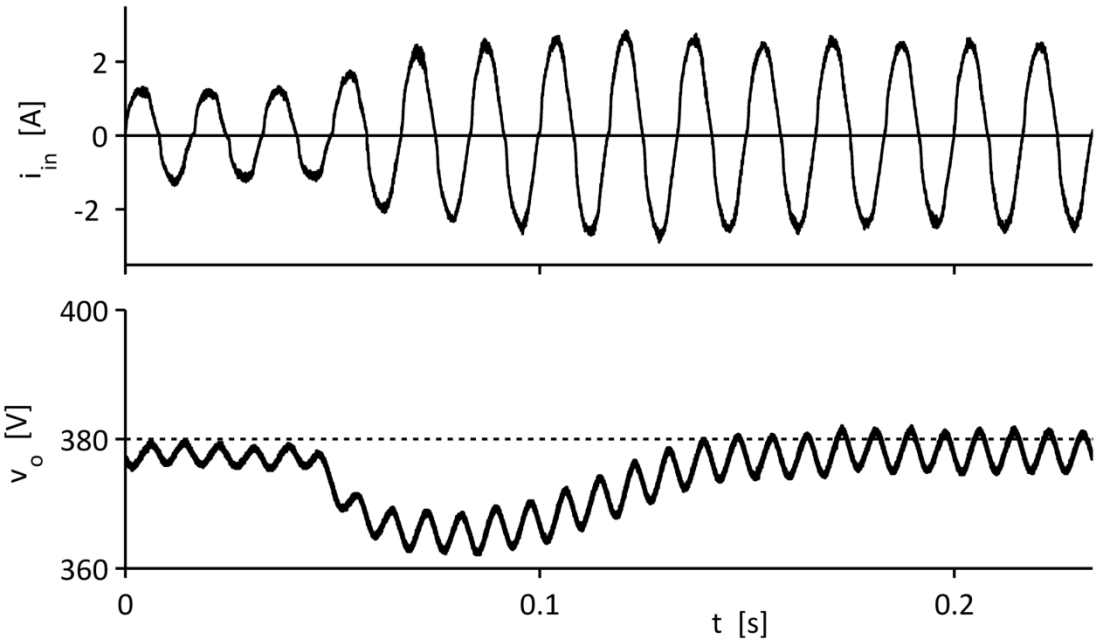


Figure 6.13: Measured input current and output voltage waveforms for a load step change from half load ($P_o = 93$ W) to full load ($P_o = 202$ W) when the computed current is used for the current control

In this case, too, the input current during the transients exhibit a low distortion. In Figure 6.13, the output voltage has a larger undershoot than in the corresponding Figure 6.11, and takes a slightly longer time to reach its steady-state value again. Discrepancies between the measured output voltage and 380 V in steady state are due to the drifts associated with temperature changes as mentioned in section 6.4.

A close comparison of several input current cycles in Figure 6.12 and Figure 6.13 also reveals small variations in the shape of individual cycles. These stem from noise that is inevitably present in the sensed voltages and impact the current computation.

The transients that occur in the input current and output voltage after turn-on are plotted in Figure 6.14. After the switch S is closed and before the switching starts, the output capacitors $C_1 \ldots C_3$ are charged to $\hat{V}_d$, here 163 V, through the diode D_b. Through the voltage control, the output voltage then gradually picks up and settles at 380 V. While the output voltage rises, the input current is strongly distorted. It assumes its normal waveshape once the output voltage has settled and steady state

is reached at $t \approx 0.12$ s. The turn-on transient lasts roughly 100 ms, very similar to the simulated case depicted in Figure 3.12.

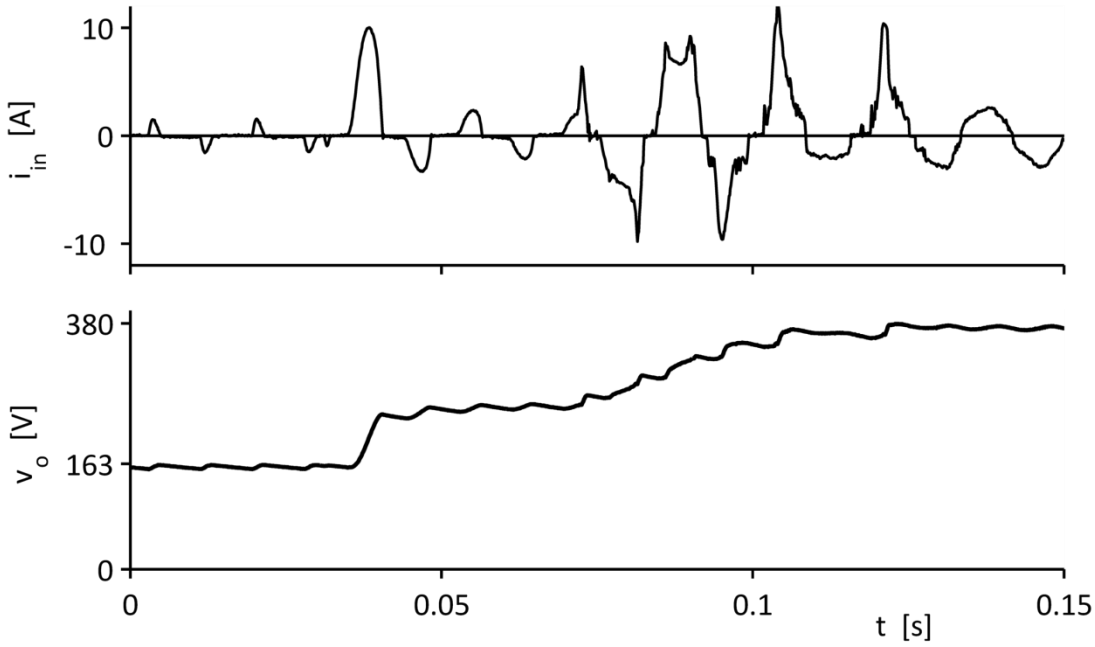


Figure 6.14: Measured input current and output voltage during turn-on

6.7 Inductor model identification

Before the inductor model was made adaptive, i.e. the feedback paths for L^{est} and R_L^{est} were closed, the estimated model parameters were inspected. To do this, the values from 200 line half-cycles were stored in the controller and then retrieved from the memory. These values are plotted in Figure 6.15 for rated load when the computed inductor current was used for the current control:

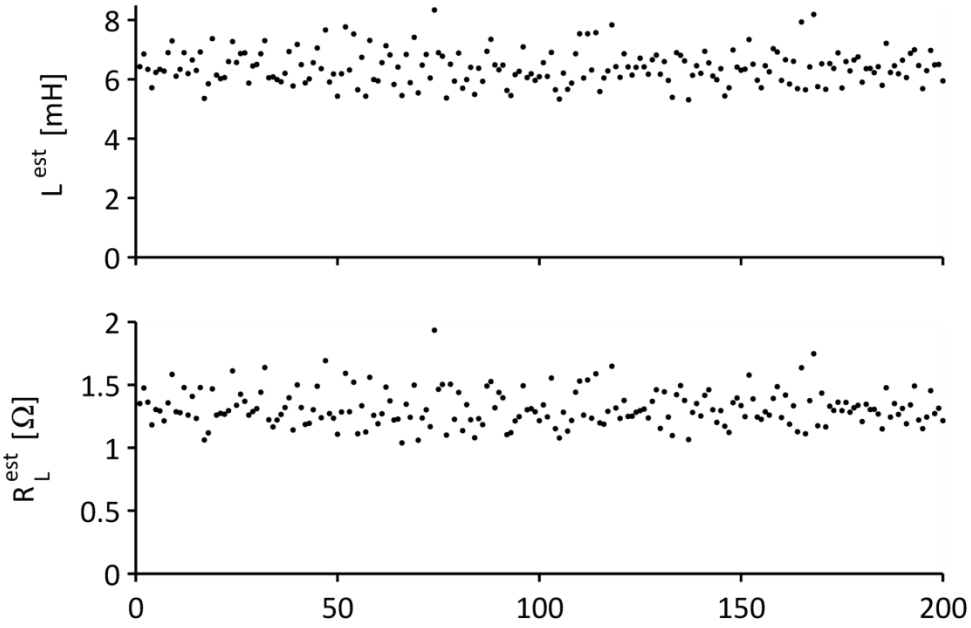


Figure 6.15: L^{est} and R_L^{est} for 200 line half-cycles without model adaptation, control with computed current, rated load ($P_o = 201$ W)

Evidently, the individual values scatter around a mean value. The mean values for the data in Figure 6.15 are $L_{av}^{est} = 6.4$ mH and $R_{L,\,av}^{est} = 1.32\ \Omega$, and the respective standard deviations are $\sigma_L = 0.58$ mH and $\sigma_{R_L} = 0.15\ \Omega$. In section 6.5, it was already found that the computed current is close to the measured current when $L^{imp} = 6.5$ mH. As expected, this value for the equivalent series resistance is slightly larger than the DC resistance of the copper wire of 1.2 Ω.

Potential sources of error in the estimation are the variables from which L^{est} and R_L^{est} are computed, that is the peak input voltage, the integral over the inductor voltage, and the peak inductor current. The peak input voltage is not very prone to large fluctuations from one line half-cycle to the next. As for `v_LsumLQC` and `v_Lsum`, their standard deviations for 200 line half-cycles relative to their mean values were only 3% and 5% respectively. The values that were computed for $\hat{V}_{o(2)}$, on the other hand, have a standard deviation of 16%, making this quantity the main contributor to the fluctuations in L^{est} and R_L^{est}. Its noise carries over into $\hat{I}_L$ via eq. (3.19), which causes the errors in L^{est} and R_L^{est}. At this point, the low-pass filter in the model

adaptation algorithm mentioned in section 3.5 comes into play and averages the noise out. Further, because in the derivation of eq. (3.19) the assumption was made that the converter is lossless, the true peak inductor current must be larger than predicted by (3.19) to account for the losses in the converter. This will affect both estimated inductor model parameters in the same way, but will not cause problems in the current control, as was explained in section 3.4.

The data discussed so far was obtained at rated output power. Results for lighter loads can be found in Table 6.2. The model parameters were estimated both with the measured and with the computed inductor current. Because computing i_L already introduces inaccuracies into the control itself, the standard deviations of L^{est} and R_L^{est} are usually higher. Meanwhile, estimating L^{est} and R_L^{est} while using the measured inductor current for the current control has no practical significance and was only done to gather data for comparison.

