On the Dynamical Modelling of DC-DC Converters

Farzin Asadi

Abstract— **Computers play an important role in analyzing and designing of modern DC-DC power converters. This tutorial paper shows how the widely used analysis techniques of averaging and linearization can be applied to DC-DC converters with the aid of computers. Obtained dynamical equations may then be used for control design.**

This paper composed of two parts. First part shows how ready to use software packages (such as PLECS®) and MATLAB® programming can be used to obtain the converter dynamics. Second part of the paper introduces a user friendly MATLAB toolbox to extract the DC-DC converter dynamics. Developed toolbox can be used for educational and/or industrial purposes.

This paper can be a good reference for researchers involved in DC-DC converters dynamics and control.

Index Terms— **DC-DC converter dynamics, dynamics of Zeta converter, MATLAB, PLECS, small signal model, state space averaging (SSA).**

I. Introduction

SWITCH MODE converters are widely used today to provide power processing for applications ranging from computing and communications to medical electronics, appliance control, transportation, and high-power transmission. Their high efficiency, small size, low weight and reduced cost, make them a good alternative for conventional linear power supplies, even at low power levels.

Switched DC-DC converters are non-linear variable structure systems. The nonlinearities arise primarily due to switching, power devices, and passive components, such as inductors, and parasitic. Various techniques can be found in literature to obtain a Linear Time Invariant (LTI) model of a switched DC-DC converter. The most well-known methods are [1]: Circuit averaging (CA) and State Space Averaging (SSA).

CA replaces the semiconductor switches (non-linear part of the converter) with an (averaged) equivalent linear circuit. In this method manipulations are carried out based on a circuit diagram [2].

SSA, uses the duty cycle as a weighting factor and combines the state equations into a single averaged state equation. The procedure of state space averaging is explained in detail in [3] and [4].

SSA has a number of advantages over circuit averaging technique. These includes:

- Ability to obtaint more transfer functions than was possible using circuit averaging technique.
- Both DC and AC transfer functions are obtained with more ease.

Foundation of State Space Averaging (SSA) was laid down in [5] and later extended in [6-8], as well as many other publications. The first attempt to model Discontinuous Conduction Mode (DCM) is presented in [9]. Accurate small signal models for DCM operation were developed by Sun J et al. (2001). A unified SSA based method to develop both Continuous Current Mode (CCM) and DCM was developed by [10]. A comprehensive survey of the modeling issues can be found in [11].

This paper presents a tutorial exposition of modelling DC-DC converters using computer tools. Extraction of converters dynamics (except for the simple second order converters) using pencil-and-paper is a difficult and error prone task.

Both ready to use software packages (PLECS) and MATLAB programming are used to show the process of DC-DC converter modelling. The paper also introduces a user friendly MATLAB toolbox named KUCA (Kocaeli University Converter Analysis suit). Developed toolbox can be used to extract the small signal transfer functions of well-known DC-DC converters. Current version of this toolbox supports the buck, boost, buck-boost, Cuk, SEPIC, fly back, forward and full bridge topologies. Developed toolbox can be used for educational and industrial purposes.

This paper is organized as follows: MATLAB implementation of SSA is studied in the second section. A buck converter and a Zeta converter are studied in this section. Third section introduces the developed toolbox. Finally, suitable conclusions are drawn.

II. Dynamics of CCM Converters

In this section we study the dynamics of DC-DC converters operating in Continuous Current Mode (CCM). State Space Averaging (SSA) is one of the most important tools to study the dynamics of converters operating in CCM. SSA has two important steps: averaging and linearization. The SSA procedure can be summarized as follows [3]:

FARZIN ASADI, is with Department of Electrical and Electronics Engineering, Maltepe University, Istanbul, Turkey, (e-mail: farzinasadi@maltepe.edu.tr).
https://orcid.org/0000-0002-5928-0807

Manuscript received June 09, 2021; accepted October 24, 2021.
DOI: 10.17694/bajece.949956

1- Circuit differential equations are written for different working modes (i.e on/off state of semiconductor switches).
2- Equations are time averaged over one period.
3- Steady state operating points are calculated by equating the derivative terms to zero.
4- The averaged equations are linearized around the steady state operating point found in the third step.
Applying this procedure is quite tedious and error prone for pencil-and-paper analysis (especially if the converter order is high). MATLAB® can be very helpful to do the mathematical machinery of SSA. We show the usefulness of MATLAB to extract the converter small signal transfer functions with two examples: A buck converter and a Zeta converter.

