control design for power electronics

Single-Phase

Voltage Source Inverter (VSI)

Tutorial –December 2018-

How to Contact:

info@powersmartcontrol.com

www.powersmartcontrol.com

Notice

Power Smart Control tutorials or other design advice, services or information, including, but not limited to, reference designs, are intended to assist designers who are developing applications that use SmartCtrl; by downloading, accessing or using any particular Power Smart Control resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this notice.

Power Smart Control reserves the right to make corrections, enhancements, improvements and other changes to its resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications with all applicable regulations, laws and other applicable requirements.

Disclaimer

Power Smart Control S.L. (PSC) makes no representation or warranty with respect to the adequacy or accuracy of this documentation or the software which it describes. In no event will PSC or its direct or indirect suppliers be liable for any damages whatsoever including, but not limited to, direct, indirect, incidental, or consequential damages of any character including, without limitation, loss of business profits, data, business information, or any and all other commercial damages or losses, or for any damages in excess of the list price for the licence to the software and documentation.

Table of contents

1. Introduction

This tutorial has been done to show how SmartCtrl can be applied to design a generic control system. In this case, a single-phase voltage-source inverter will serve as an example to demonstrate the SmartCtrl capabilities to design multiple-loops structures.

Along the tutorial, several aspects will be highlighted:

1. The SmartCtrl´ s "Equation Editor" module can be applied to develop small signal models for the power converter (plant), current and voltage sensors, etc. This is a very easy to use tool that allows the user to operate transfer functions (complex functions in nature) as simple variables of an equation. Any text editor can be used to write the "text code" containing the model. In addition, The Equation Editor module is provided with its own text editor, with some syntax examples to ease the "model generation" procedure.

2. The steady-state values, plant and sensor transfer functions for both loops can be stored in the same "text code" and, along the design process, send to the design environment the transfer function required for each particular loop.

3. In the design process, it is very useful to use some transfer functions of the inner loop to design the outer loop. SmartCtrl includes a powerful import / export function that allows the user to visualize in the design environment, at any time; any transfer function that might be required.

4. Digital control inherent delay can be considered at any time just by clicking on the corresponding icon. No manual design is needed, it is only needed the specification of the sampling frequency, bits number, etc and SmartCtrl will compute the delay and show the corresponding Bode plots, as well as the compensator digital coefficients.

The proposed control structure for the single-phase voltage-source inverter is shown in Figure 1.

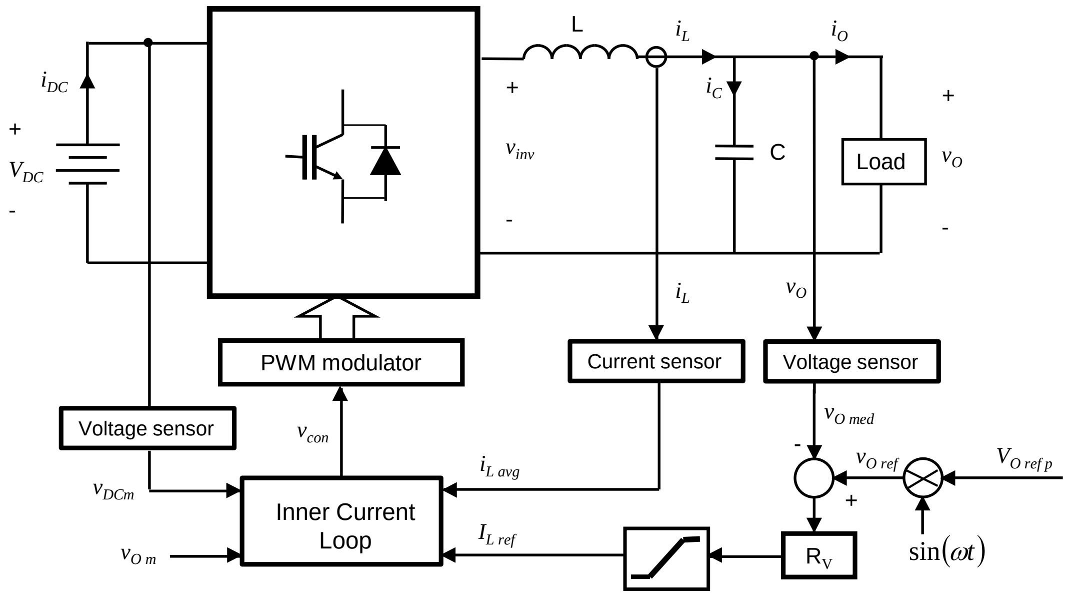

Figure 1: VSI inverter control structure

The tutorial is structured as follows:

1. First, a simple theoretical introduction, regarding the double-loop control of a single-phase voltage-source inverter, is provided. According to this theoretical model, the complete "text code" is given in ANNEX A.

2. Once the procedure to design both loops has been shown, some PSIM simulations are provided to show the effect of the use of the feed-forward control actions.

<u>The main specifications are the following:</u>

- General parameters

  S_{NOM} = 5 kVA Nominal apparent power

  $V_{DC\,m}$ = 400 V, DC input voltage.

  $V_{O\,rms}$ = 230 V, Reference rms output voltage

  f = 50Hz; Output voltage frequency

- The inverter uses a unipolar PWM modulation with 10 kHz carrier frequency.
- The output LC filter components are:

  L = 200 μH

  RL = 100 mΩ (Inductor equivalent series resistance)

  C = 33 μF.

  RC = 10 mΩ (Capacitor equivalent series resistance)

- The voltage sensor parameters are:

  Gain: K_v = 10/500 = 0.02

Cut-off frequency = 3 kHz

- The current sensor parameters are:

  Gain: K_i = 10/40 = 0.25

  Cut-off frequency = 3 kHz

The design of nested control loops is carried out from the inside to the outside of the control structure. This is, the inner control loop is designed first and then, the outer control loop is designed. So, in this case, the current control loop must be designed first.

2. The design of the current control loop

The first step is to develop a basic dynamic modelling of the inner current loop. Let's consider v_{inv}, as the output voltage of the IGBT bridge, when the inverter switching output voltage is averaged at switching frequency. Thus switching ripple is neglected.

If unipolar PWM modulation is considered, Vinv is given by [1].

$$\mathrm{v}_{\mathrm{inv}} = \frac{v_{DC}}{V_{tri\,p}} v_{\mathrm{mod}} \qquad [1]$$

Where: Vtrip is the amplitude of the triangular carrier signal and Vmod is the modulating signal.

Then, the average inductor voltage (switching ripple is neglected) is given by [2].

$$v_L = L\frac{di_L}{dt} + RL \cdot i_L = v_{inv} - v_O \qquad [2]$$

Since this is already a linear system, the inductor current, can be expressed as in [3].

$$i_L = \frac{\frac{v_{DC}}{V_{tri\,p}} v_{\mathrm{mod}} - v_O}{s \cdot L + RL} = \frac{\frac{v_{DC}}{V_{tri\,p}} v_{\mathrm{mod}} - v_O}{Z_L(s)} \qquad [3]$$

At this point, the power stage is already modelled. Therefore, the next step is the modelling of the current sense and conditioning in the Laplace domain. The sensed current will be the first harmonic component of the actual current (low pass filter) scaled down by the sensor constant (Ki), as expressed in [4].

$$\mathrm{i}_{\mathrm{L\,med}} = \frac{\mathrm{K}_i}{1 + \frac{s}{\omega_{coi}}} \mathrm{i}_L = G_{cs}(s) \cdot \mathrm{i}_L \qquad [4]$$

Where: Ki is the sensor constant and ωcoi is the cut-off angular frequency (ωcoi=2·π·fcoi).

From the equations [3] and [4], the block diagram of the inner control loop can be obtained and it is shown in Figure 2.

Note that the power plant presents two external perturbations: the inverter input voltage (vDC) and the output voltage (vo) that must be taken into account.

Therefore, the use of feed-forward techniques will help to simplify the plant that is going to be controlled. The main target of the feed-forward is the decoupling of the power stage plant from the external perturbations (vDC and vo).

To do so, the insertion of an additional transfer function that accounts for the feed-forward action is needed: FF(s) in Figure 3. This function is unknown a priori, and it is determined as follows.

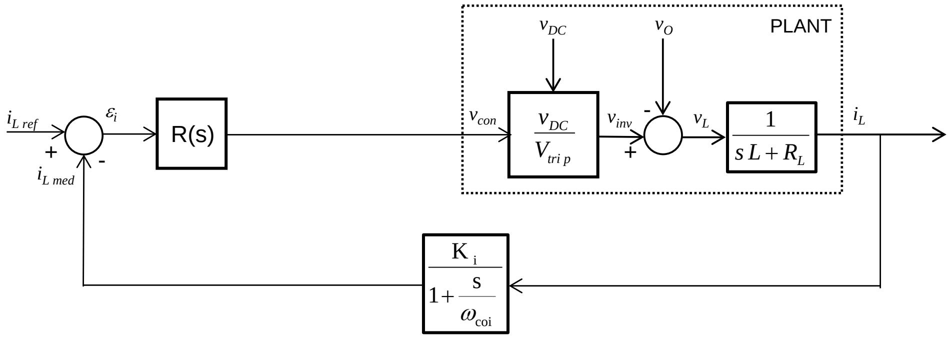

Figure 2: Block diagram of the inner current control loop

2.1 How is the feed-forward function (FF(s)) determined?

Since the objective of the feed-forward is the decoupling of the power stage plant from

Vdc and Vo, then it can be said that the insertion of FF(s) is intended to transform the block diagram in Figure 2 into the one depicted in Figure 3. In each case, the equations that define the inductor voltage (VL) as a function of the regulator output (X1) are given in [5] and [6].