Table 6.2: Averages and standard deviations of L^{est} and R_L^{est} for 200 line half-cycles without model adaptation, control with measured (R_c) and computed current, at various loads

P_o	i_L	L_{av}^{est} [mH]	$\frac{\sigma_L}{L_{av}^{est}}$	$R_{L,\,av}^{est}$ [Ω]	$\frac{\sigma_{R_L}}{R_{L,\,av}^{est}}$
201 W	measured	6.45	5.1%	1.37	7.1%
	computed	6.40	9.0%	1.32	10.9%
151 W	measured	6.27	4.8%	1.62	6.3%
	computed	6.01	14.9%	1.37	15.9%
101 W	measured	5.62	5.7%	1.93	6.6%
	computed	5.68	12.8%	1.48	13.2%
51 W	measured	4.58	6.5%	1.65	7.0%
	computed	4.90	19.3%	1.21	15.5%

Overall, the estimated model parameters show a good consistency down to low output powers. The estimated inductance becomes smaller as the output power, and with it the inductor current, decreases. As for the inductor ESR, temperature changes account for some variations in the estimated values. The wire temperature

could not be kept constant in the individual runs. To conclude, the estimation of the inductor model parameters works satisfactorily and yields reproducible results.

6.8 Adaptation of the inductor model parameters

The experiments in the previous section revealed that the model identification returns usable outcomes. In the next step, these results were fed to the inductor model using the parameter adaptation algorithm described in section 3.5.

Figure 6.16 and Figure 6.17 illustrate the adaptation for some exemplary conditions. The plots show L^{imp} and R_L^{imp} with respect to time for different initial values. After the adaptation is turned on at $t = 0$, the implemented parameters asymptotically approach the estimated values. These results are close to the ones obtained through simulation and plotted in Figure 4.12 and Figure 4.13.

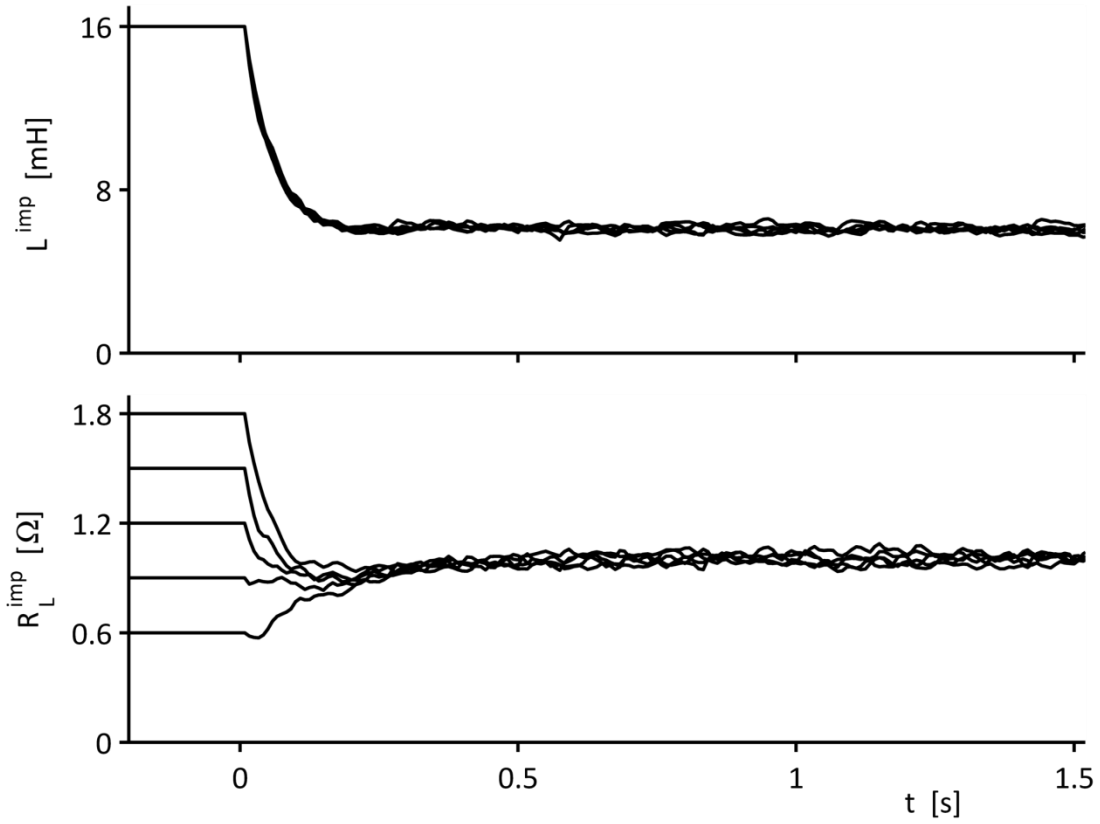


Figure 6.16: Inductor model parameter adaptation for $L^0 = 16$ mH and various values of R_L^0

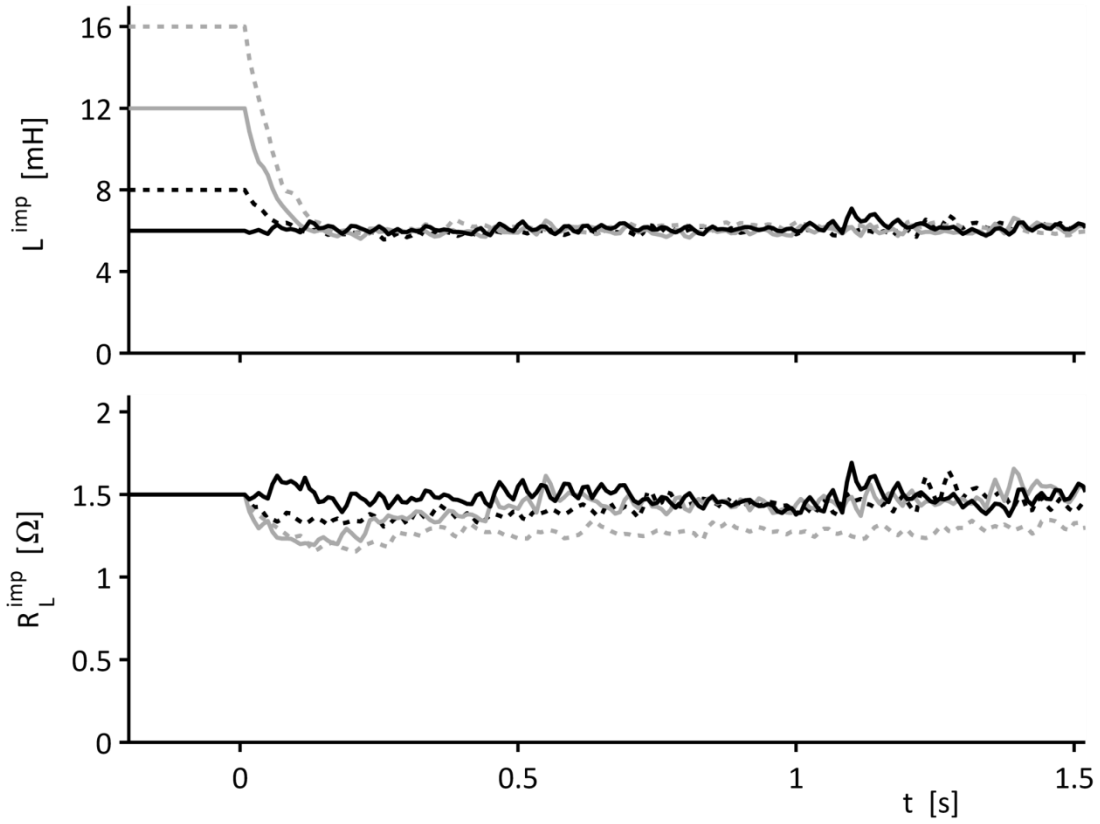


Figure 6.17: Inductor model parameter adaptation for $R_L^0 = 1.5\ \Omega$ and various values of L^0

When L^0 was set to values less than 6 mH, the current control became unstable and large oscillations occurred in the input current. In all other tested cases, the model adaptation converges to 6.1 mH and 1.3 … 1.5 Ω. A possible workaround to avoid the instability is to implement an L^0 larger than the nominal value such that this situation is avoided. The control is more robust when L^0 has a larger value.

Figure 6.18 shows the effect of the model adaptation on the input current for two exemplary cases of initial parameters. Plotted are the input current waveforms before and after the inductor model is adapted.

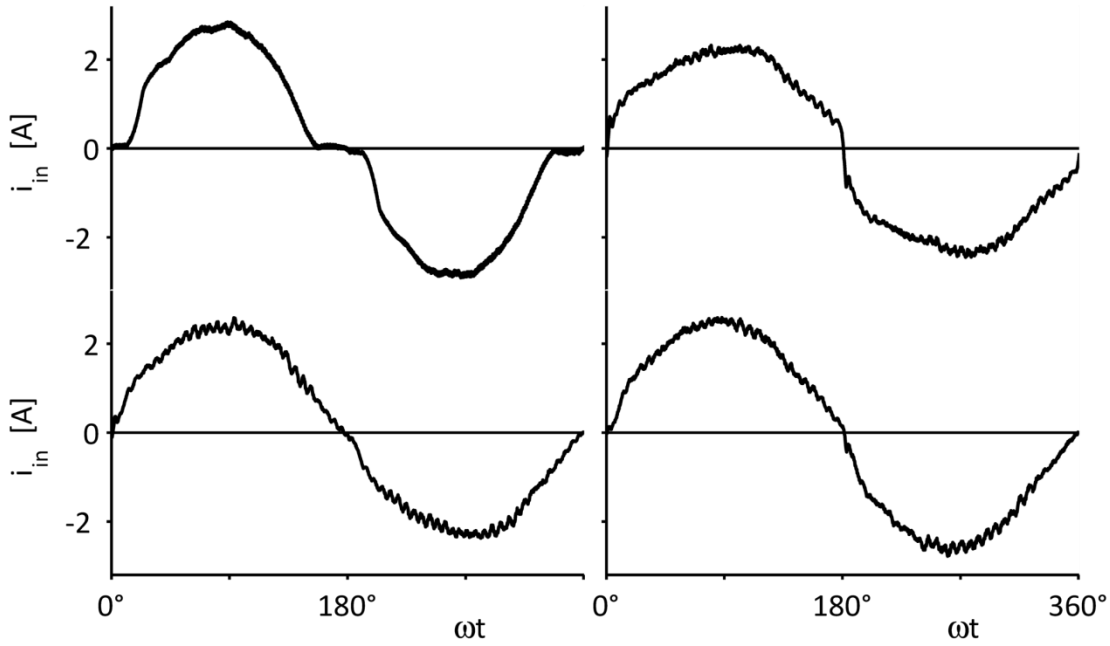


Figure 6.18: Input current before (top) and after (bottom) model adaptation at rated load ($P_o = 202$ W); left: $L^0 = 16$ mH, $R_L^0 = 0.9\ \Omega$, right: $L^0 = 6$ mH, $R_L^0 = 1.2\ \Omega$

Adapting the model with the estimated inductance and ESR drastically reduces the distortion of the input current. The total harmonic distortions of the waveforms in Figure 6.18 are listed in Table 6.3. They were computed with the FFT Analysis Tool from the waveforms sampled for 1 s at 80 kHz. For comparison, the table also contains the values for the corresponding simulated current from Figure 4.2. These are the minimum values that can theoretically be achieved with the present controller designs and ideal circuit elements and model adaptation. Also included are the values that were obtained when the measured inductor current was used by the current controller (plot Figure 6.2).