Schematic of a buck converter is shown in Figure 1. The buck converter composed of two switches: a MOSFET switch and a diode. In this schematic, Vg, rg, L, rL, C, rC and R shows input DC source, input DC source internal resistance, inductor, inductor Equivalent Series Resistance(ESR), capacitor, capacitor ESR and load, respectively. iO is a fictitious current source added to the schematic in order to calculate the output impedance of converter. In this section we assume that converter works in Continuous Current Mode(CCM). MOSFET switch is controlled with the aid of a Pulse Width Modulator (PWM) controller. MOSFET switch keeps closed for $D.T$ seconds and $(1-D).T$ seconds open. D and T show duty ratio and switching period, respectively.

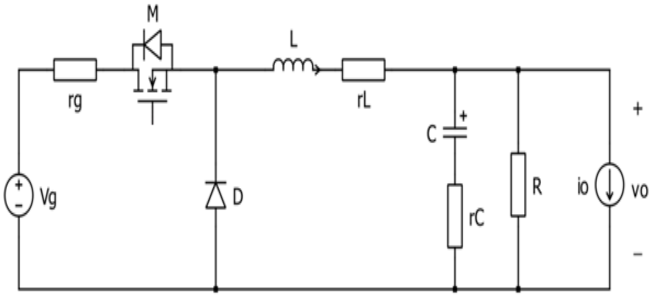


Fig. 1. Schematic of a buck converter.

When MOSFET is closed, the diode is opened (Figure 2).

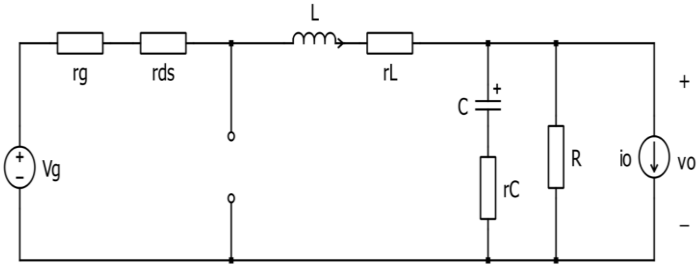


Fig. 2. Equivalent circuit of a buck converter for closed MOSFET.

The circuit differential equations can be written as:

$$\begin{cases} \dfrac{di_L(t)}{dt} = \dfrac{1}{L}\left(-\left(r_g + r_{ds} + r_L + \dfrac{R \times r_C}{R + r_C}\right) i_L - \dfrac{R}{R + r_C} v_C + \dfrac{R \times r_C}{R + r_C} i_O + v_g\right) & (1) \\ \dfrac{dv_C(t)}{dt} = \dfrac{1}{C}\left(\dfrac{R}{R + r_C} i_L - \dfrac{1}{R + r_C} v_C - \dfrac{R}{R + r_C} i_O\right) & (2) \end{cases}$$

$$v_o = r_C C \frac{dv_C}{dt} + v_C = \frac{R \times r_C}{R + r_C} i_L + \frac{R}{R + r_C} v_C - \frac{R \times r_C}{R + r_C} i_O \qquad (3)$$

When MOSFET is opened, the diode is closed (Figure 3).

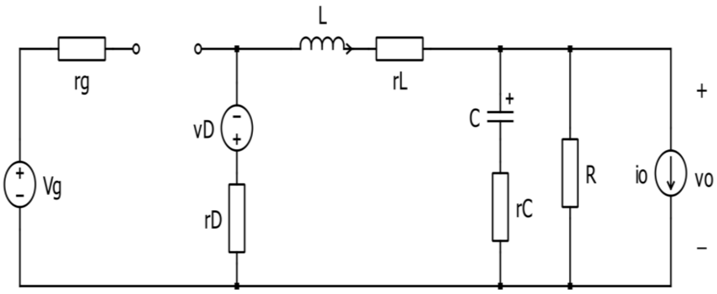


Fig. 3. Equivalent circuit of a buck converter for opened MOSFET.

The circuit differential equations can be written as:

$$\begin{cases} \dfrac{di_L(t)}{dt} = \dfrac{1}{L}\left(-\left(r_D + r_L + \dfrac{R \times r_C}{R + r_C}\right) i_L - \dfrac{R}{R + r_C} v_C + \dfrac{R \times r_C}{R + r_C} i_O - v_D\right) & (4) \\ \dfrac{dv_C(t)}{dt} = \dfrac{1}{C}\left(\dfrac{R}{R + r_C} i_L - \dfrac{1}{R + r_C} v_C - \dfrac{R}{R + r_C} i_O\right) & (5) \end{cases}$$

$$v_o = r_C C \frac{dv_C}{dt} + v_C = \frac{R \times r_C}{R + r_C} i_L + \frac{R}{R + r_C} v_C - \frac{R \times r_C}{R + r_C} i_O \qquad (6)$$

Consider a converter with the values given in Table 1.