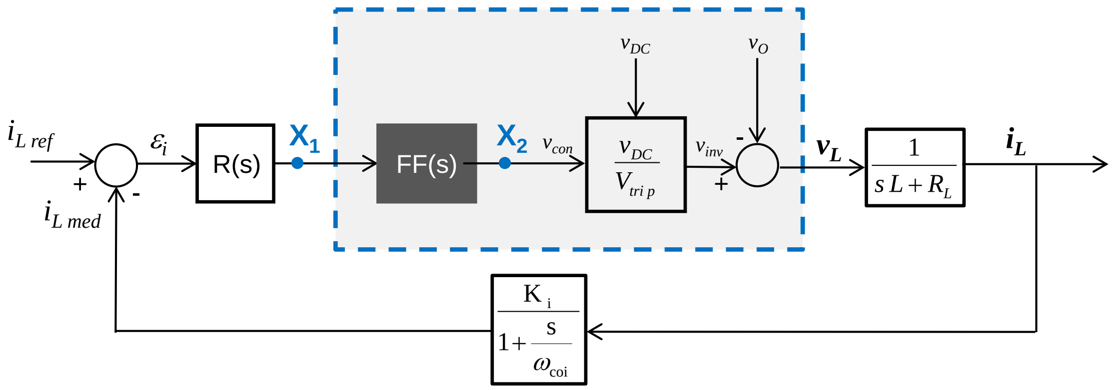

Figure 3: Block diagram of the inner current control loop with FF

$$i_{L\,med} = \frac{K_i}{1+\dfrac{s}{\omega_{coi}}}\, i_L = G_{cs}(s)\cdot i_L \qquad [4]$$

$$X_1 \cdot FF(s) \cdot \frac{v_{DC}}{V_{trip}} - v_o = v_L \;\Rightarrow\; X_2 \cdot \frac{v_{DC}}{V_{trip}} - v_o = v_L \qquad [5]$$

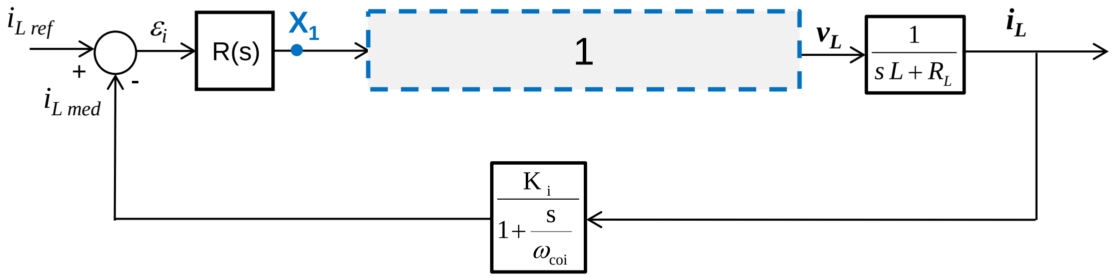

Figure 4: Objective block diagram of the inner current control loop with FF

$$X_1 = v_L \qquad [6]$$

Given (5) and (6), in order to achieve a successful feed-forward, they must be equal [7]. And, from this equality, the relationship between X1 and X2 can be determined [8] and thus the content of the black-box FF(s).

$$X_2 \cdot \frac{v_{DC}}{\mathrm{V}_{\mathrm{trip}}} - v_o = X_1 \qquad [7]$$

$$X_2 = (X_1 + v_o) \cdot \frac{V_{trip}}{v_{DC}} \qquad [8]$$

Therefore, the final block diagram of the inner current loop including the feed-forward is depicted in Figure 5.

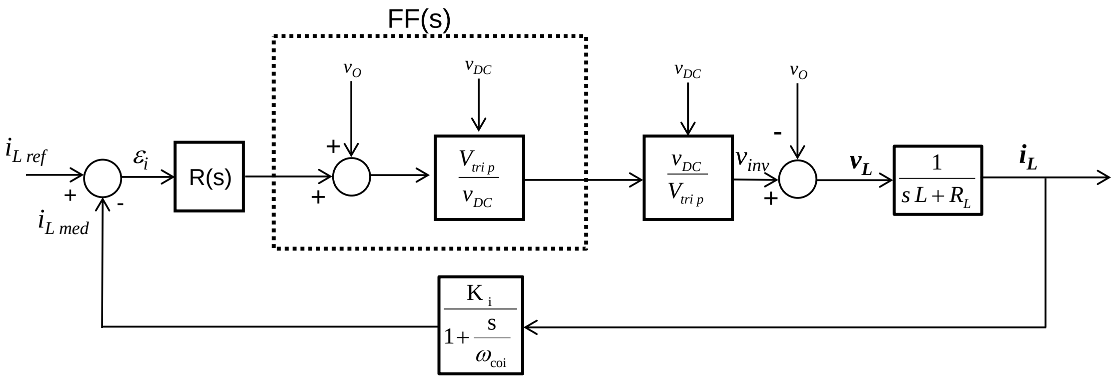

Figure 5: Block diagram of the inner current control loop with feed-forward of vo and vDC

As a consequence, for the purpose of the inner current loop compensator, the block diagram is reduced to the one shown in Figure 6. In Figure 6, it is clear that the plant transfer function for the purpose of the current control loop compensator design is reduced to the inductor admittance and it is independent of the inverter input voltage (v_{DC}) and the output voltage (v_o), see Figure 5.

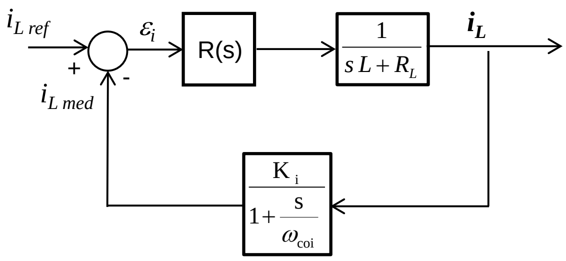

Figure 6: Block diagram of the current control loop with FF

So, let's design the current control loop compensator.

2.2 Design of the current control loop compensator

Since the considered topology is not included within the pre-defined ones in SmartCtrl then, the custom definition of the system is needed.

Before starting the design process, the user can open the "text code" of the voltage source inverter and have a look at the typical structure (it is not mandatory) and syntax of a text file that contains the models for plants, sensors, loop gains, etc.

To do so, please go to the Tools Menu and click on Equation Editor. Then click on "open" button and browse the file "Single Phase VSC.tromod".

Note that, although any text editor can be used to develop the text contained in the text code, the extension ".tromod" must be used in order to guarantee that SmartCtrl recognizes the text file as a model file.

When the desired file is selected, the window shown on Figure 7 is open. Then press on equation editor, and the text code on ANNEX A can be edited.

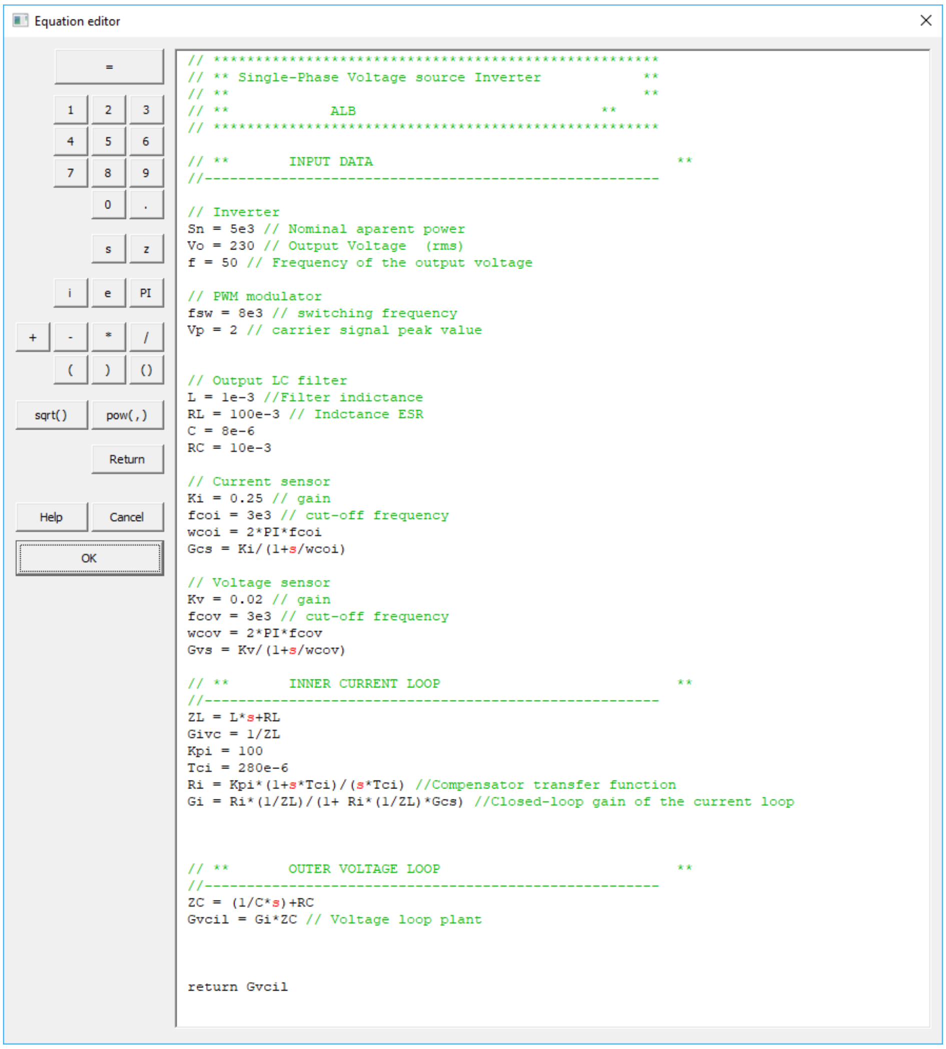

Figure 7: Equation Editor Edition environment

Once the user is familiarized with the text code structure and syntax, the design process can start. Remember that the steady-state values, plant and sensor transfer functions for both loops can be stored in the same “text code”, and the user only has to send to the design environment the transfer function he needs for the design of each particular loop.