Table 6.3: Low-order harmonics of the input current relative to the fundamental in ‰ and its total harmonic distortion before and after model adaptation at rated load

	h	2	3	5	7	9	11	THD
simulated		0	16	6	6	5	5	2%
control with measured current		0	20	18	4	6	14	3%
$L^0 = 16$ mH, $R_L^0 = 0.9\ \Omega$	before adaptation	1	133	93	45	15	24	17%
	after adaptation	4	20	28	17	6	12	4%
$L^0 = 6$ mH, $R_L^0 = 1.2\ \Omega$	before adaptation	3	122	85	44	17	21	15%
	after adaptation	0	38	31	23	11	9	6%

The spectra before and after the model adaptation for the cases above are shown in Figure 6.19:

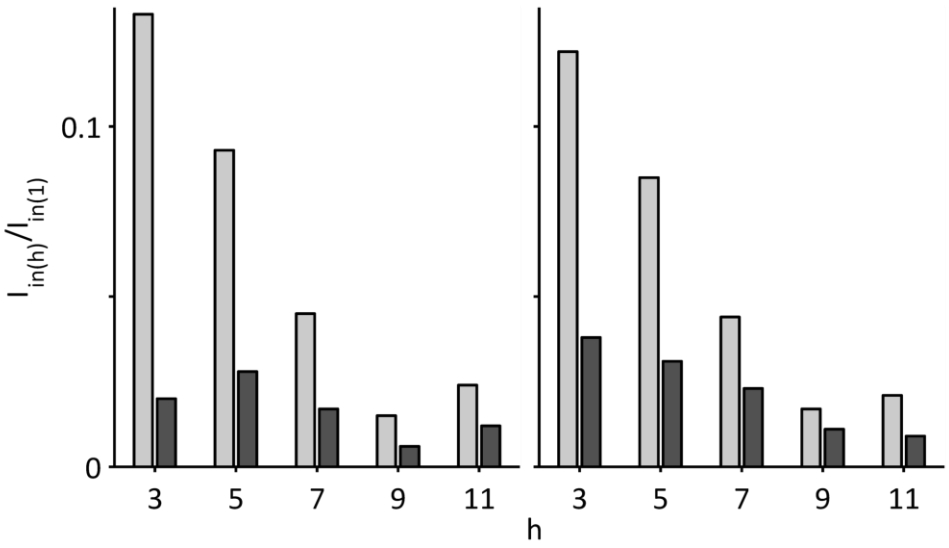


Figure 6.19: Low-order harmonics of the input current relative to the fundamental before (light) and after (dark) model adaptation at rated load;
left: $L^0 = 16$ mH, $R_L^0 = 0.9\ \Omega$, right: $L^0 = 6$ mH, $R_L^0 = 1.2\ \Omega$

Table 6.3 contains the values when the converter operates with its rated load. At smaller output powers, the harmonic content relative to the fundamental is higher, and so is therefore the total harmonic distortion (Table 6.4). The magnitudes of the input current harmonics, meanwhile, decrease with the output power.

Table 6.4: Low-order harmonics of the input current relative to the fundamental in ‰ and its total harmonic distortion after model adaptation at various loads

	h	2	3	5	7	9	11	THD
$L^0 = 16$ mH $R_L^0 = 0.9\ \Omega$	$P_o = 150$ W	1	21	25	15	6	10	4%
	$P_o = 100$ W	3	32	22	23	9	22	6%
$L^0 = 6$ mH $R_L^0 = 1.2\ \Omega$	$P_o = 150$ W	4	39	18	16	10	12	5%
	$P_o = 100$ W	9	57	28	17	13	15	7%

Another index that indicates the performance of the current shaping is the power factor. Equation (1.1) can also be expressed in terms of the total harmonic distortion as

$$\mathrm{PF} = \cos\varphi_{(1)}\ \frac{I_{in(1)}}{I_{in}} = \cos\varphi_{(1)}\ \frac{1}{\sqrt{1+\mathrm{THD}^2}} \tag{6.3}$$

It is evident from Figure 6.9 that there is negligible phase difference between the input voltage and current. Consequently, the displacement power factor $\cos\varphi_{(1)}$ is unity. A total harmonic distortion of 6%, for example, then leads to a power factor of $\frac{1}{\sqrt{1+0.06^2}} = 0.998$.

7 Conclusions

7.1 Summary of the present work

The thesis started with an explanation of the principle of power factor correction with the help of power electronic converters in chapter 1. After a thorough literature research and review of published methods of current sensorless control in chapter 2, it was concluded that no control scheme for boost-type switch-mode rectifiers has been published that adapts itself to uncertainties and changes in the inductor parameters. Such a concept that estimates and updates the model parameters during operation was developed in chapter 3 after the effects of inductor model mismatch were studied. Then controllers for the inductor current and output voltage were designed using linear control theory.

The simulation results presented in chapter 4 demonstrated the operation of the controllers and proved the feasibility of the parameter estimation and model adaptation. The model adaptation worked reliably over a broad range of parameter mismatches. In addition, the control scheme was tested on hardware. A prototype of the converter was built and a digital controller was programmed to undertake its control (chapter 5). With this experimental setup, it was shown in chapter 6 that the model parameter identification and adaptation perform similarly as predicted by the simulations.

The devised model identification proves to be a useful and valuable method to reduce the distortion of the input current. With the developed control scheme, the inductance and equivalent series resistance of the inductor are constantly updated during operation. This way, the controller can account for changes due to material tolerances, temperature variations, inductor aging etc.

7.2 Suggestions for future work

Using a large inductor with a low utilization, such as inductor 2, to get a close-to-linear characteristic is not economically feasible in mass produced converters. However, the input current resulting from a nonlinear inductor might be much less distorted if the controller uses a piecewise linear inductor model. Observing Figure 6.5, suitable intervals for obtaining separate inductances could be $0 \ldots 90°$, θ_1 and θ_2.

In the experiments, the converter became unstable when the implemented inductance was too small. This behaviour did not occur in the simulations and can therefore be attributed to hardware characteristics not modelled in the simulation schematic. However, it would be very beneficial to use a more comprehensive nonlinear model of the converter to do a thorough investigation of its stability and determine the stability boundaries.

Such a model could be extended to include the model identification scheme. It would then allow examining the model adaptation transients and devising a more sophisticated model adaptation algorithm. In this work, the chosen low-pass filter aimed to avoid interactions between the model adaptation from the output voltage control. Decoupling the two loops may not be necessary and a more advantageous model adaptation algorithm could be designed with an appropriate model.

8 References

[1] N. Mohan, T. M. Undeland, and W. P. Robbins, *Power electronics: converters, applications, and design*, 3rd ed. Hoboken, New Jersey, United States of America: John Wiley & Sons, 2003.

[2] ON Semiconductor, "Power Factor Correction (PFC) Handbook.", 2011, http://www.onsemi.com/pub_link/Collateral/HBD853-D.PDF.

[3] G. A. Karvelis, S. N. Manias, and G. Kostakis, "A comparative evaluation of power converters used for current harmonics elimination," in *8th International Conference On Harmonics and Quality of Power Proceedings*, vol. 1, 1998, pp. 227-232.

[4] C. Qiao and K. M. Smedley, "A topology survey of single-stage power factor corrector with a boost type input-current-shaper," in *Fifteenth Annual IEEE Applied Power Electronics Conference and Exposition*, vol. 1, 2000, pp. 460-467.

[5] K. Matsui et al., "A comparison of various buck-boost converters and their application to PFC," in *Proceedings of the 28th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, Sevilla, Spain, 2002, pp. 30-36.

[6] B. Singh et al., "A review of single-phase improved power quality AC-DC converters," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 5, pp. 962-981, October 2003.

[7] O. García, J. A. Cobos, R. Prieto, P. Alou, and J. Uceda, "Single phase power factor correction: a survey," *IEEE Transactions on Power Electronics*, vol. 18, no. 3, pp. 749-755, May 2003.

[8] F. Beltrame, L. Roggia, L. Schuch, and J. R. Pinheiro, "A comparison of high power single-phase power factor correction pre-regulators," in *2010 IEEE International Conference on Industrial Technology (ICIT)*, Viña del Mar, Chile, 2010, pp. 625-630.

[9] J. P. M. Figueiredo, F. L. Tofoli, and B. L. A. Silva, "A review of single-phase PFC topologies based on the boost converter," in *9th IEEE/IAS International Conference on Industry Applications (INDUSCON)*, São Paulo, Brazil, November 2010, pp. 1-6.

[10] M. M. Jovanović and Y. Jang, "State-of-the-art, single-phase, active power-factor-correction techniques for high-power applications - an overview," *IEEE Transactions on Industrial Electronics*, vol. 52, no. 3, pp. 701-708, June 2005.

[11] R. W. Erickson, *Fundamentals of power electronics*. New York, New York, United States of America: Chapman & Hall, 1997.

[12] P. C. Todd, "UC3854 Controlled Power Factor Correction Circuit Design", Unitrode Application Note U-134; http://www.ti.com/lit/an/slua144/slua144.pdf.

[13] J. C. Salmon, "Techniques for minimizing the input current distortion of current-controlled single-phase boost rectifiers ," *IEEE Transactions on Power Electronics*, vol. 8, no. 4, pp. 509-520, October 1993.

[14] L. H. Dixon, "Control Loop Cookbook", http://www.ti.com/lit/ml/slup113a/slup113a.pdf.

[15] C. P. Basso, *Switch-mode power supply SPICE Cookbook*. New York, New York, United States of America: McGraw-Hill, 2001.

[16] R. C. Dorf and R. H. Bishop, *Modern control systems*, 11th ed. Upper Saddle River, New Jersey, United States of America: Pearson Prentice Hall, 2008.

[17] K. Ogata, *Discrete-time control systems*, 2nd ed. Upper Saddle River, New Jersey, United States of America: Prentice Hall, 1995.

[18] J. G. Kassakian, M. F. Schlecht, and G. C. Verghese, *Principles of power electronics*. Reading, Massachusetts, United States of America: Addison-Wesley, 1991.

[19] S. Ziegler, R. C. Woodward, H. H.-C. Iu, and J. B. Lawrence, "Current Sensing Techniques: A Review," *IEEE Sensors Journal*, vol. 9, no. 4, pp. 354-376, 2009.

[20] R. Lenk, "Application Bulletin AB-20. Optimum Current Sensing Techniques in CPU Converters", 1998.