TABLE 1.
THE BUCK CONVERTER PARAMETERS (SEE FIGURE 1)

	Nominal Value
Output voltage, Vo	20 V
Duty ratio, D	0.4
Input DC source voltage, Vg	50 V
Input DC source internal resistance, rg	0.5 Ω
MOSFET Drain-Source resistance, rds	40 mΩ
Capacitor, C	100 µF
Capacitor Equivalent Series Resistance(ESR), rC	0.05 Ω
Inductor, L	400 µH
Inductor ESR, rL	10 mΩ
Diode voltage drop, vD	0.7 V
Diode forward resistance, rD	10 mΩ
Load resistor, R	20 Ω
Switching Frequency, Fsw	20 KHz

The code given in appendix (Program 1), extracts the converter transfer functions. The code uses the SSA to calculate the transfer functions. Following results are obtained after running the code:

$$\frac{\tilde{v}_o(s)}{\tilde{i}_o(s)} = -0.0499 \frac{(s + 2 \times 10^5)(s + 580)}{s^2 + 1203s + 2.523 \times 10^7} \qquad (7)$$

$$\frac{\tilde{v}_o(s)}{\tilde{v}_g(s)} = 49.875 \frac{(s + 2 \times 10^5)}{s^2 + 1203s + 2.523 \times 10^7} \qquad (8)$$

$$\frac{\tilde{v}_o(s)}{\tilde{d}(s)} = 6257.7 \frac{s + 2 \times 10^5}{s^2 + 1203s + 2.523 \times 10^7} \qquad (9)$$

Bode diagram of these transfer functions are shown in Figure 4-6.

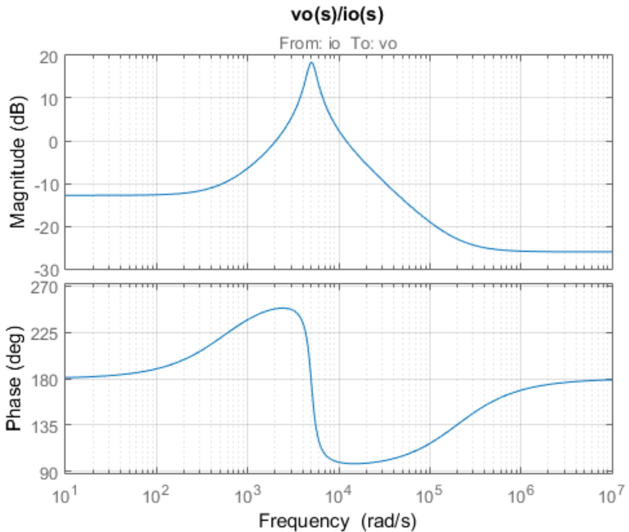


Fig 4. $\frac{\hat{v}_o(s)}{\hat{\iota}_o(s)} = -0.0499\,\frac{(s+2\times10^5)(s+580)}{s^2+1203s+2.523\times10^7}$ Bode diagram.

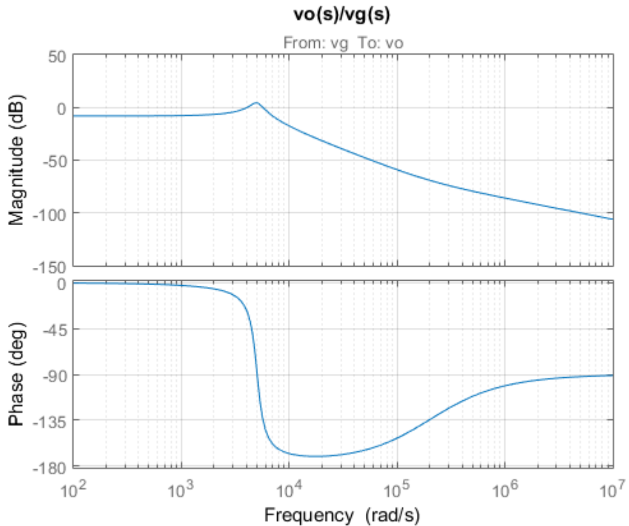


Fig 5. $\frac{\hat{v}_o(s)}{\hat{v}_g(s)} = 49.875\,\frac{(s+2\times10^5)}{s^2+1203s+2.523\times10^7}$ Bode diagram.