Going back to the design of the current control loop, the “design a generic control system” option is going to be used (see Figure 8). Within this option, an equation editor is the tool provided for the definition of both the plant and the sensor.

Starting with the plant definition, the procedure is summarized in Figure 9. After clicking on compile, the transfer function is stored as the plant of the generic control system.

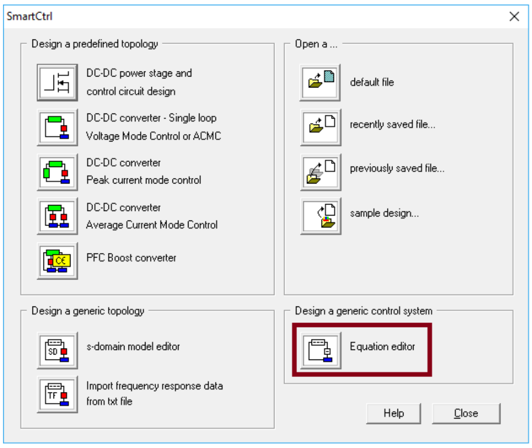

Figure 8: Select the "Design a generic control system option"

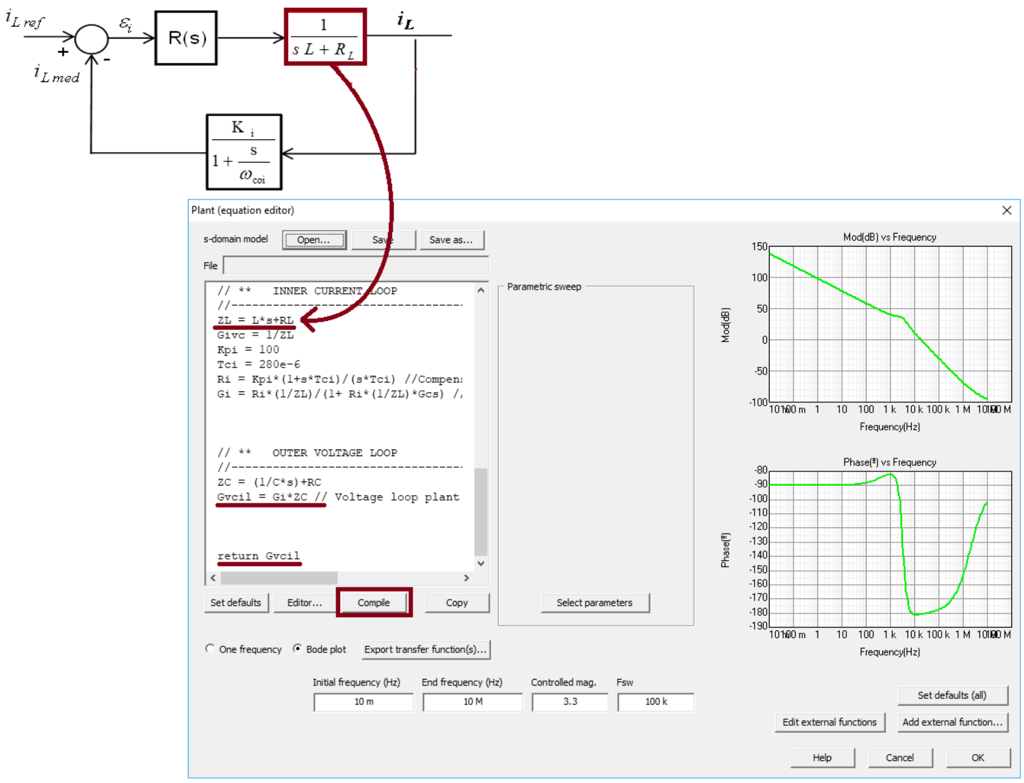

Figure 9: : Definition of the current control loop plant

Afterwards, the sensor transfer function must be defined in the same way. The process is also summarized in Figure 10. After completing the sensor definition, click OK and SmartCtrl will show the window in Figure 11 to continue with the control system definition.

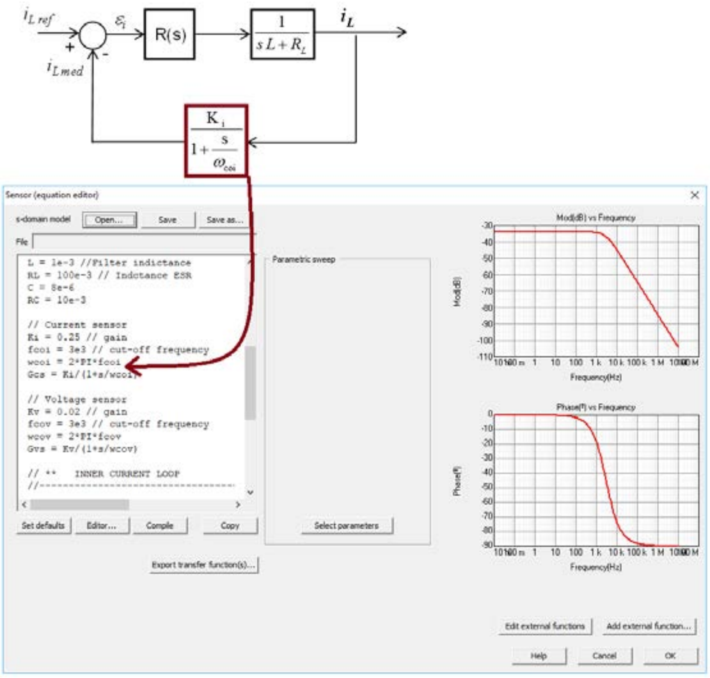
Figure 10: Definition of the current sensor

Once the plant and the current sensor are defined, then the compensator type must be selected. In this case, it is a PI compensator as shown in Figure 11.

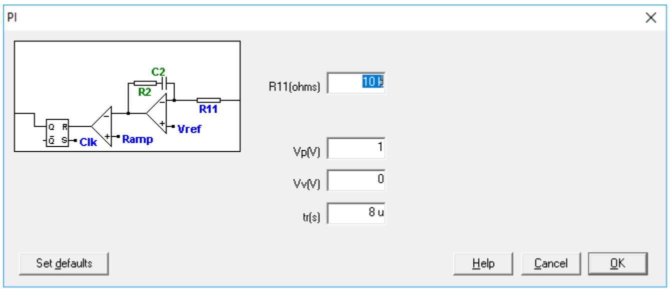

Figure 11: PI compensator parameters

Then the program is able to show the solution map that provides the sets of phase margin-crossover frequency that led to stable solutions. This solution map provides an easy tool for the selection of the control loop initial solution, for instance fc=2.57kHz and PM=37.5º (see Figure 12), that can be optimized later.

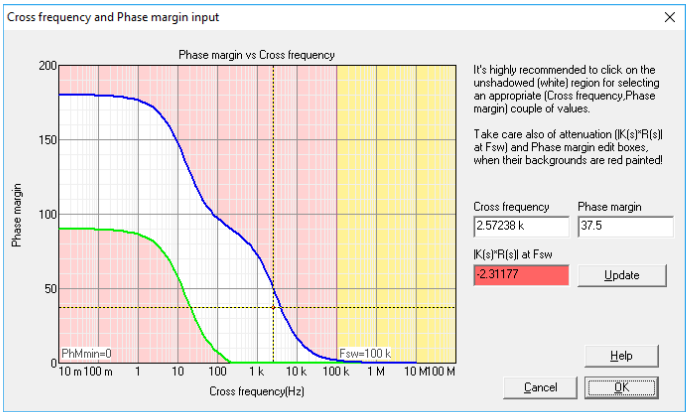

Figure 12: Compensator selection and solution map (fci=2.57kHz, PMi=37.5º)

Click OK twice to proceed, and the graphic and text panels will show the Bode plots, Nyquist plot and transient response, as well as the solution map, in order to help the optimization of the control loop design. An optimized compensator seeks to fulfil the following guidelines:

- Try to obtain the maximum open loop gain at any frequency. |T(s)| must perform the highest value always. In that way, the effect of the perturbations to the control system will be minimized.
- The maximum crossover frequency (fc) is limited by the effective cut-off frequency of the current loop. Dynamical interaction can occur if outer loop is faster than inner loop. It is preferable to set below fci. In this way, the current loop is a constant within the bandwidth of the VSC loop
- An overdamped response is in general preferable. Damping factor is increasing as PM increases. However, if a very high PM is selected, it would penalize the overall gain of T(s)

The optimization of the compensator performance can be easily carried out with the help of SmartCtrl, since the transient response and the Bode and Nyquist plots can be checked simultaneously at a glance. Additionally, on the right hand side of the window, sliders for the crossover frequency and the phase margin are available, as well as the solution map. This last one provides the designer with a powerful tool, since it shows the feasible solutions space for a given plant and compensator type in a graphical and straightforward way. In Figure 13 it can be seen that the initial solution (fc=1kHz, PM=60º) can be improved and better features can be obtained. For instance, for fc=2kHz and PM=45º, a higher control to output gain is achieved.