[21] L. Dixon, "Average Current Mode Control of Switching Power Supplies", Unitrode Application Note U-140; http://www.ti.com/lit/an/slua079/slua079.pdf.

[22] Linfinity Microelectronics, "Linfinity Application Note AN-7. A Simple Current-Sense Technique Eliminating a Sense Resistor", 1998.

[23] E. Dallago, M. Passoni, and G. Sassone, "Lossless current sensing in low-voltage high-current DC/DC modular supplies," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 6, pp. 1249-1252, December 2000.

[24] L. Hua and S. Luo, "Design considerations of time constant mismatch problem for inductor DCR current sensing method," in *Twenty-First Annual IEEE Applied Power Electronics Conference and Exposition*, 2006, pp. 1368-1374.

[25] V. Boscaino, P. Livreri, F. Marino, and M. Minieri, "Current-sensing technique for current-mode controlled voltage regulator modules," *Microelectronics Journal*, vol. 39, no. 12, pp. 1852-1859, December 2008.

[26] C. Chang, "Lossless current sensing and its application in current mode control," in *39th IEEE Annual Power Electronics Specialists Conference*, Rhodes, Greece, 2008, pp. 4086-4091.

[27] Z. Luo, C. Wang, M. Xu, P. Kong, and F. C. Lee, "DCR current sensing technique for PFC circuits," in *Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition*, 2008, pp. 1384-1388.

[28] F. J. Azcondo, A. de Castro, F. J. Diaz, and O. Garcia, "Current Sensorless Power Factor Correction based on Digital Current Rebuilding," in *Proceedings of the Twenty-Fourth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2009, pp. 774-779.

[29] J. F. Azcondo, A. de Castro, V. M. Lopez, and O. Garcia, "Power Factor Correction Without Current Sensor Based on Digital Current Rebuilding," *IEEE Transactions on Power Electronics*, vol. 25, no. 6, pp. 1527-1536, 2010.

[30] V. M. Lopez, F. J. Azcondo, F. J. Diaz, and A. de Castro, "Autotuning digital controller for current sensorless power factor corrector stage in continuous conduction mode," in *Proceedings of the IEEE 12th Workshop Control and Modeling for Power Electronics (COMPEL)*, 2010, pp. 1-8.

[31] Y.-K. Lo and N.-C. Lia, "Constant-frequency current slope control for single-phase switch-mode rectifiers without current sensors," in *Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, 1998, pp. 559-561.

[32] Y.-K. Lo, H.-J. Chiu, and S.-Y. Ou, "Constant-switching-frequency control of switch-mode rectifiers without current sensors," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 5, pp. 1172-1174, 2000.

[33] M. Chen and J. Sun, "Feedforward current control of boost single-phase PFC converters ," *IEEE Transactions on Power Electronics*, vol. 21, no. 2, pp. 338-345, March 2004.

[34] H. S. Athab and P. K. Shadhu Khan, "A Simple Effective Duty Cycle Controller for High Power Factor Boost Rectifier," in *7th International Conference on Power Electronics and Drive Systems*, Bangkok, Thailand, 2007, pp. 664-668.

[35] H. S. Athab and D. D.-C. Lu, "Simple controller for single-phase power factor correction rectifier," *IET Power Electronics*, vol. 3, no. 4, pp. 590-600, July 2010.

[36] H.-C. Chen, C.-C. Lin, and J.-Y. Liao, "Modified Single-Loop Current Sensorless Control for Single-Phase Boost-Type SMR With Distorted Input Voltage," *IEEE Transactions on Power Electronics*, vol. 26, no. 5, pp. 1322-1328, 2011.

[37] T. Ohnishi and M. Hojo, "DC voltage and current sensorless single phase PFC converter," in *Proceedings of the IEEE 33rd Annual Power Electronics Specialists Conference*, vol. 3, 2002, pp. 1137-1142.

[38] T. Ohnishi and M. Hojo, "DC voltage sensorless single-phase PFC converter," *IEEE Transactions on Power Electronics*, vol. 19, no. 2, pp. 404-410, 2004.

[39] I. W. Merfert, "Stored-duty-ratio control for power factor correction," in *Proceedings of the Fourteenth Annual Applied Power Electronics Conference and Exposition (APEC)*, vol. 2, 1999, pp. 1123-1129.

[40] I. Merfert, "Analysis and application of a new control method for continuous-mode boost converters in power factor correction circuits," in *28th Annual IEEE Power Electronics Specialists Conference (PESC)*, vol. 1, 1997, pp. 96-102.

[41] W. Zhang, G. Feng, Y.-F. Liu, and B. Wu, "A digital power factor correction (PFC) control strategy optimized for DSP," *IEEE Transactions on Power Electronics*, vol. 19, no. 6, pp. 1474-1485, 2004.

[42] A. P. Finazzi et al., "Current-sensorless PFC Boost converter with preprogrammed control strategy," in *2011 IEEE International Symposium on Industrial Electronics (ISIE)*, Gdańsk, Poland, 2011, pp. 182-187.

[43] A. Garcia, A. de Castro, O. Garcia, and F. J. Azcondo, "Pre-calculated duty cycle control implemented in FPGA for power factor correction ," in *35th Annual Conference of IEEE Industrial Electronics*, Porto, Portugal, 2009, pp. 2955-2960.

[44] H.-C. Chen, "Duty Phase Control for Single-Phase Boost-Type SMR," *IEEE Transactions on Power Electronics*, vol. 23, no. 4, pp. 1927-1934, 2008.

[45] H.-C. Chen and C.-K. Huang, "Current sensorless control for single-phase boost-type SMR," in *Proceedings of the Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2008, pp. 1221-1226.

[46] H.-C. Chen, "Single-Loop Current Sensorless Control for Single-Phase Boost-Type SMR," *IEEE Transactions on Power Electronics*, vol. 24, no. 1, pp. 163-171, 2009.

[47] J. W. Kimball and P. T. Krein, "A current-sensorless digital controller for active power factor correction control based on Kalman filters," in *Proceedings of the Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2008, pp. 1328-1333.

[48] H.-C. Chen, Z.-H. Wu, and J.-Y. Liao, "Modeling and Small-Signal Analysis of a Switch-Mode Rectifier With Single-Loop Current Sensorless Control," *IEEE Transactions on Power Electronics*, vol. 25, no. 1, pp. 75-84, 2010.

[49] Micrometals, Inc., "Power Conversion & Line Filter Applications", Issue L, 2007, http://micrometals.com/pcparts/PC_L.pdf.

[50] Magnetics, "Powder Core Catalog", 2011, http://www.mag-inc.com.

[51] G. Garcea, S. Saggini, D. Zambotti, and M. Ghioni, "Digital auto-tuning system for inductor current sensing in VRM applications," in *Twenty-First Annual IEEE Applied Power Electronics Conference and Exposition*, 2006, pp. 493-498.

[52] H. Pooya Forghani-zadeh and G. A. Rincón-Mora, "A lossless, accurate, self-calibrating current-sensing technique for DC-DC converters," in *31st Annual Conference of IEEE Industrial Electronics Society*, 2005, pp. 549-554.

[53] H. Pooya Forghani-zadeh and G. A. Rincón-Mora, "An Accurate, Continuous, and Lossless Self-Learning CMOS Current-Sensing Scheme for Inductor-Based DC-DC Converters," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 3, pp. 665-679, March 2007.

[54] Z. Lukić, Z. Zhao, S. M. Ahsanuzzaman, and A. Prodić, "Self-tuning digital current estimator for low-power switching converters," in *Twenty-Third Annual IEEE Applied Power Electronics Conference and Exposition*, 2008, pp. 529-534.

[55] P. Midya, P. T. Krein, and M. F. Greuel, "Sensorless current mode control—An observer-based technique for DC-DC converters," *IEEE Transactions on Power Electronics*, vol. 16, no. 4, pp. 522-526, July 2001.

[56] N. Mohan, *Power Electronics: A First Course*. United States of America: John Wiley & Sons, 2012.

[57] Panasonic Corporation, "Large Can Aluminum Electrolytic Capacitors. TS-HB Series", http://www.panasonic.com/industrial/components/pdf/pic_ts-hb_series_dne.pdf.

[58] R. D. Middlebrook and S. Ćuk, "A general unified approach to modeling switching-converter power stages," *International Journal of Electronics*, vol. 42, no. 6, pp. 521-550, 1977.

[59] S. M. Ćuk, "Modelling, analysis, and design of switching converters," Ph.D. thesis, California Institute of Technology, Pasadena, California, United States of America, 1977.

[60] H. D. Venable, "The K factor: A new mathematical tool for stability analysis and synthesis", without year, Linear Technology Reference Reading #4.

[61] The MathWorks, Inc., "SIMULINK. Dynamic System Simulation for MATLAB", Version 7.6 (R2010b), 2010, http://www.mathworks.com/products/simulink.

[62] The MathWorks, Inc., "SimPowerSystems. Model and simulate electrical power systems", Version 5.3 (R2010b), 2010, http://www.mathworks.com/products/simpower.

[63] M. B. Patil, V. Ramanarayanan, and V. T. Ranganathan, *Simulation of power electronic circuits*.: Alpha Science, 2009.

[64] The MathWorks, Inc., "Simulink User's Guide", 2010.

[65] C. Antoine, "A single-phase AC-to-DC boost power factor correction converter with low line current harmonic distortion," Master of Engineering project report, University of Victoria, Victoria, British Columbia, 2008.

[66] Texas Instruments Incorporated, "TMS320F2810, TMS320F2811, TMS320F2812, TMS320C2810, TMS320C2811, TMS320C2812 Digital Signal Processors. Data Manual", Rev. T, 2012, http://www.ti.com/litv/pdf/sprs174t.

[67] Spectrum Digital, Inc., "eZdspF2812 Technical Reference", Rev. F, 2003, http://c2000.spectrumdigital.com/ezf2812/docs/ezf2812_techref.pdf.

[68] Spectrum Digital, Inc., "eZdspF2812 Schematic", Rev. C, 2003, http://c2000.spectrumdigital.com/ezf2812/docs/ezf2812_schem.pdf.

[69] Texas Instruments Incorporated, "TMS320C28x DSP CPU and Instruction Set Reference Guide", Rev. E, 2009, http://www.ti.com/lit/ug/spru430e/spru430e.pdf.

[70] K. Ogata, *Discrete-time control systems*. Upper Saddle River, New Jersey, United States of America, 1987.

[71] Texas Instruments Incorporated, "Code Composer Studio (CCStudio) Integrated Development Environment (IDE) v5", 2012, http://www.ti.com/tool/ccstudio.

[72] Texas Instruments Incorporated, "TMS320x281x Event Manager (EV) Reference Guide", Rev. E, 2007, http://www.ti.com/litv/pdf/spru065e.

[73] Texas Instruments Incorporated, "C64x+ IQMath Library - A Virtual Floating Point Engine", 2012, http://www.ti.com/tool/sprc542.