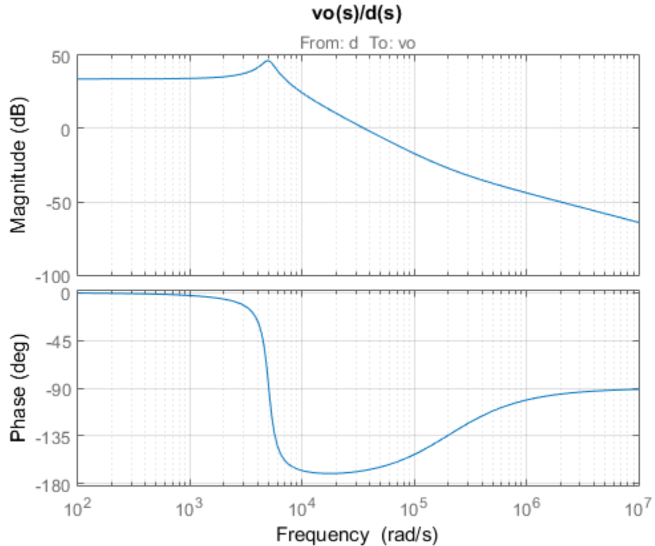


Fig. 6. $\frac{\hat{v}_o(s)}{\hat{d}(s)} = 6257.7\,\frac{s+2\times10^5}{s^2+1203s+2.523\times10^7}$ Bode diagram.

According to the analysis results, the converter with parameters given in Table 1, can be modelled as shown in Figure 7.

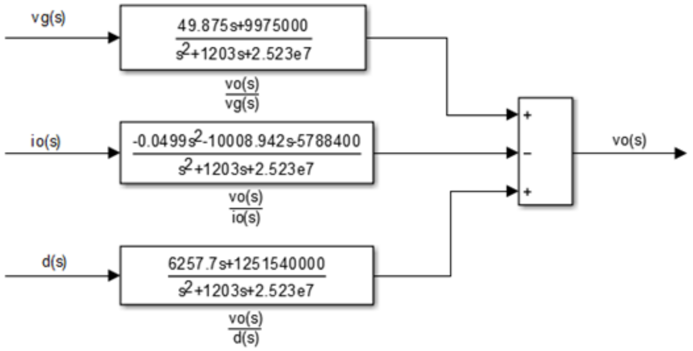


Fig. 7. Dynamic model of the converter.

Since the injected test current (io in Figure 1) does not enter the positive end of output voltage, the obtained transfer function for output impedance (Equation 7) must be multiplied by -1 to be converted to the correct form of output impedance.

III. DYNAMICS OF ZETA CONVERTER

Schematic of a Zeta converter is shown in Figure 8. The Zeta converter composed of two switches: a MOSFET switch and a diode. In this schematic, Vg, rg, Li, rLi, Ci, rCi and R shows input DC source, input DC source internal resistance, i^{th} inductor, i^{th} inductor Equivalent Series Resistance (ESR), i^{th} capacitor, i^{th} capacitor ESR and load, respectively. iO is a fictitious current source added to the schematic in order to calculate the output impedance of converter. In this section we assume that converter works in Continuous Current Mode (CCM). MOSFET switch is controlled with the aid of a Pulse Width Modulator (PWM) controller. MOSFET switch keeps closed for $D.T$ seconds and $(1-D).T$ seconds open. D and T show duty ratio and switching period, respectively.

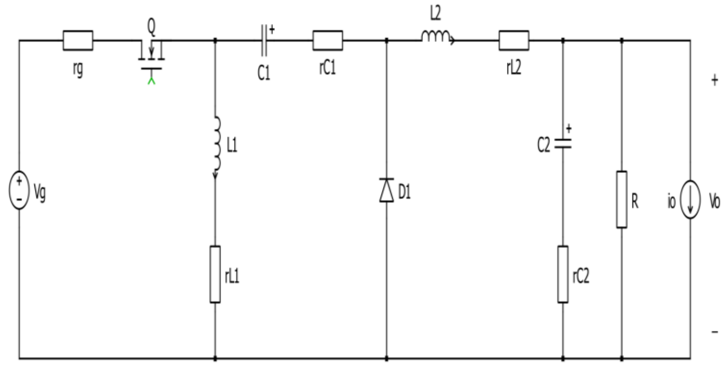


Fig. 8. Schematic of Zeta converter.

When MOSFET is closed, the diode is opened (Figure 9).

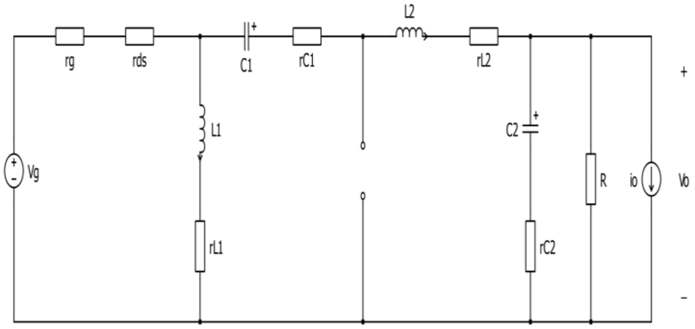


Fig. 9. Closed MOSFET.