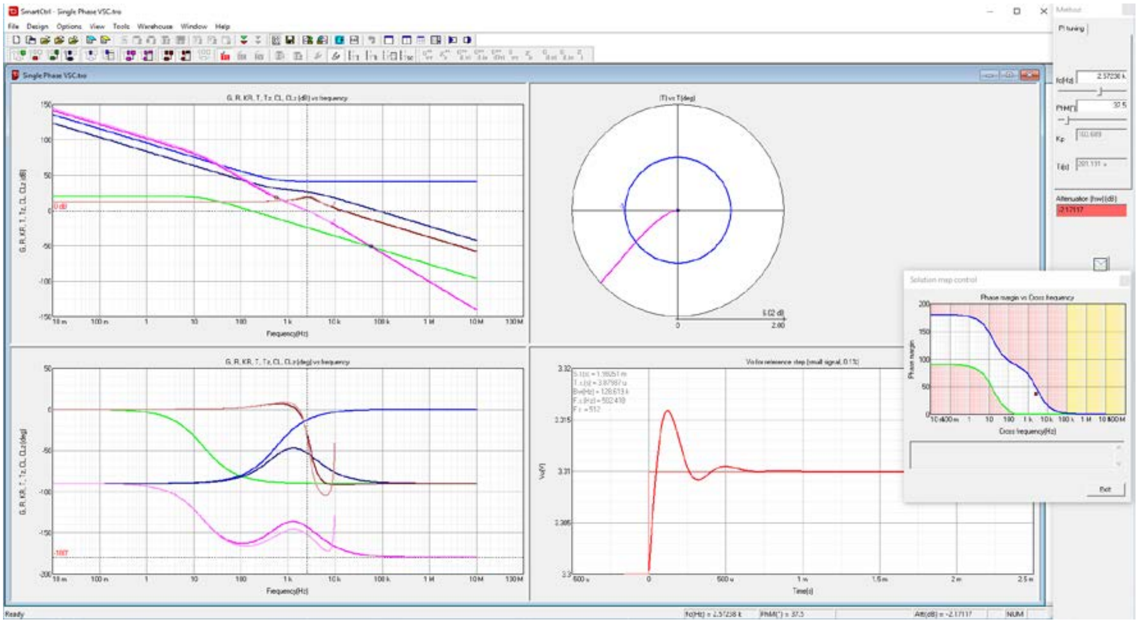

Figure 13: Graphic panels from SmartCtrl

For each design, checking the output data panel, the designer will find the resulting compensator given in different formats as depicted in Figure 14.

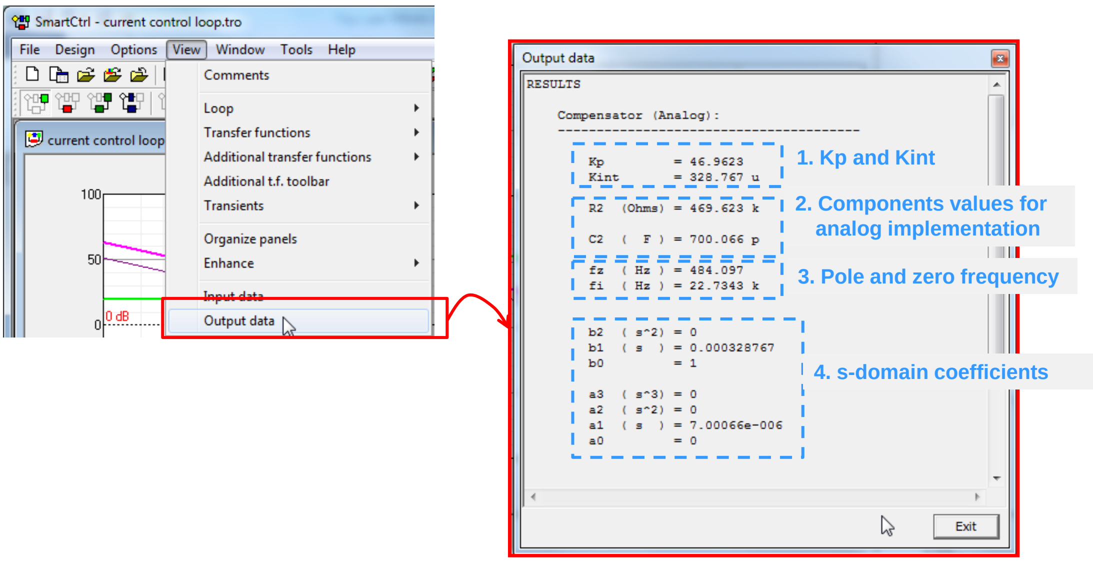

Figure 14: Output data panel for the current control loop

Regarding the digital implementation, it is also available in SmartCtrl. As summarized in Figure 15, through the digital control option and after defining the sampling frequency, bits number and accumulated delay, the effect of the digital implementation can be represented along with the analog Bode plots. Additionally, the compensator Z-domain coefficients can be found in the output data panel.

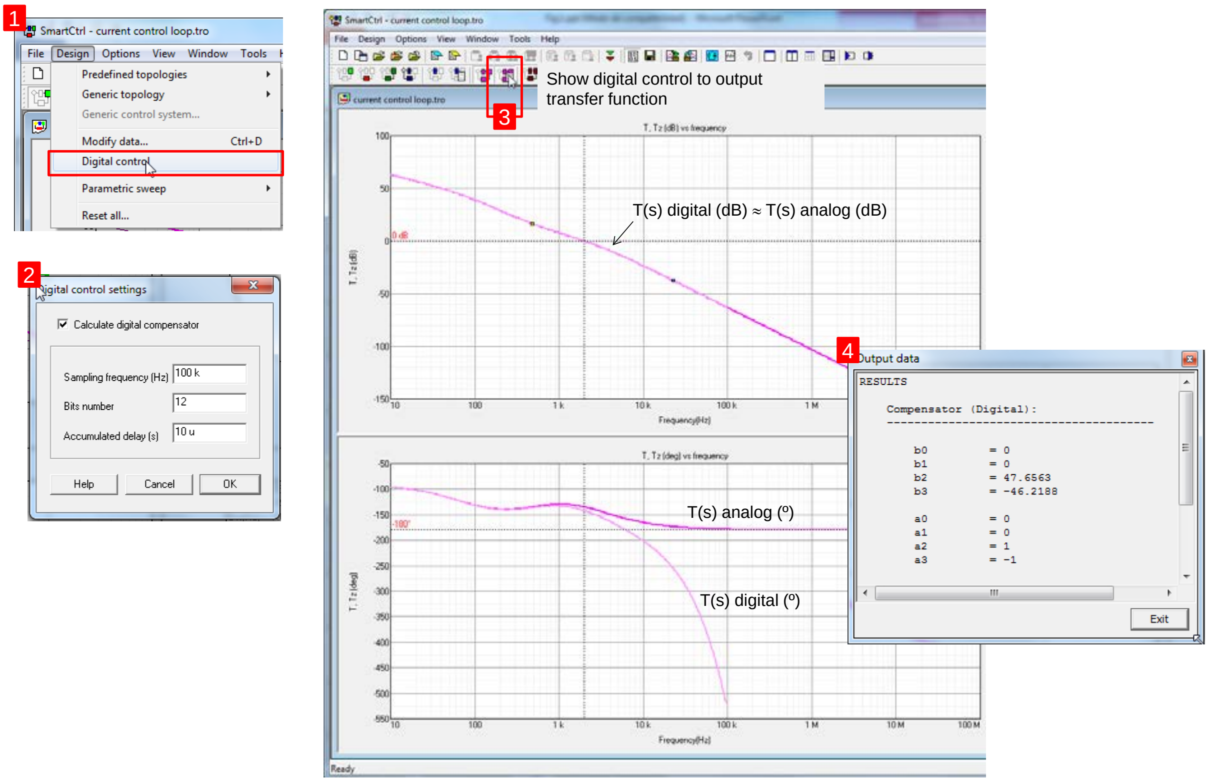

Figure 15: Represent the digital effects and obtain the compensator coefficients

Right now, the current control loop is already designed. So, it is time to begin with the design of the voltage control loop.

3. The design of the voltage control loop

Since the control structure is formed by two nested control loops, the outer voltage loop provides the reference to the inner current loop, which behaves as a controlled current source as depicted in Figure 16.