[74] Texas Instruments Incorporated, "C281x C/C++ Header Files and Peripheral Examples", v. 1.20, 2009, http://www.ti.com/tool/sprc097.

[75] International Organization for Standardization, "International standard ISO/IEC 9899: Information technology - Programming languages - C", Third Edition, 2011.

[76] Texas Instruments Incorporated, "TMS320C28x Optimizing C/C++ Compiler v6.0 User's Guide", Rev. D, 2011.

[77] Texas Instruments Incorporated, "Hardware Design Guidelines for TMS320F28xx and TMS320F28xxx DSCs", Rev. B, 2011, http://www.ti.com/lit/an/spraas1b/spraas1b.pdf.

[78] ON Semiconductor, "MC33078, MC33079, NCV33078, NCV33079 Low Noise Dual/Quad Operational Amplifiers", Rev. 9, 2011, http://www.onsemi.com/pub/Collateral/MC33078-D.PDF.

[79] International Rectifier, "SMPS MOSFET IRF840A", 2003.

[80] Microchip Technology Inc., "MCP1406/07", 2012, http://ww1.microchip.com/downloads/en/DeviceDoc/22019B.pdf.

Appendix A

Design of the converter prototype

Appendix A outlines the dimensioning of the main circuit elements C_1 and L, supplementing the explanations in section 5.1. The original selection of the circuit components was done in [65] using the design equations from [1].

Inductor

The inductance had been chosen so that the converter operates in continuous conduction mode at the rated load and the peak-to-peak input/inductor current ripple under this condition is 25 % of the peak current. The rise in the inductor current while the transistor is on is $\Delta i_L = \frac{v_d}{L} D T_{sw}$. When the transistor is off, the inductor current falls by the same amount $\Delta i_L = \frac{V_o - v_d}{L}(1-D)T_{sw}$. Because $DT_{sw} + (1-D)T_{sw} = T_{sw} = \frac{1}{f_{sw}}$, one can write

$$\frac{1}{f_{sw}} = \Delta i_L L \left[\frac{1}{v_d} + \frac{1}{V_o - v_d}\right] \tag{A.1}$$

The maximum possible current ripple occurs when $v_d = \frac{1}{2} V_o$. In this application, the highest appearing v_d is $\hat{V}_{in} = \sqrt{2} \cdot 120\ \text{V}$. To limit Δi_L at this voltage to 25% of $\hat{I}_L = \sqrt{2} \cdot \frac{200\ \text{W}}{120\ \text{V}}$, L must be chosen as

$$\begin{aligned} L &\geq \frac{1}{\Delta i_L \cdot f_{sw} \left[\frac{1}{v_d} + \frac{1}{V_o - \frac{V_o}{2}}\right]} \\ &= \frac{1}{\left(0.25 \cdot \sqrt{2} \cdot \frac{200\ \text{W}}{120\ \text{V}}\right) \cdot 20\ \text{kHz} \cdot \left[\frac{1}{\sqrt{2} \cdot 120\ \text{V}} + \frac{1}{380\ \text{V} - \sqrt{2} \cdot 120\ \text{V}}\right]} \\ &= 7.97\ \text{mH} \approx 8\ \text{mH} \end{aligned} \tag{A.2}$$

Output capacitance

In case of the bulk output capacitor, its capacitance C_1 had been set to achieve a peak-to-peak output voltage ripple of less than 1.5 % of the nominal output voltage. In section 3.4, the peak value of the second harmonic of the output voltage, $\hat{V}_{o(2)}$, was derived. Its peak-to-peak value is twice the amount given by eq. (3.17), that is

$$V_{o,pp} = 2\hat{V}_{o(2)} = \frac{\hat{V}_d \hat{I}_L - R_L \hat{I}_L^2}{2\omega C \cdot V_o} \tag{A.3}$$

Neglecting the R_L term, the numerator in eq. (A.3) is $\hat{V}_d \hat{I}_L = 2P_{in} \approx 2P_o$. If one aims to achieve an output voltage ripple of less than 1.5 % of the nominal output voltage, then C_1 must fulfill

$$C_1 > \frac{P_o}{\omega \cdot 0.015\, V_o \cdot V_o} = \frac{200\ \text{W}}{2\pi \cdot 60\ \text{Hz} \cdot 0.015 \cdot (380\ \text{V})^2} = 245\ \mu\text{F} \tag{A.4}$$

The nearest standard value is 270 µF.