The circuit differential equations can be written as:

$$\begin{cases} L_1\dfrac{di_{L_1}}{dt}=-\left(r_{L_1}+r_g+r_{ds}\right)i_{L_1}-\left(r_g+r_{ds}\right)i_{L_2}+v_g & (10)\\ L_2\dfrac{di_{L_2}}{dt}=-\left(r_g+r_{ds}\right)i_{L_1}-\left(r_g+r_{ds}+r_{C_1}+r_{L_2}+\dfrac{R\times r_{C_2}}{R+r_{C_2}}\right)i_{L_2}+v_{C_1}-\dfrac{R}{R+r_{C_2}}v_{C_2}+\dfrac{R\times r_{C_2}}{R+r_{C_2}}i_o+v_g & (11)\\ C_1\dfrac{dv_{C_1}}{dt}=-i_{L_2} & (12)\\ C_2\dfrac{dv_{C_2}}{dt}=\dfrac{R}{R+r_{C_2}}i_{L_2}-\dfrac{1}{R+r_{C_2}}v_{C_2}-\dfrac{R}{R+r_{C_2}}i_o & (13)\end{cases}$$

$$v_o=r_{C_2}C_2\frac{dv_{C_2}}{dt}+v_{C_2}=\frac{R\times r_{C_2}}{R+r_{C_2}}i_{L_2}+\frac{R}{R+r_{C_2}}v_{C_2}-\frac{R\times r_{C_2}}{R+r_{C_2}}i_o \qquad (14)$$

When MOSFET is opened, the diode is closed (Figure 10). In Figure 10, forward biased diode is modelled with a voltage source (VD) and a series resistance (rD).

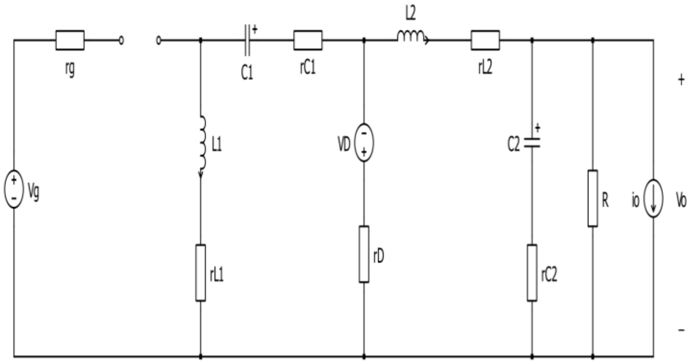


Fig. 10. Opened MOSFET.

The circuit differential equations can be written as:

$$\begin{cases} L_1\dfrac{di_{L_1}}{dt}=-\left(r_{L_1}+r_{C_1}+r_D\right)i_{L_1}-r_D i_{L_2}-v_{C_1}-v_D & (15)\\ L_2\dfrac{di_{L_2}}{dt}=-r_D i_{L_1}-\left(r_D+r_{L_2}+\dfrac{R\times r_{C_2}}{R+r_{C_2}}\right)i_{L_2}-\dfrac{R}{R+r_{C_2}}v_{C_2}+\dfrac{R\times r_{C_2}}{R+r_{C_2}}i_o-v_D & (16)\\ C_1\dfrac{dv_{C_1}}{dt}=i_{L_1} & (17)\\ C_2\dfrac{dv_{C_2}}{dt}=\dfrac{R}{R+r_{C_2}}i_{L_2}-\dfrac{1}{R+r_{C_2}}v_{C_2}-\dfrac{R}{R+r_{C_2}}i_o & (18)\end{cases}$$

$$v_o=r_{C_2}C_2\frac{dv_{C_2}}{dt}+v_{C_2}=\frac{R\times r_{C_2}}{R+r_{C_2}}i_{L_2}+\frac{R}{R+r_{C_2}}v_{C_2}-\frac{R\times r_{C_2}}{R+r_{C_2}}i_o \qquad (19)$$

Consider a converter with the values given in Table 2.

TABLE 2.
THE ZETA CONVERTER PARAMETERS (SEE FIGURE 8)

	Nominal Value
Output voltage, Vo	5.2 V
Duty ratio, D	0.23
Input DC source voltage, Vg	20 V
Input DC source internal resistance, rg	0.0 Ω
MOSFET Drain-Source resistance, rds	10 mΩ
Capacitor, C1	100 µF
Capacitor Equivalent Series Resistance(ESR), rC1	0.19 Ω
Capacitor, C2	220 µF
Capacitor Equivalent Series Resistance(ESR), rC2	0.095 Ω
Inductor, L1	100 µH
Inductor ESR, rL1	1 mΩ
Inductor, L2	55 µH
Inductor ESR, rL2	0.55 mΩ
Diode voltage drop, vD	0.7 V
Diode forward resistance, rD	10 mΩ
Load resistor, R	6 Ω
Switching Frequency, Fsw	100 KHz