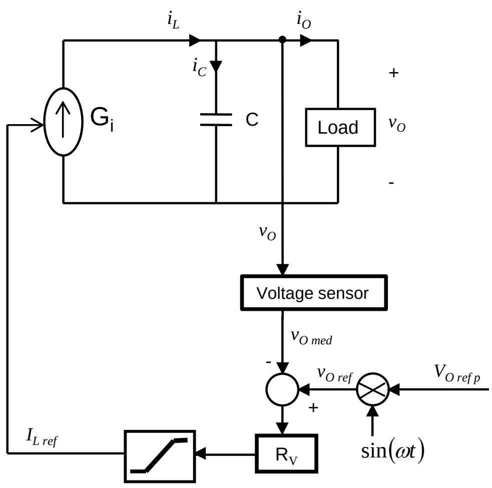

Figure 16: Voltage control loop and current control loop behaviour

Therefore, the output voltage v_o corresponds to equation [9].

$$v_o = (i_L - i_o)\left(\frac{1}{sC} + R_C\right) \qquad [9]$$

Where i_L is given by [10] and the closed loop transfer function of the current control loop (Gi) is expressed in [11].

$$i_L = G_i(s) \cdot i_{L\_ref} \qquad [10]$$

$$G_i(s) = \frac{R_i \cdot \frac{1}{Z_L(s)}}{1 + R_i \cdot \frac{1}{Z_L(s)} \cdot Gcs(s)} \qquad [11]$$

Where R_i is the current regulator that was calculated previously.

Finally, the voltage sensor behavior is analogous to the one of the current sensor, and is given in [12].

$$v_{o\,med} = \frac{K_v}{1 + \frac{s}{\omega_{cov}}} v_o = G_{vs}(s) \cdot v_o \qquad [12]$$

So, the block diagram of the voltage control loop is the one shown in Figure 17.

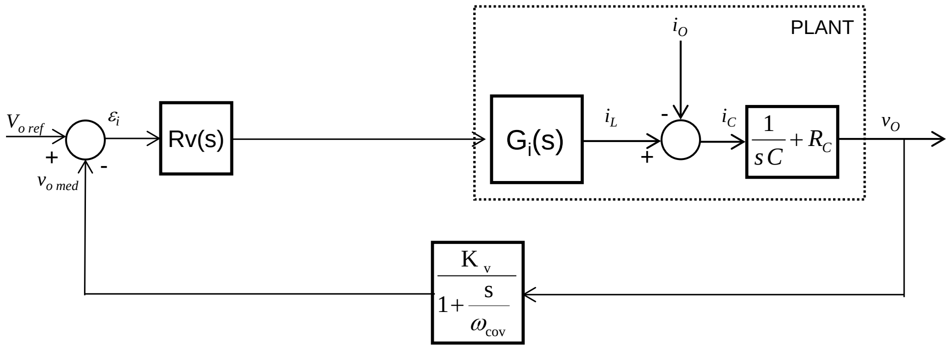

Figure 17: Voltage control loop

As in the case of the current control loop, it would be useful if the plant could be reduced to the output capacitor impedance. This is, by means of a feed-forward technique, achieve a plant independent of the current control loop (Gi(s)) and the load current (i_o).

Following an analogous procedure to the one explained for the current control loop, the block diagram of the voltage control loop when a FF is implemented would be the one in Figure 18.

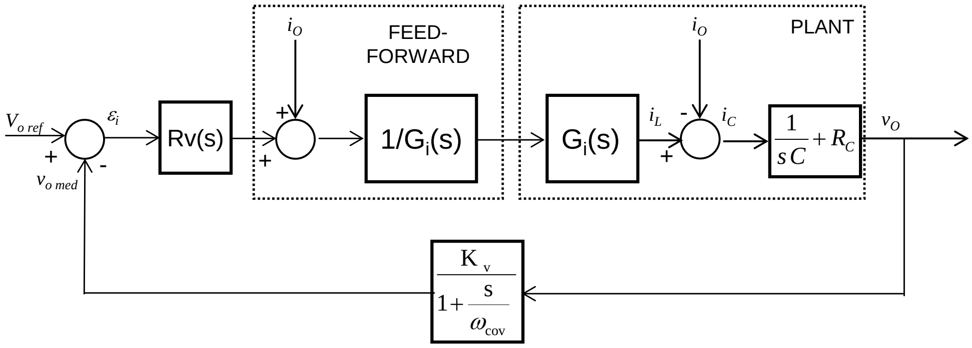

Figure 18: Voltage loop with FF

However, while adding i_o is easy, the implementation of 1/Gi(s) in the FF path is difficult, since it is a complex transfer function. That being said, if the crossover frequency of the voltage control loop is low enough, then the closed loop transfer function can be approximated as a constant [13]. And so, the implementation of Gi(s) is now very easy.

$$G_i(s) \approx \frac{1}{K_i} \qquad [13]$$

Where: K_i is the current sensor constant.

Thus, in order to be able to implement the feed-forward, a new design constraint must be kept in mind: "the selected crossover frequency of the voltage control loop must be low enough so that the closed loop current control transfer function behaves as a constant". The way in which this design constraint is taken into account will be explained later.

3.1 How to obtain and export Gi(s)

Prior to the design of the voltage control loop, the closed loop frequency response of the current control loop must be obtained and exported. In order to be used later to account for the voltage loop crossover limitation that allows the implementation of the feed-forward.

As summarized in Figure 19, to obtain the closed loop transfer function the first step is to type the obtained PI constants in order to calculate the closed loop frequency response. Next, it is needed de definition of which function must be returned, and finally click on compile to calculate the selected function.

As it can be seen in Figure 19, on the right hand side of the window the Bode plots for the considered transfer function are shown.

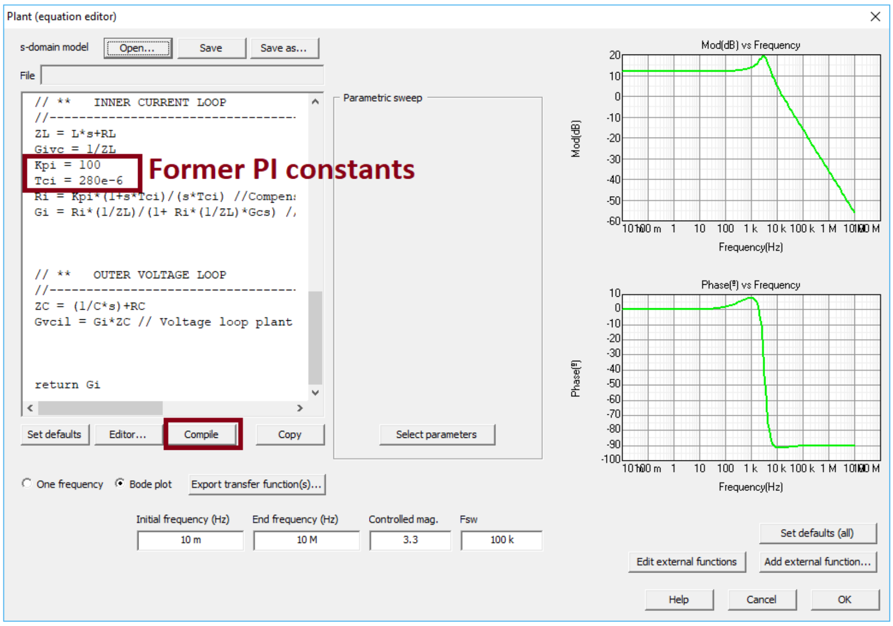

Figure 19: Summary on how to obtain the current closed loop transfer function

Right afterwards the frequency response can be exported for its later use, as depicted in Figure 20.

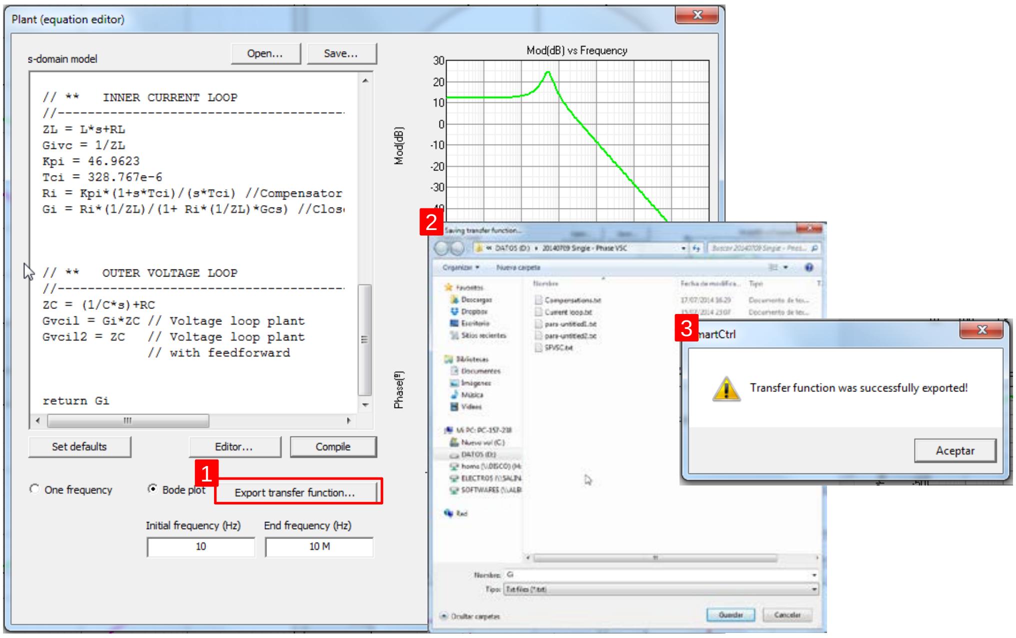

Figure 20: Summary on how to export a transfer function after compiling it

3.2 Design of the voltage control loop compensator

Assuming that the feed-forward is implemented, then for the purpose of the voltage loop compensator, the block diagram is reduced to the one shown in Figure 21. As it can be seen, the plant is reduced to the output capacitor impedance, and it is independent from the closed current loop and from the load current.