Appendix B

Digital controller source code

```
/* main.c */

/*front matter*/
#include "DSP281x_Device.h"   // define CPU commands, variable types, include peripheral header files
#include "DSP281x_Examples.h"    // define CPU clock, include various header files
#include "IQmathLib.h"           // IQmath library

interrupt void adc_isr();
void shutdown();

volatile int samplecount=0;   // counter for the number of samples taken in current switching cycle
int NSPSC;                    // number of samples per switching cycle
volatile long v_dsum=0;       // sum of v_d for current switching cycle
volatile long v_Qsum=0;       // sum of v_Q for current switching cycle
volatile long v_osum=0;       // sum of v_o for current switching cycle
volatile long i_Lsum=0;       // sum of i_L for current switching cycle
volatile int processed=1;     // completely sampled switching cycle has been processed (boolean)

int main() {

    unsigned int   i;         // for use in any small loop or other local temporary use

/* ============= system control ================================== */

EALLOW;
    SysCtrlRegs.WDCR=0x0068;              // disable watchdog timer
EDIS;
    InitPll(0xA);                         // set clock PLL to 10: SYSCLKOUT = 5*XCLKIN = 150 MHz
EALLOW;
    SysCtrlRegs.HISPCP.all=0x0000;        // set high-speed peripheral clock pre-scaler, factor 1
    SysCtrlRegs.LOSPCP.all=0x0002;        // set low-speed peripheral clock pre-scaler
    SysCtrlRegs.PCLKCR.bit.EVAENCLK=1;    // enable peripheral clock for event manager A
    SysCtrlRegs.PCLKCR.bit.ADCENCLK=1;    // enable peripheral clock for ADC
EDIS;
```

```
/* ============= GPIO pins ================================== */

// configure all GPIO pins as digital outputs and set to 0
EALLOW;
    GpioMuxRegs.GPAMUX.all=0;
    GpioMuxRegs.GPBMUX.all=0;
    GpioMuxRegs.GPDMUX.all=0;
    GpioMuxRegs.GPEMUX.all=0;
    GpioMuxRegs.GPFMUX.all=0;
    GpioMuxRegs.GPGMUX.all=0;
    GpioMuxRegs.GPADIR.all=0xFFFF;
    GpioMuxRegs.GPBDIR.all=0xFFFF;
    GpioMuxRegs.GPDDIR.all=0xFFFF;
    GpioMuxRegs.GPEDIR.all=0xFFFF;
    GpioMuxRegs.GPFDIR.all=0xFFFF;
    GpioMuxRegs.GPGDIR.all=0xFFFF;
    GpioDataRegs.GPACLEAR.all=0xFFFF;
    GpioDataRegs.GPBCLEAR.all=0xFFFF;
    GpioDataRegs.GPDCLEAR.all=0xFFFE;
    GpioDataRegs.GPECLEAR.all=0xFFFF;
    GpioDataRegs.GPFCLEAR.all=0xFFFF;
    GpioDataRegs.GPGCLEAR.all=0xFFFF;

    // configure required GPIO pins as peripheral (timer) outputs
    GpioMuxRegs.GPAMUX.bit.T1PWM_GPIOA6=1;
    GpioMuxRegs.GPAMUX.bit.T2PWM_GPIOA7=1;
EDIS;

/* ============= timers ===================================== */

// GP timer 1 setup -- provides time base for sampling
    EvaRegs.T1CON.bit.FREE=1;     // complete timer period on emulation suspend
    EvaRegs.T1CON.bit.SOFT=0;     // ... second part of it
    EvaRegs.T1CON.bit.TMODE=2;    // continuous up-counting mode
```

```
    EvaRegs.T1CON.bit.TPS=0;          // input clock prescaler, factor 1
    EvaRegs.T1CON.bit.TCLKS10=0;      // use HSPCLK as clock
    EvaRegs.T1CON.bit.TCLD10=0;       // reload compare register T1CMPR at beginning of sampling period
    EvaRegs.T1CON.bit.TECMPR=1;       // enable timer compare
    EvaRegs.T1PR=187;                 // period register = 150MHz/f_s-1
    EvaRegs.T1CMPR=60;                // compare register for GP T1
    EvaRegs.T1CNT=0;                  // initialize counter register

// GP timer 2 setup -- provides time base for switching
    EvaRegs.T2CON.bit.FREE=1;         // complete timer period on emulation suspend
    EvaRegs.T2CON.bit.SOFT=0;         // ... second part of it
    EvaRegs.T2CON.bit.TMODE=2;        // continuous up-counting mode
    EvaRegs.T2CON.bit.TPS=0;          // input clock prescaler, factor 1
    EvaRegs.T2CON.bit.T2SWT1=1;       // start together with T1
    EvaRegs.T2CON.bit.TCLKS10=0;      // use HSPCLK as clock
    EvaRegs.T2CON.bit.TCLD10=2;       // reload compare register T2CMPR immediately
    EvaRegs.T2CON.bit.TECMPR=1;       // enable timer compare
    EvaRegs.T2CON.bit.SET1PR=0;       // use own period register (rather than T1's)
    EvaRegs.T2PR=7519;                // period register = 150MHz/f_sw-1
    EvaRegs.T2CMPR=3000;              // compare register for GP T2
    EvaRegs.T2CNT=0;                  // initialize counter register

// GP timer control register of EVA (for TxPWM)
    EvaRegs.GPTCONA.bit.T2CTRIPE=0;   // disable trip function that can drive output to high Z
    EvaRegs.GPTCONA.bit.T1CTRIPE=0;
    EvaRegs.GPTCONA.bit.TCMPOE=1;     // enable compare output
    EvaRegs.GPTCONA.bit.T2TOADC=0;    // do not start ADC
    EvaRegs.GPTCONA.bit.T1TOADC=2;    // start ADC on period match
    EvaRegs.GPTCONA.bit.TCMPOE=1;     // enable timer compare outputs
    EvaRegs.GPTCONA.bit.T2CMPOE=1;    // enable timer 2 compare output
    EvaRegs.GPTCONA.bit.T1CMPOE=1;    // enable timer 1 compare output
    EvaRegs.GPTCONA.bit.T2PIN=1;      // polarity of T2PWM
    EvaRegs.GPTCONA.bit.T1PIN=1;      // polarity of T1PWM
//  timers are enabled before the main loop is entered
```

```
/* ============= ADC ======================================== */

    AdcRegs.ADCTRL1.bit.RESET=1;          // reset ADC
    asm(" RPT #50 || NOP");               // required delay after reset before modifying ADC registers
    AdcRegs.ADCTRL3.bit.ADCBGRFDN=0x3;    // power up bandgap/reference circuitry
    for (i=0;i<2500;i++) asm(" RPT #255 || NOP");    // required 5 ms delay (5.1 ms)
    AdcRegs.ADCTRL3.bit.ADCPWDN=1;        // power up rest of ADC
    for (i=0;i<20;i++) asm(" RPT #255 || NOP");      // required 20 us delay (38 us)

    AdcRegs.ADCTRL1.bit.SUSMOD=1;         // upon emulation suspend finish ADC sequence
    AdcRegs.ADCTRL1.bit.ACQ_PS=1;         // length of the S/H pulse in ADC clock cycles
    AdcRegs.ADCTRL1.bit.CPS=0;            // ADC clock prescaler, factor 1
    AdcRegs.ADCTRL1.bit.CONT_RUN=0;       // start-stop mode (as opposed to continuous conversion)
    AdcRegs.ADCTRL1.bit.SEQ_OVRD=0;       // disable sequencer override, NR
    AdcRegs.ADCTRL1.bit.SEQ_CASC=0;       // dual sequencer mode (as opposed to cascaded mode)
    AdcRegs.ADCTRL2.bit.RST_SEQ1=1;       // reset sequencer 1
    AdcRegs.ADCTRL2.bit.INT_ENA_SEQ1=1;   // enable interrupt request by SEQ1
    AdcRegs.ADCTRL2.bit.INT_MOD_SEQ1=0;   // interrupt after every EOS (as opposed to every second one)
    AdcRegs.ADCTRL2.bit.EVA_SOC_SEQ1=1;   // allow event manager A to start conversion sequence
    AdcRegs.ADCTRL3.bit.ADCEXTREF=0;      // use internal reference sources
    AdcRegs.ADCTRL3.bit.ADCCLKPS=3;       // ADC clock divider, factor 1/6
    AdcRegs.ADCTRL3.bit.SMODE_SEL=0;      // sequential sampling mode (as opposed to simultaneous)
    AdcRegs.ADCMAXCONV.bit.MAX_CONV1=3;   // # of conversions in autoconversion session of SEQ1 = 4
    AdcRegs.ADCCHSELSEQ1.bit.CONV00=0;    // sample ADCINA0 (v_d)
    AdcRegs.ADCCHSELSEQ1.bit.CONV01=1;    // sample ADCINA1 (v_o)
    AdcRegs.ADCCHSELSEQ1.bit.CONV02=2;    // sample ADCINA2 (v_Q)
    AdcRegs.ADCCHSELSEQ1.bit.CONV03=3;    // sample ADCINA3 (i_L)
    AdcRegs.ADCST.bit.INT_SEQ1_CLR=1;     // clear SEQ1 interrupt flag bit
```

```
/* ============= interrupts ================================== */

// configure interrupt control and peripheral interrupt expansion table
    InitPieCtrl();                          // clear all PIEIER and PIEIFR registers
    IER = 0x0000;                           // disable all interrupts on CPU level
    IFR = 0x0000;                           // clear CPU IR flag register
    InitPieVectTable();                     // initialize and enable PIE vector table
EALLOW;
    PieVectTable.ADCINT=&adc_isr;           // define adc_isr as interrupt service routine
EDIS;
    IER |= M_INT1;                          // enable CPU IR group 1
    EINT;                                   // enable interrupts globally
    PieCtrlRegs.PIECRTL.bit.ENPIE=1;        // enable PIE vector table
    PieCtrlRegs.PIEACK.all=0xFFFF;          // clear PIEACK
    PieCtrlRegs.PIEIER1.bit.INTx6=1;        // enable ADCINT
    ERTM;                                   // clear DBGM (debug enable mask bit): service halt requests and
                                            // breakpoints, allow emulator to access registers in real time

/* ===== variable declarations and constant definitions ===== */

    // global Q = 22 defined in IQmathLib.h
    #define    f_sw 19947                   // switching frequency; 1875/94 kHz
    static _iq f_sw_10 =0;                  // f_sw>>10 in IQ for controller parameters
    #define    V_oref  _IQ(380.0)           // reference output voltage
    #define    C     _IQ(270.0e-6*32)      // C <<5

    #define    L_0   _IQ(8.0e-3)            // initially implemented L
    #define    R_L_0 _IQ(1.0)               // initially implemented R_L

    // controller parameters
    #define    Ki    _IQ(31400.0/1024.0)   // current controller gain >>10
    #define    wzi   _IQ(5360.0/1024.0)    // current controller zero >>10
    #define    wpi   _IQ(74600.0/1024.0)   // current controller pole >>10
    #define    Kv    _IQ(0.0985*32)        // voltage controller gain <<5
```

```
#define    wzv  _IQ(22.4)        // voltage controller zero
#define    wpv  _IQ(176.0)       // voltage controller pole

#define    dmin 0                // lower bound for d
#define    dmax _IQ(1.0)         // upper bound for d
#define    kappamin _IQ30(0.0001) // lower bound for kappa
#define    kappamax _IQ30(0.022)  // upper bound for kappa
#define    eVmin   _IQ(-30.0)    // lower bound for eV
#define    eVmax   _IQ(30.0)     // upper bound for eV

static _iq a1i=0, a2i=0, b0i=0, b1i=0, b2i=0;    // current controller coefficients
static _iq30 a1v=0, a2v=0, b0v=0, b1v=0, b2v=0;  // voltage controller coefficients

static _iq L_imp=L_0;            // value of L implemented in the inductor model
static _iq R_L_imp=R_L_0;        // value of R_L implemented in the inductor model
static _iq L_est=L_0;            // estimated value of L
static _iq R_L_est=R_L_0;        // estimated value of R_L
static _iq L_corr[2];            // correction of L with respect to L_0 (with previous)
static _iq R_L_corr[2];          // correction of R_L with respect to R_L_0 (with previous)
static _iq deltaL[2];            // difference between previous and current estimated L (with previous)
static _iq deltaR_L[2];          // difference between previous and current estimated R_L (with previous)
static int adapt=0;              // enables/disables adaptation (boolean)

static int NSCPLHC;              // number of switching cycles per line half-cycle
static int NSCPLHC2;             // NSCPLHC/2
static long T2per;               // period of timer 2 (register value)

static int LHCcount=0;    // counter for the number of switching cycles completed in the current line half-cycle
static _iq v_L[2];               // average of v_L over one switching cycle (with previous)
static _iq v_o  =0;              // average of v_o over one switching cycle
static _iq i_LM[2];              // measured i_L (with previous)
static _iq i_LC[2];              // computed i_L (with previous)
   // most recent value has index 0: i_L[0]=i_L[k], i_L[1]=i_L[k-1]
```

```
static _iq V_dp =_IQ(170.0);      // peak input voltage
#define    LHCarlen 180           // length of the arrays that store data over one line half-cycle
static _iq v_dLHC[LHCarlen];      // all values of v_d for one line half-cycle
static _iq v_dsum10=0;            // sum of v_d from 10 switching cycles (for ZCD)
static _iq v_osumLQC1=0;          // sum of (v_o-V_oref) for first line quarter-cycle
static _iq v_osumLQC2=0;          // sum of (v_o-V_oref) for second line quarter-cycle
static _iq V_o2 =0;               // peak-to-peak value of the output voltage ripple = peak of 2nd harmonic

static int zcdavstart;            // value of LHCcount at which averaging of v_d starts
static _iq zcdthresh=_IQ(20.0);   // threshold value for line voltage zero crossing detection

static _iq sine[LHCarlen];        // look-up table with one half-cycle of sine values

static _iq v_Lsum =0;             // for integrating v_L
static _iq v_LsumLQC=0;           // to store v_Lsum after one line quarter-cycle is completed

static _iq fi_L1=0, fi_L2=0, fi_L3=0;    // factors used in the computation of i_L

static _iq eV[3];                 // output voltage error (with 2 previous)
static _iq eI[3];                 // inductor current error (with 2 previous)