The code given in appendix (Program 2), extracts the transfer functions. Following results are obtained after running the code:

$$\frac{\tilde{v}_o(s)}{\tilde{i}_o(s)}=-.093519\frac{\left(s+4.785\times10^4\right)(s+1163)\left(s^2+1396s+6.882\times10^7\right)}{\left(s^2+2239s+4.76\times10^7\right)\left(s^2+2767s+1.026\times10^8\right)} \qquad (20)$$

$$\frac{\tilde{v}_o(s)}{\tilde{v}_g(s)}=391.08\frac{\left(s+4.785\times10^4\right)\left(s^2+1473s+7.7\times10^7\right)}{\left(s^2+2239s+4.76\times10^7\right)\left(s^2+2767s+1.026\times10^8\right)} \qquad (21)$$

$$\frac{\tilde{v}_o(s)}{\tilde{d}(s)}=43775\frac{\left(s+4.785\times10^4\right)\left(s^2+1371s+7.696\times10^7\right)}{\left(s^2+2239s+4.76\times10^7\right)\left(s^2+2767s+1.026\times10^8\right)} \qquad (22)$$

Bode diagram of these transfer functions are shown in Figure 11-13.

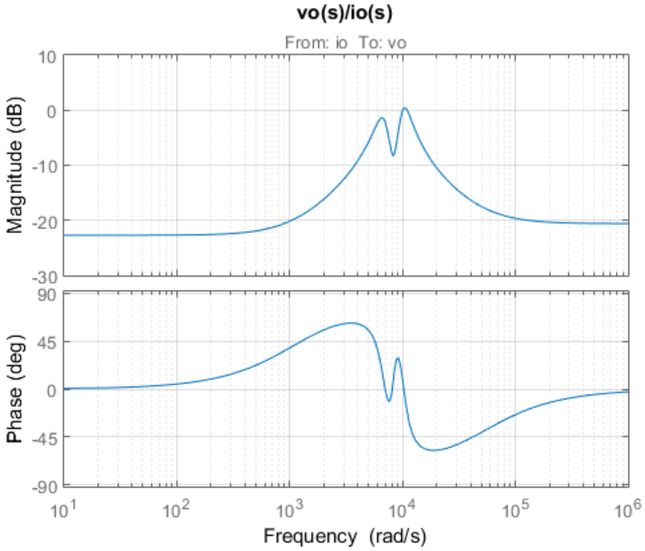


Fig. 11. Bode diagram of

$$\frac{\tilde{v}_o(s)}{\tilde{i}_o(s)}=.093519\frac{\left(s+4.785\times10^4\right)(s+1163)\left(s^2+1396s+6.882\times10^7\right)}{\left(s^2+2239s+4.76\times10^7\right)\left(s^2+2767s+1.026\times10^8\right)}.$$

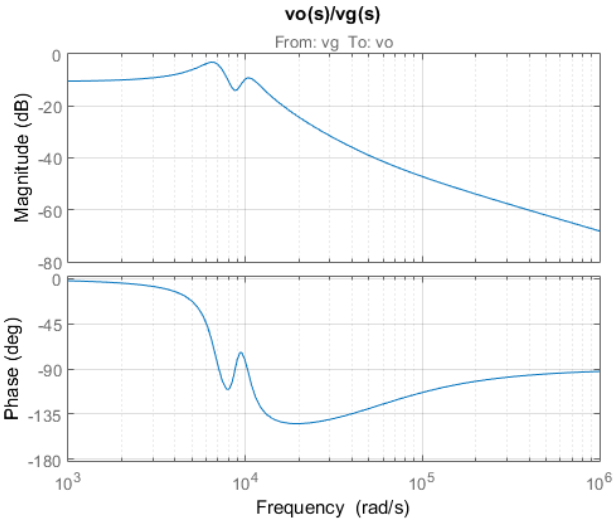


Fig. 12. Bode diagram of

$$\frac{\tilde{v}_o(s)}{\tilde{v}_g(s)} = 391.08\frac{(s+4.785\times10^4)(s^2+1473s+7.7\times10^7)}{(s^2+2239s+4.76\times10^7)(s^2+2767s+1.026\times10^8)}.$$