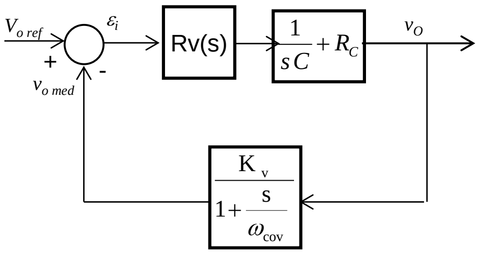

Figure 21: voltage control loop with FF

So, the first step is the definition of the plant, as summarized in Figure 22, and the definition of the sensor in Figure 23.

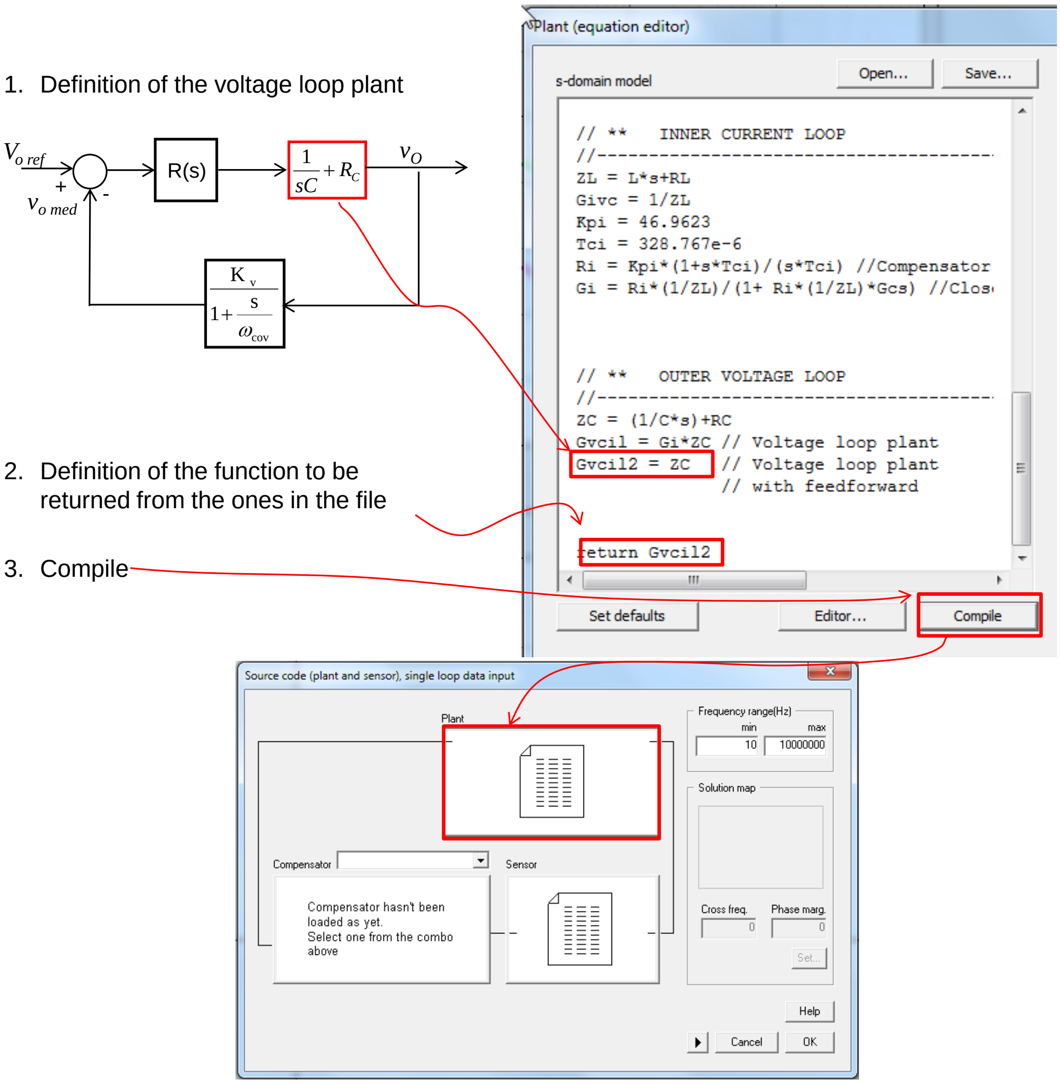

Figure 22: Definition of the voltage loop plant

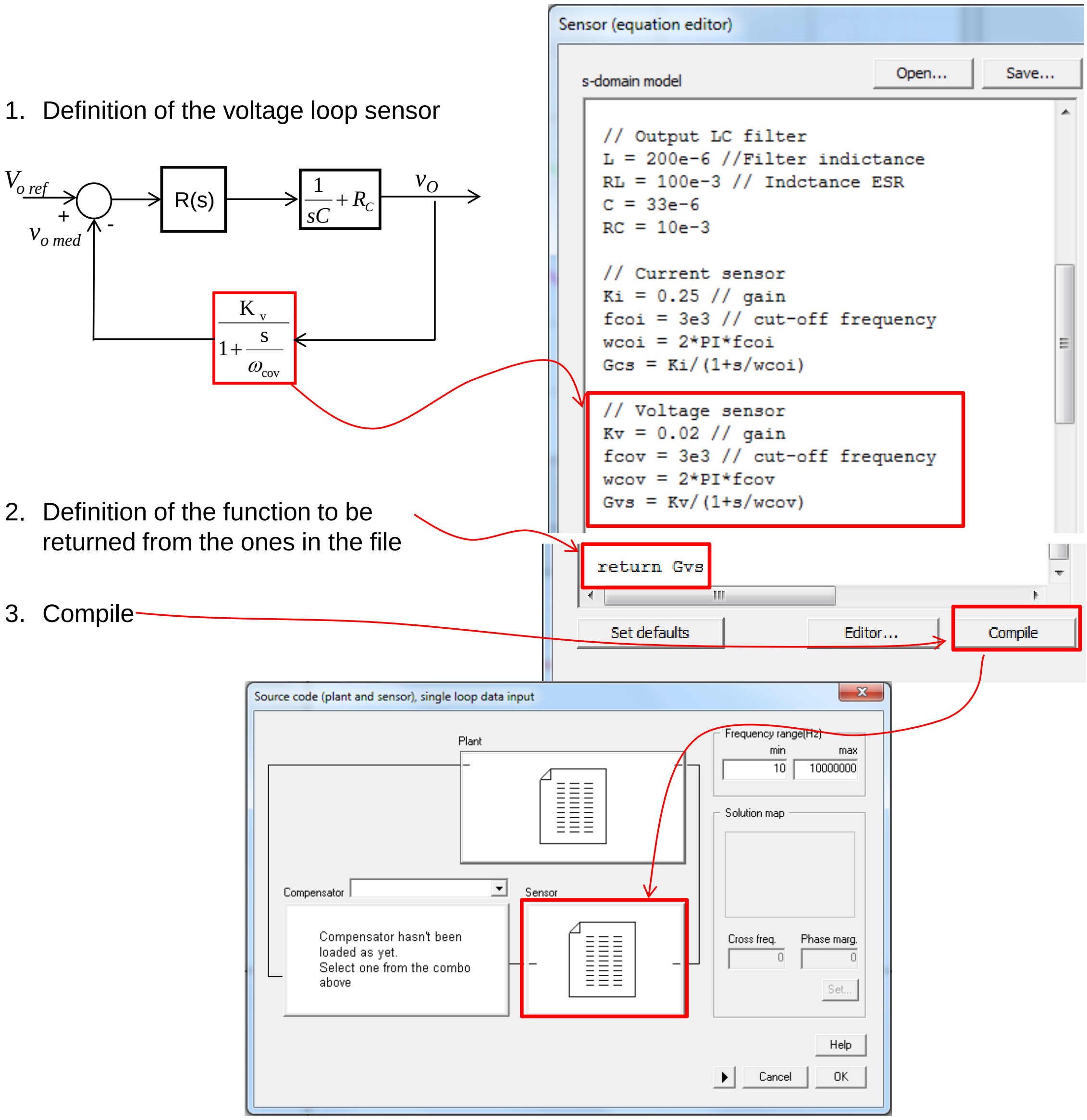

Figure 23: Definition of the voltage loop sensor

And finally, select the compensator type and use the solution map to stablish the crossover frequency and the phase margin (see Figure 24)

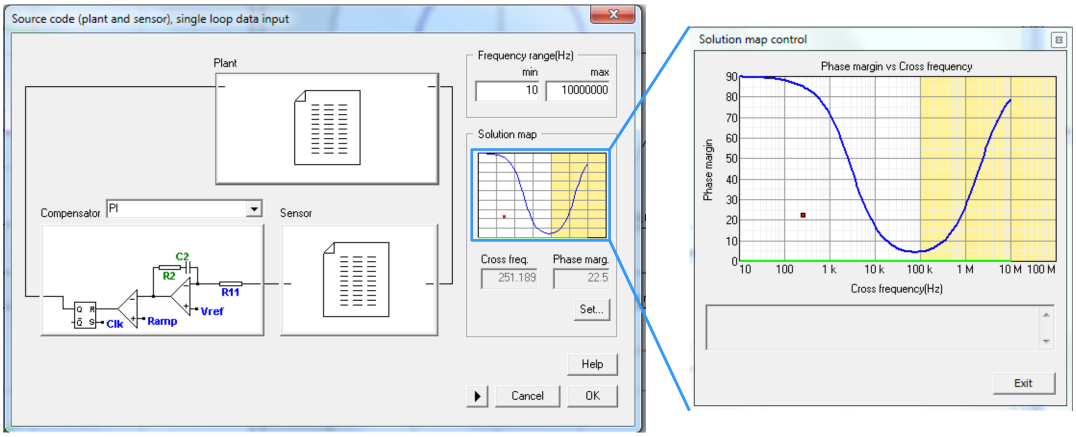

Figure 24: Select a compensator fc, PM for voltage loop

Be aware that, in this case, the modulator must be set to have unity gain, since there is no modulator in the outer control loop. Once the system is defined, the results panels are displayed and the regulator can be calculated.