static _iq30 kappa[3];            // emulated conductance; i_L = kappa*v_d (with 2 previous)
static _iq d[3];                  // duty cycle (with 2 previous)

#define    tauL _IQ(0.04)         // time constant for adjustment of L
static _iq fL1=0, fL2=0;          // factors used in adjustment of L
#define    tauR_L _IQ(0.04)       // time constant for adjustment of R_L
static _iq fR_L1=0, fR_L2=0;      // factors used in adjustment of R_L

static _iq facL;                  // factor for computing L
static _iq facR_L;                // factor for computing R_L
#define    oneover120pi _IQ(2.65258e-3)  // = 1/(120*pi)
static _iq Vdp_Vo2 =_IQ(1.0);     // V_dp/V_o2 >>8
```

```
    static long v_dbuf =0;            // switching cycle sum buffers
    static long v_obuf =0;
    static long v_Qbuf =0;
    static long i_Lbuf =0;

/* ============= computation of parameters ================== */

    f_sw_10=_IQ(((float)f_sw)/1024.0);    // f_sw>>10 in IQ
    NSPSC=40;                             // number of samples per switching cycle
    NSCPLHC=f_sw/120.0+.5;                // number of switching cycles per line half-cycle
    NSCPLHC2=NSCPLHC>>1;                  // NSCPLHC/2
    zcdavstart=NSCPLHC2+10;               // value of LHCcount at which averaging of v_d starts

    a1i=_IQdiv(f_sw_10<<2,((f_sw_10<<1)+wpi));          // current controller coefficients
    a2i=_IQ(1.0)-a1i;
    b0i=_IQmpy( _IQdiv(Ki,f_sw_10)>>1, _IQdiv( ((f_sw_10<<1)+wzi), ((f_sw_10<<1)+wpi) ));
    b1i=_IQmpy( _IQdiv(Ki,f_sw_10<<1), _IQdiv( wzi<<1, ((f_sw_10<<1)+wpi) ));
    b2i=b1i-b0i;

    a1v=_IQ30div(f_sw_10<<2,((f_sw_10<<1)+(wpv>>10)));   // voltage controller coefficients
    a2v=_IQ30(1.0)-a1v;
    b0v=_IQ30rmpy( _IQ30div(Kv,f_sw_10), _IQ30div((f_sw_10<<1)+(wzv>>10),(f_sw_10<<1)+(wpv>>10)) );
    b1v=_IQ30rmpy( _IQ30div(Kv,f_sw_10), _IQ30div(wzv>>9,(f_sw_10<<1)+(wpv>>10)) );
    b2v=b1v-b0v;
        // all bv are <<16

    fL1=_IQdiv( _IQmpy(tauL,f_sw_10<<1)-(_IQ(1.0)>>10), _IQmpy(tauL,f_sw_10<<1)+(_IQ(1.0)>>10));
        // factors used in adjustment of L, = (2*tauL*f_sw-1)/(2*tauL*f_sw-1)
    fL2=_IQdiv(_IQ(1.0),_IQmpy(tauL<<7,f_sw_10)+(_IQ(1.0)>>4));
        // = 1/(2*tauL*f_sw+1) <<4
    fR_L1=_IQdiv( _IQmpy(tauR_L,f_sw_10<<1)-(_IQ(1.0)>>10), _IQmpy(tauR_L,f_sw_10<<1)+(_IQ(1.0)>>10));
        // factors used in adjustment of R_L, = (2*tauR_L*f_sw-1)/(2*tauR_L*f_sw-1)
    fR_L2=_IQdiv(_IQ(1.0),_IQmpy(tauR_L<<7,f_sw_10)+(_IQ(1.0)>>4));
        // = 1/(2*tauR_L*f_sw+1) <<4
```

```
    facR_L=_IQdiv(_IQ(1.0),_IQmpy(_IQmpy(f_sw_10,C),V_oref));   // factor for computing R_L, = 1 / (f_sw*C*V_oref) <<8
               // divisor: >>10 from f_sw_10, <<5 from C, >>3 from factor 8; compensate with >>8 from Vdp_Vo2
    facL=_IQmpy( _IQdiv(_IQ(1.0),_IQmpy(_IQmpy(f_sw_10,C),V_oref)) , _IQ(0.021221));
               // factor for computing L, =  1 / (f_sw*C*V_oref*15*pi) <<10
               // divisor: >>10 from f_sw_10, <<5 from C, >>5 from factor 480/15;
               // compensate with >>8 from Vdp_Vo2 and >>2 from v_LsumLQC

    T2per=EvaRegs.T2PR;

/* ============= array initializations ====================== */

    v_L[0]=v_L[1]=0;
    i_LM[0]=i_LM[1]=0;
    i_LC[0]=i_LC[1]=0;
    for (i=0;i<3;i++){
        eV[i]=0;                  // previous voltage errors
        eI[i]=0;                  // previous current errors
        kappa[i]=_IQ(.015);       // previous voltage controller outputs
        d[i]=_IQ(.7);             // previous current controller outputs
    }
    L_corr[0]=L_corr[1]=0;
    R_L_corr[0]=R_L_corr[1]=0;
    deltaL[0]=deltaL[1]=0;
    deltaR_L[0]=deltaR_L[1]=0;

    for (i=0;i<NSCPLHC;i++)       // look-up table with one half-cycle of sine values
        sine[i]=_IQsinPU(_IQdiv(_IQ(i),_IQ(NSCPLHC*2)));
    for (i=NSCPLHC;i<LHCarlen;i++)
        sine[i]=0;


    EvaRegs.T1CON.bit.TENABLE=1;  // enable timers 1 and 2
```

```
// ============= infinite loop ==============================
    while(1){

    if (samplecount==NSPSC && processed==0){
            // when one entire switching cycle has been captured and not yet been processed

        EALLOW; GpioDataRegs.GPBSET.bit.GPIOB5=1; EDIS;
            // use GPIOB5 (pin 35) to indicate that computation is in progress; busy flag
        v_dbuf=v_dsum;          // copy switching cycle sums to buffers
        v_obuf=v_osum;
        v_Qbuf=v_Qsum;
        i_Lbuf=i_Lsum;
        v_dsum=0;               // reset switching cycle sums
        v_Qsum=0;
        v_osum=0;
        i_Lsum=0;
        samplecount=0;          // reset counter for the number of samples taken in the current switching cycle

        /*compute switching cycle averages*/
        v_L[1]=v_L[0];          // copy values from previous switching cycle to higher indices
        i_LM[1]=i_LM[0];
        /* taking voltage divider ratio, left alignment of conversion register value, conversion to voltage
         * value from 12-bit integer conversion result and division by NSPSC into account */
        if (v_dbuf>17383) v_dLHC[LHCcount]=_IQmpyI32(_IQ(.0011574),(v_dbuf-17383)>>4);
            else v_dLHC[LHCcount]=0;
        if (v_obuf>8184) v_o=_IQmpyI32(_IQ(.0012979),(v_obuf-8184)>>3);
               else v_o=_IQ(0.0);
        v_L[0]=v_dLHC[LHCcount]-_IQmpyI32(_IQ(.0013071),(v_Qbuf+1214)>>3);
        if (LHCcount<20 && v_L[0]<0)
            v_L[0]=v_L[1]=0;           // restrict v_L to positive voltages at beginning of LHC
        i_LM[0]=_IQmpyI32(_IQ(.0013072),(i_Lbuf+17076)>>10);
```

```
/*overvoltage/overcurrent protection*/
        if ( v_o>_IQ(418.0) || i_LM[0]>_IQ(3.4) ){
            shutdown();           // force gating signal to low, set error flag, disable interrupts
            return 0;             // exit program
        }

        /*determine peak input voltage of current line half-cycle*/
        // compute the mean value of four values around T/4
        if (LHCcount==NSCPLHC2+1){
            V_dp=( (v_dLHC[LHCcount]>>1)+(v_dLHC[NSCPLHC2]>>1)+(v_dLHC[NSCPLHC2-1]>>1)+(v_dLHC[NSCPLHC2-2]>>1) )>>1;
            zcdthresh=V_dp>>2; // threshold for ZCD
        }

        /*determine V_o(2)*/
        if (LHCcount<NSCPLHC){                    // only consider values from feasible interval
            if (LHCcount<=NSCPLHC2)
                v_osumLQC1+=(v_o-V_oref)>>1;      // integrate v_or during first line quarter-cycle
            else v_osumLQC2+=(v_o-V_oref)>>1;     // integrate v_or during second line quarter-cycle
        // V_o2 is computed at ZC
        }

        /*compute estimated L, R_L*/
        v_Lsum+=v_L[0]>>2;                        // add current v_L to sum
        if (LHCcount==NSCPLHC2)                   // line quarter-cycle completed
            v_LsumLQC=v_Lsum;                     // v_LsumLQC corresponds to S/H output for L
        // estimates are computed at ZC

        /*update implemented L, R_L*/
        if (adapt){                               // when adaptation is enabled
        deltaL[0]=L_est-L_0;                      // compute model parameter error
        L_corr[0]=_IQmpy(fL1,L_corr[1]) + _IQmpy(fL2,(deltaL[0]+deltaL[1])>>4); // LP filter for deltaL
        L_imp=L_0+L_corr[0];
```

```
deltaR_L[0]=R_L_est-R_L_0;
R_L_corr[0]=_IQmpy(fR_L1,R_L_corr[1]) + _IQmpy(fR_L2,(deltaR_L[0]+deltaR_L[1])>>4);  // LP filter for deltaR_L
R_L_imp=R_L_0+R_L_corr[0];

L_corr[1]=L_corr[0];      // overwrite values from last cycle with current ones
R_L_corr[1]=R_L_corr[0];
deltaL[1]=deltaL[0];
deltaR_L[1]=deltaR_L[0];
}

/*compute i_L*/
i_LC[1]=i_LC[0];          // copy value from previous switching cycle to higher index
fi_L1=_IQmpy(f_sw_10,L_imp<<10);
fi_L2=_IQdiv(fi_L1-(R_L_imp>>1),fi_L1+(R_L_imp>>1));
fi_L3=_IQdiv(_IQ(.5),fi_L1+(R_L_imp>>1));
i_LC[0]=_IQrmpy(fi_L2,i_LC[1])+_IQrmpy(fi_L3,v_L[0]+v_L[1]);
    // v_L/(R+sL) as difference equation

/*voltage controller*/
kappa[2]=kappa[1];        // copy values from previous switching cycles to higher indices
kappa[1]=kappa[0];
eV[2]=eV[1];
eV[1]=eV[0];
eV[0]=_IQsat(V_oref-v_o,eVmax,eVmin);            // compute and limit voltage error
    // controller equation: kappa[k] = a1v*kappa[k-1] + a2v*kappa[k-2] + b0v*eV[k] + b1v*eV[k-1] + b2v*eV[k-2];
    // all bv are <<16
kappa[0]= _IQ30mpy(a1v,kappa[1]) + _IQ30mpy(a2v,kappa[2]) + ((_IQ30mpyIQX(b0v,30, eV[0]>>8,GLOBAL_Q)
    + _IQ30mpyIQX(b1v,30, eV[1]>>8,GLOBAL_Q) + _IQ30mpyIQX(b2v,30, eV[2]>>8,GLOBAL_Q))>>8);
kappa[0]=_IQsat(kappa[0],kappamax,kappamin);     // limit kappa
```

```
/*current controller*/
d[2]=d[1];             // copy values from previous switching cycles to higher indices
d[1]=d[0];
eI[2]=eI[1];
eI[1]=eI[0];
eI[0]=_IQmpyIQX(_IQmpy(V_dp,sine[LHCcount]),GLOBAL_Q, kappa[0],30) - i_LC[0];  // compute current error
eI[0]=_IQsat(eI[0],_IQ(1.0),_IQ(-1.0));      // limit eI
    // controller equation: d[k] = a1i*d[k-1] + a2i*d[k-2] + b0i*eI[k] + b1i*eI[k-1] + b2i*eI[k-2];
d[0]=_IQmpy(a1i,d[1]) + _IQmpy(a2i,d[2]) + _IQmpy(b0i,eI[0])+ _IQmpy(b1i,eI[1]) + _IQmpy(b2i,eI[2]);
d[0]=_IQsat(d[0],dmax,dmin);                 // limit d
EvaRegs.T2CMPR=_IQmpyI32int(d[0],T2per);     // update compare register

/*determine whether zero crossing of line voltage occurred*/
// average v_d over 10 switching cycles (v_dsum10)
if (LHCcount==zcdavstart)
    v_dsum10=
         (v_dLHC[zcdavstart]>>2)+(v_dLHC[zcdavstart-1]>>2)+(v_dLHC[zcdavstart-2]>>2)+(v_dLHC[zcdavstart-3]>>2)
        +(v_dLHC[zcdavstart-4]>>2)+(v_dLHC[zcdavstart-5]>>2)+(v_dLHC[zcdavstart-6]>>2)
        +(v_dLHC[zcdavstart-7]>>2)+(v_dLHC[zcdavstart-8]>>2)+(v_dLHC[zcdavstart-9]>>2);
else {if (LHCcount>zcdavstart){
    v_dsum10=v_dsum10-(v_dLHC[LHCcount-10]>>2)+(v_dLHC[LHCcount]>>2);
        // subtract value from 10 switching cycles before and add latest one
    if (v_dsum10<=zcdthresh){
        // when averaged v_d drops below threshold, ZC occurs
        EALLOW; GpioDataRegs.GPFSET.bit.GPIOF4=1; EDIS;      // use GPIOF4 (pin 3) to flag ZC
        v_dLHC[0]=v_dLHC[LHCcount];       // copy current value to the beginning of the array
        LHCcount=-1;                      // reset the switching cycle counter
        i_LC[0]=i_LC[1]=0;                // reset computed i_L to avoid integrator drift
        d[0]=d[1]=_IQ(1.0);               // force d to 1 to alleviate cusp distortion

        // compute V_o2
        V_o2=_IQmpy(_IQ(.0188996),v_osumLQC2-v_osumLQC1);
        v_osumLQC1=0;                     // reset integrals of v_or
        v_osumLQC2=0;
```

```
            // calculate R_L_est and L_est from integration results
            Vdp_Vo2=_IQdiv(V_dp>>8,V_o2);                          // V_dp/V_o2 >>8
            R_L_est=_IQmpy( _IQmpy(v_Lsum,facR_L) , Vdp_Vo2<<2 );   // compute estimate for R_L
            L_est=_IQmpy( _IQmpy(v_LsumLQC>>2,facL) , Vdp_Vo2<<2 ) - _IQmpy(R_L_est,oneover120pi);
                // compute estimate for L

            v_Lsum=0;
            v_LsumLQC=0;
        }}
        else if (LHCcount==20){
            EALLOW; GpioDataRegs.GPFCLEAR.bit.GPIOF4=1; EDIS;}          // reset ZC flag pin
        }

        LHCcount++;  // increment counter for the number of switching cycles completed in the current line half-cycle
        if (LHCcount>=180){       // failed to detect line voltage zero crossing
            shutdown();           // force gating signal to low, set error flag pin, disable interrupts
            return 0;             // exit program
        }

        processed=1;                                          // mark current switching cycle as processed
        EALLOW; GpioDataRegs.GPBCLEAR.bit.GPIOB5=1; EDIS;    // clear busy flag
        }   // end of if (samplecount==NSPSC && processed==0)
    }   // end of while(1)
}   // end of main
```

```
interrupt void adc_isr(){
    v_dsum+=AdcRegs.ADCRESULT0;
    v_osum+=AdcRegs.ADCRESULT1;
    v_Qsum+=AdcRegs.ADCRESULT2;
    i_Lsum+=AdcRegs.ADCRESULT3;

    AdcRegs.ADCTRL2.bit.RST_SEQ1=1;                 // reset sequencer 1
    AdcRegs.ADCST.bit.INT_SEQ1_CLR=1;               // clear SEQ1 interrupt flag bit
    PieCtrlRegs.PIEACK.all=PIEACK_GROUP1;           // clear acknowledge bit for group 1

    samplecount++;                                  // increment sample counter
    processed=0;                                    // mark current switching cycle as unprocessed
}

void shutdown(){
    EvaRegs.GPTCONA.bit.T2PIN=0;                    // force gating signal to low
    EALLOW; GpioDataRegs.GPASET.bit.GPIOA2=1; EDIS; // use GPIOA2 (pin 11) to flag error
    DINT;                                           // disable all interrupts
}
```