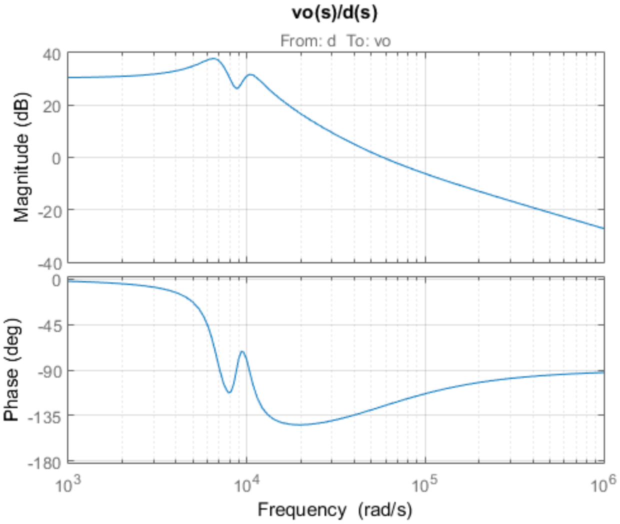


Fig. 13. Bode diagram of

$$\frac{\tilde{v}_o(s)}{\tilde{d}(s)} = 43775\frac{(s+4.785\times10^4)(s^2+1371s+7.696\times10^7)}{(s^2+2239s+4.76\times10^7)(s^2+2767s+1.026\times10^8)}.$$

Following block diagram can be drawn for the converter.

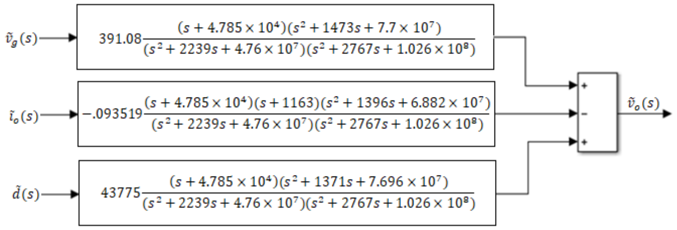

$\tilde{v}_g(s)$
$391.08\frac{(s+4.785\times10^4)(s^2+1473s+7.7\times10^7)}{(s^2+2239s+4.76\times10^7)(s^2+2767s+1.026\times10^8)}$
$\tilde{i}_o(s)$
$-.093519\frac{(s+4.785\times10^4)(s+1163)(s^2+1396s+6.882\times10^7)}{(s^2+2239s+4.76\times10^7)(s^2+2767s+1.026\times10^8)}$
$\tilde{d}(s)$
$43775\frac{(s+4.785\times10^4)(s^2+1371s+7.696\times10^7)}{(s^2+2239s+4.76\times10^7)(s^2+2767s+1.026\times10^8)}$
+
+
+
$\tilde{v}_o(s)$

Fig. 14. Block diagram of the studied converter.

IV. VERIFICATION OF OBTAINED RESULT

PLECS (Piecewise Linear Electrical Circuit Simulation) is a software tool for system level simulation of electrical circuits developed by Plexim [12]. PLECS comes in two versions: Standalone and Simulink version. The standalone has its own solver and can be run independently. The Simulink version, as the name suggests, runs under the MATLAB/Simulink environment and uses the Simulink solver. PLECS has a free trial version which can be used for period of one month.

PLECS can be used to verify the obtained results. The schematic shown in Figure 15 extracts the output impedance. Extracted output impedance is shown in Figure 16. Obtained result is the same as Figure 11.

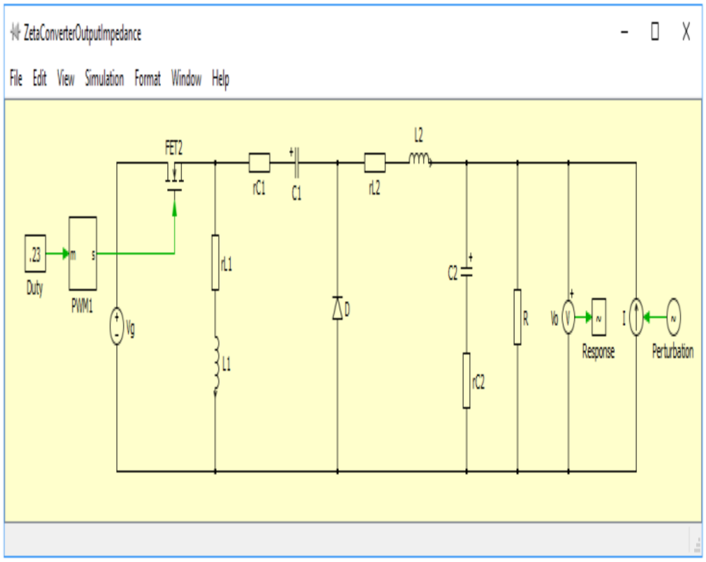

ZetaConverterOutputImpedance
File Edit View Simulation Format Window Help
FET2
L2
rC1
C1
rL2
rL1
C2
D
R
Vo
Response
Perturbation
.23
Duty
PWM1
Vg
L1
rC2

Fig. 15. Simulation diagram to extract the output impedance $(\frac{\tilde{v}_o(s)}{\tilde{i}_o(s)})$.