It should be reminded that there is an additional design constraint regarding the maximum crossover frequency in order to guarantee that the feed-forward works. In order to keep this restriction in mind while designing the voltage control loop PI, it is recommended to include the frequency response of the current closed loop in the Bode plots. This frequency response was previously exported, and now it is going to be imported to be represented together with the Bode Plots of the voltage control loop.

The process is summarized in Figure 25.

Following the same export/import process, the control to output frequency response of the current control loop is also included for additional information during the voltage control loop design.

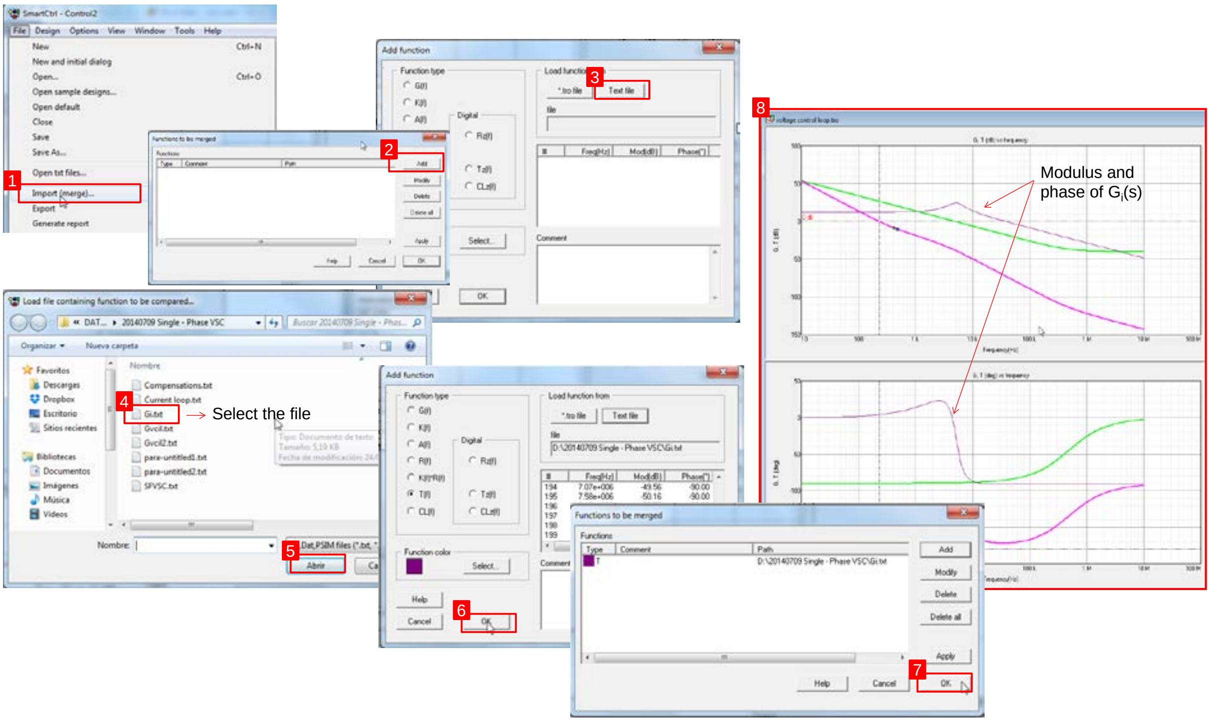

Figure 25: Import the current closed loop frequency response

The imported closed loop frequency response provides information related to the behavior of Gi(s) (in order to predict the FF performance) and the control to output of the current control loop establishes the maximum crossover frequency of the voltage control loop to avoid dynamic interaction between the two nested loops. (see Figure 26)

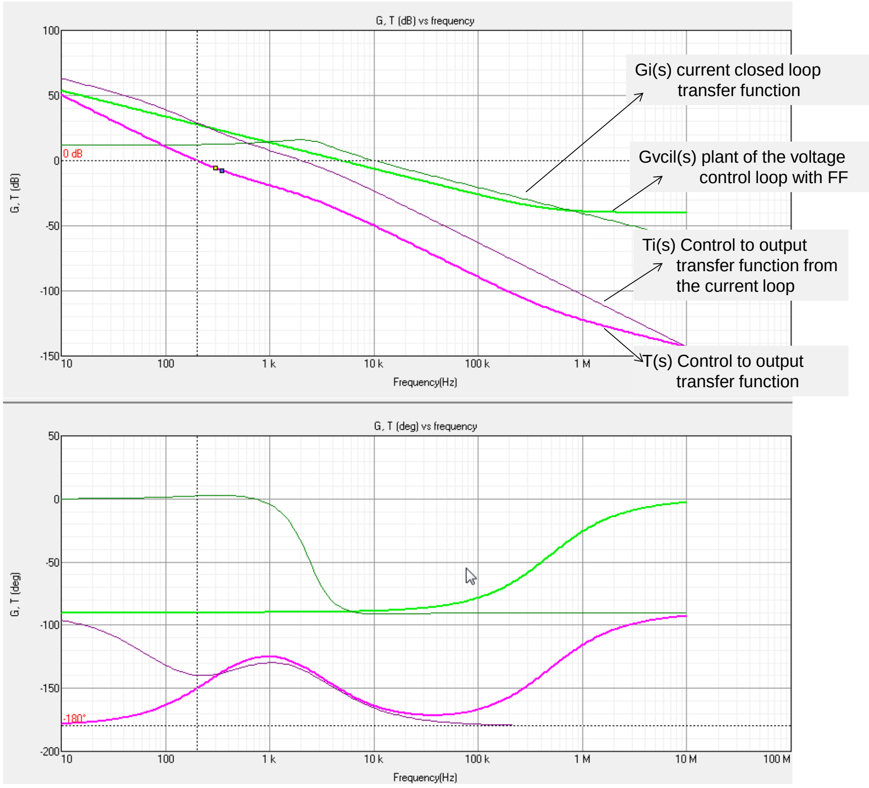

Figure 26: Bode plots for the design of the voltage control loop compensator

As stated before, the outer loop must be slower than the inner one, in order to avoid dynamic interaction between the loops and thus instability. This is so, because the plant of the inner loop can be assumed to be almost constant within the bandwidth of the voltage control loop. Therefore, in this situation the assumption made while designing the feed-forward is also valid ($G_i(s) \approx 1/Ki$) and it will be able to eliminate the influences of the current closed loop.

So, it can be said that the maximum crossover frequency for the voltage control loop is limited by the crossover frequency of the current control loop.

In order to illustrate this effect, let´s consider the schematic shown in Figure 20 and the simulation results provided in Figure 28, Figure 29 and Figure 30. In Figure 28 it can be observed that the output voltage follows the reference without any error and it is able to attend to either input voltage or load current steps almost instantly, even though the crossover frequency of the voltage control loop is only 500 Hz.

In Figure 29, it is shown that even with a voltage loop crossover frequency equal to the one in the current loop, the system performances are still stable. However, when a higher crossover frequency is selected, it is shown in Figure 30 that both loops interact and tend to the instability.