Appendix C

Calibration of the analog-to-digital converter

The analog-to-digital converter found in the digital controller ideally has a linear relationship between the input voltage and the conversion result register value with a fixed slope and zero offset. In practice, the characteristic may differ due to gain, offset and nonlinearity errors. To compensate for the gain and offset errors of the ADC, its behaviour is determined experimentally. Results show a slight dependence on the ADC settings, which must therefore match the ones that are used during converter operation ($f_s = 797.872\ldots$ kHz, `CPS=0`, `ADCCLKPS=3`, `ACQPS=1`).

This procedure was carried out separately for each channel that is used. For the acquisition of the output voltage, the rectified input voltage and the switch voltage, the voltage to be digitized was applied in the circuit with a regulated DC power supply. This way, the respective voltage divider and voltage follower are factored in as well. In case of the inductor current, a test current was passed through the sensing resistor. The conversion results were recorded as a function of the current. Table B.1 and Figure B.1 contain the measurement data.

Table B.1: Analog-to-digital converter calibration measured data

v_d [V]	N	v_o [V]	N	v_Q [V]	N	i_L [A]	N
0	51	0	430	0	303	0.0	50
10	3664	20	3451	20	3323	-0.2	3957
20	7275	40	6502	40	6283	-0.4	7831
30	10824	60	9553	60	9227	-0.6	11692
40	14321	80	12550	80	12153	-0.8	15557
50	17798	100	15508	100	15160	-1.0	19562
60	21244	120	18622	120	18218	-1.2	23461
70	25034	140	21755	140	21176	-1.4	27490
80	28387	160	24697	160	24254	-1.6	31288
90	31562	180	27748	180	27238	-1.8	35157
100	35129	200	30813	200	31179	-2.0	39026
110	38498	220	33743	220	33293	-2.2	43046
120	41903	240	37161	240	36341	-2.4	47128
130	45413	260	40273	260	39440	-2.6	50992
140	48761	280	43375	280	42569	-2.8	54498
150	52130	300	46408	300	45742	-3.0	58700
160	55720	320	49550	320	48857	-3.2	62907
170	59190	340	52721	340	51947	-3.4	65520
180	62485	360	55812	360	55097		
185	64220	380	58858	380	58261		
		400	61779	400	61464		
		415	64381	415	63954		

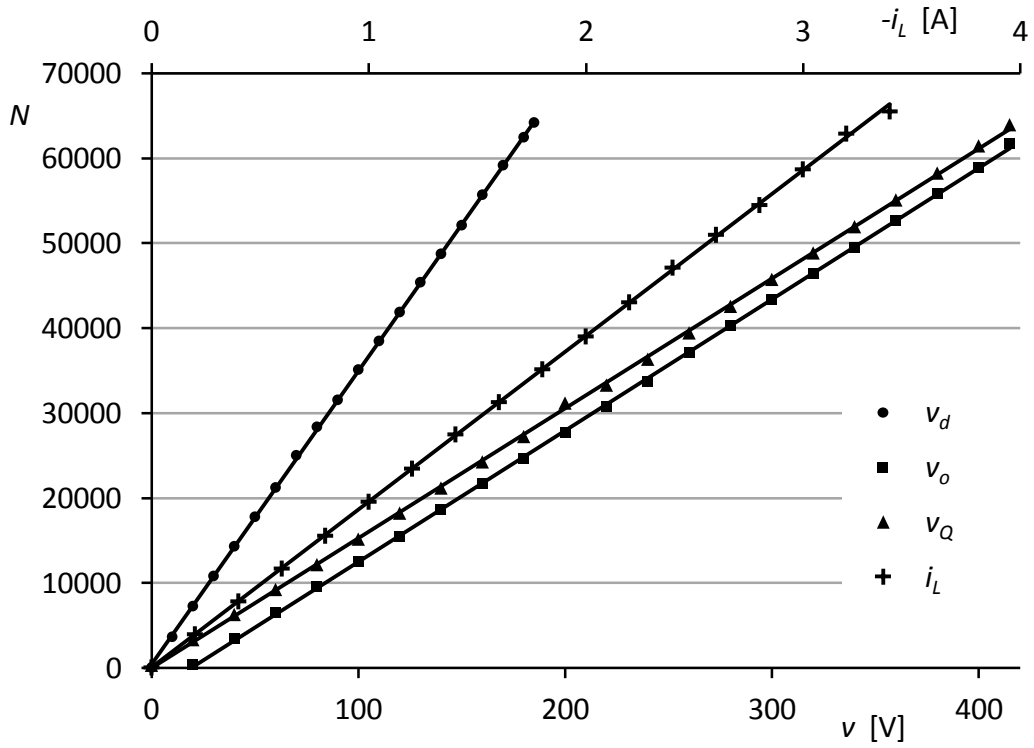


Figure B.1: Analog-to-digital converter calibration

Linear regressions lines were found from the register values as functions of the applied signals (Table B.2). The obtained expressions were then inverted to determine the values of the converted signals.

Table B.2: Analog-to-digital converter calibration regression lines

signal	ADC channel	linear regression line	
v_d	ADCINA0	$N = 345.6 \cdot \frac{v_d}{\mathrm{V}} + 435$	$\frac{v_d}{\mathrm{V}} = 0.002894 \cdot (N - 435)$
v_o	ADCINA1	$N = 154.1 \cdot \frac{v_o}{\mathrm{V}} + 205$	$\frac{v_o}{\mathrm{V}} = 0.006489 \cdot (N - 205)$
v_Q	ADCINA2	$N = 153.0 \cdot \frac{v_Q}{\mathrm{V}} - 30$	$\frac{v_Q}{\mathrm{V}} = 0.006535 \cdot (N + 30)$
i_L	ADCINA3	$N = -19584 \cdot \frac{i_L}{\mathrm{A}} - 21$	$\frac{i_L}{\mathrm{A}} = -5.106 \cdot 10^{-5} \cdot (N + 21)$

Appendix D

Inductor details

Table C.1: Comparison of the inductors [49]

		inductor 1		inductor 2	
core	type	toroidal iron powder core			
	part number	T184-40		T300-2D	
	outer diameter [in/mm]	1.84	46.7	3.04	77.2
	inner diameter [in/mm]	0.95	24.1	1.93	49.0
	height [in/mm]	0.71	18.0	1.00	25.4
	B_r [G/mT]	1000	100	36	3.6
	H_c $\left[\mathrm{Oe}/\frac{\mathrm{A}}{\mathrm{m}}\right]$	4.6	366	3.5	279
	$\frac{\mathrm{nH}}{N^2}$ [1]	143.0		22.8	
	tolerance in $\frac{\mathrm{nH}}{N^2}$	$\pm 10\%$		$\pm 5\%$	
	hysteresis curve	Figure 6.6		Figure 6.7	
winding	N	236		592	
	wire gauge	AWG 20		AWG 18	
	cross sectional area	0.519 mm^2		0.823 mm^2	
	DC resistance[2]	0.61 Ω		1.18 Ω	
inductance	theoretical inductance[3]	7.96 mH		7.99 mH	
	measured inductance[4]	8.35 mH 9.20 mH 8.50 mH 8.42 mH		7.85 mH 7.85 mH 7.85 mH 7.97 mH	
	$\frac{L(\hat{H})}{L(H\to 0)}$	0.47		0.98	

[1] — at $f = 10$ kHz and $\hat{B} = 10$ G $= 1$ mT
[2] — see Table C.2
[3] — $L = \left(\frac{\mathrm{nH}}{N^2}\right) \cdot N^2$, valid for low excitations
[4] — from top to bottom: using Philips RCL meter PM 6303,
using HP LCR meter 4262A with $f = 120$ Hz, 1 kHz, 10 kHz

Table C.2: Measurement of the inductors' DC resistance

inductor 1			inductor 2		
I [A]	V [V]	R [Ω]	I [A]	V [V]	R [Ω]
1.006	0.6067	0.603	1.002	1.181	1.179
2.007	1.233	0.614	2.003	2.368	1.182
3.001	1.847	0.615	3.001	3.533	1.177