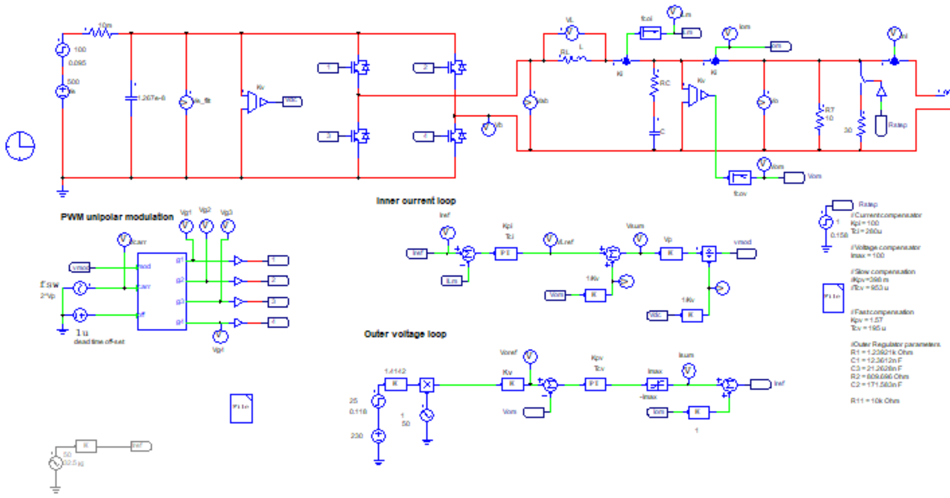

Figure 27: Simulation schematic with input voltage step and load current step

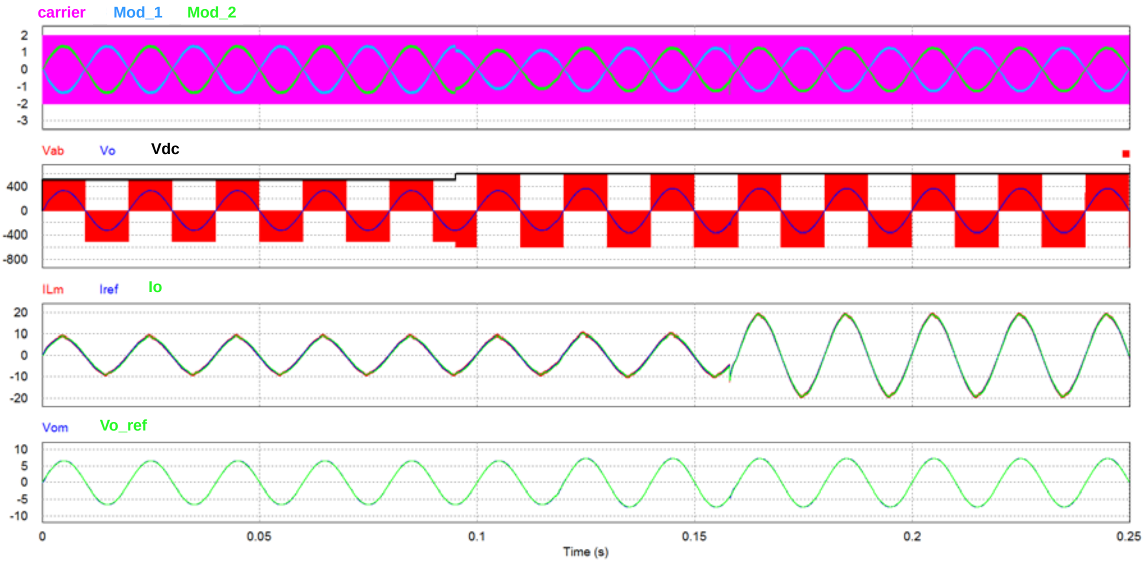

Figure 28: Simulation results with FF, fc=500Hz and PM=30º

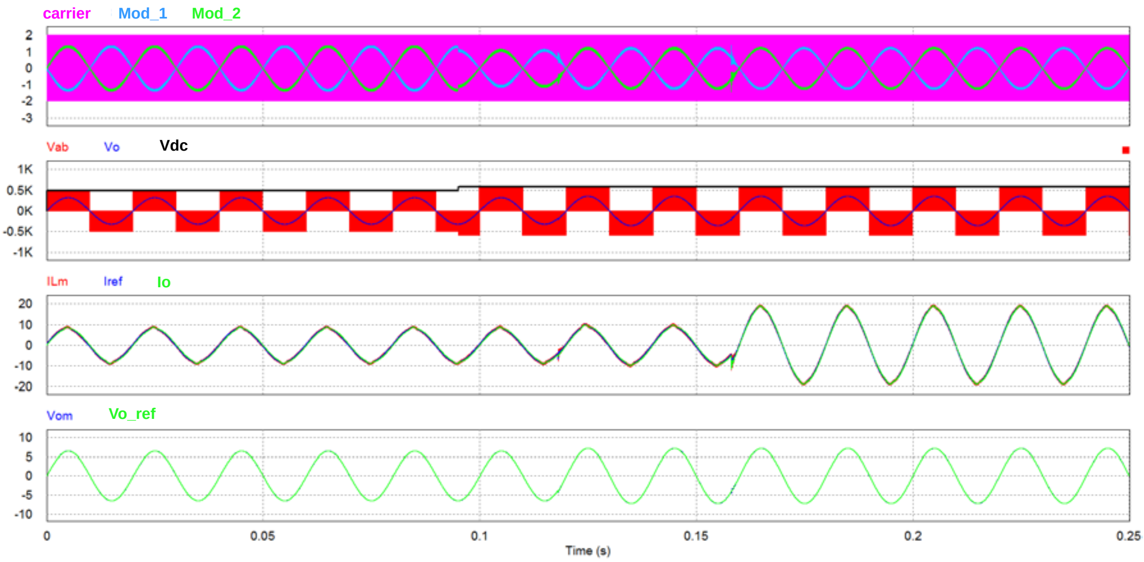

Figure 29: Simulation results with FF, fc=2k and PM=30º

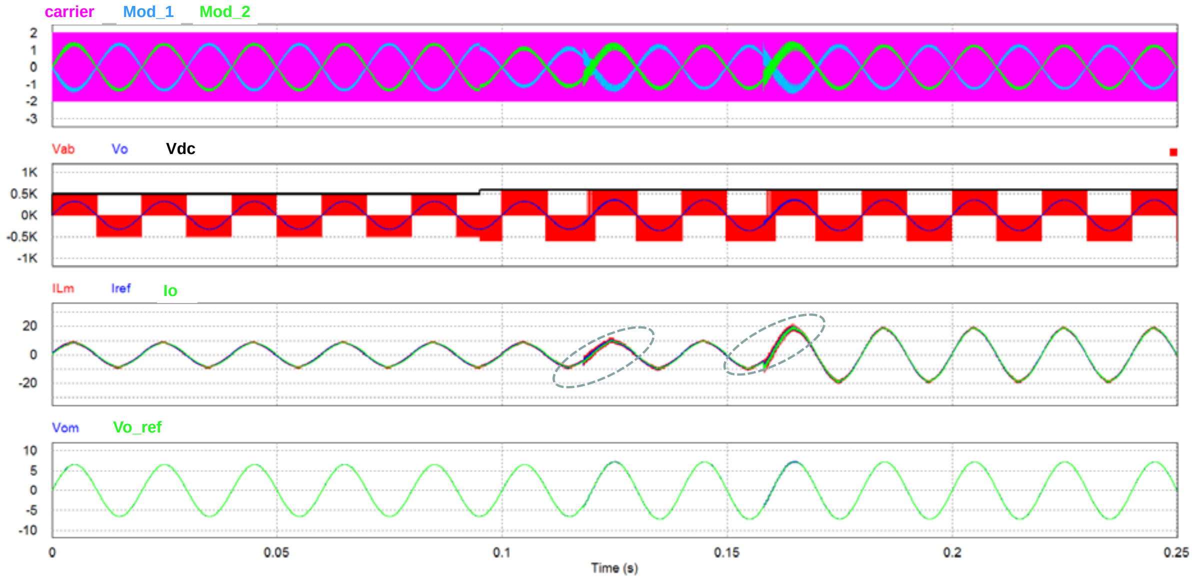

Figure 30: Simulation results with FF, fc=2.3kHz and PM=30º

At this point, the procedure to design the current and the voltage control loops of a single phase voltage source inverter with SmartCtrl has been completed.

It is noticeable that, given the model equations (plant and sensor) of any system, SmarCtrl provides a very fast and efficient tool to accurately design any control loop.

ANNEX A: “Text code” to design the double-loop control of a single-phase voltage-source inverter

```
//*****************************************************
// ** Single-Phase Voltage source Inverter           **
// **                                                **
// ** A. Lazaro 20140701                             **
// ****************************************************

// **  INPUT DATA                                   **
//----------------------------------------------------

// Inverter
Sn = 5e3 // Nominal apparent power
Vo = 230 // Output Voltage  (rms)
f = 50 // Frequency of the output voltage

// PWM modulator
fsw = 10e3   // switching frequency
Vp = 2       // carrier signal peak value
Vpp = 2*Vp   //carrier signal peak to peak value
Gmod = 1/Vpp // modulator gain


// Output LC filter
L = 200e-6 //Filter inductance
RL = 100e-3 // Inductance ESR
C = 33e-6
RC = 10e-3

// Current sensor
Ki = 0.25 // gain
fcoi = 3e3 // cut-off frequency
wcoi = 2*PI*fcoi
Gcs = Ki/(1+s/wcoi)

// Voltage sensor
Kv = 0.02 // gain
fcov = 3e3 // cut-off frequency
wcov = 2*PI*fcov
Gvs = Kv/(1+s/wcov)

// **  INNER CURRENT LOOP                           **
//----------------------------------------------------
ZL = L*s+RL
Givc = 1/ZL
Kpi = 46.9623
Tci = 328.767e-6
Ri = Kpi*(1+s*Tci)/(s*Tci) //Compensator transfer function
Ti = (1/ZL)*Ri*Gcs*Gmod //Control to output transfer function of the current
loop
Gi = (1/ZL)*Ri*Gmod /(1+Ti) //Closed-loop gain of the current loop


// **  OUTER VOLTAGE LOOP                           **
//----------------------------------------------------
ZC = (1/C*s)+RC
Gvcil = Gi*ZC // Voltage loop plant without io feedforward
Gvcil2 = ZC   // Voltage loop plant with io feedforward

return Gvcil2
```