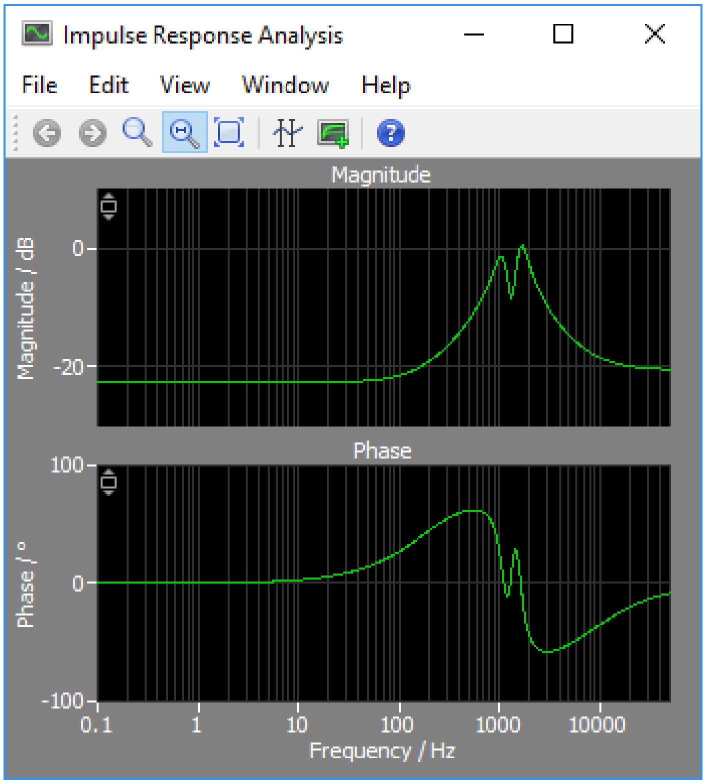


Fig. 16. Output impedance of the studied Zeta converter (0.1 Hz-50 KHz Range).

Schematics to extract the audio susceptibility $(\frac{\tilde{v}_o(s)}{\tilde{v}_g(s)})$ and control-to-output $(\frac{\tilde{v}_o(s)}{\tilde{d}(s)})$ are shown in Figure 17 and 18, respectively. Analysis results are shown in Figure 19 and 20.

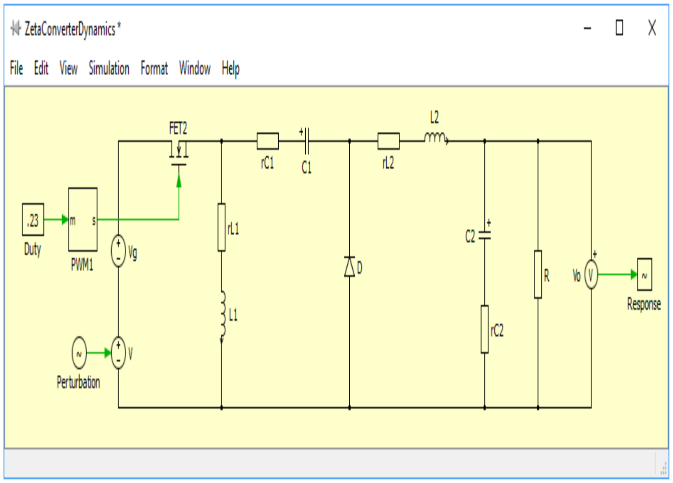


Fig. 17. Simulation diagram to extract the audio susceptibility $\left(\frac{\hat{v}_o(s)}{\hat{v}_g(s)}\right)$.

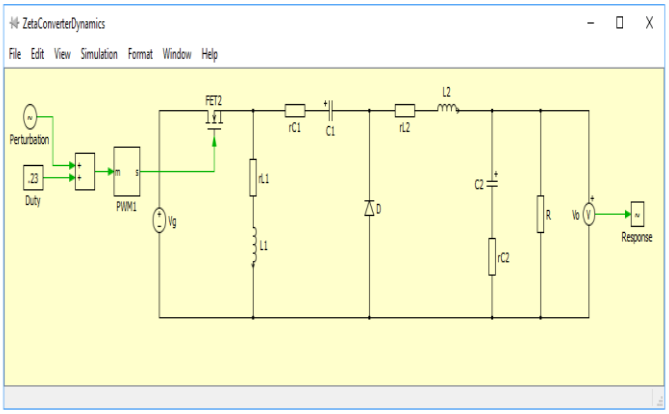


Fig. 18. Simulation diagram to extract the control-to-output $\left(\frac{\hat{v}_o(s)}{\hat{d}(s)}\right)$.

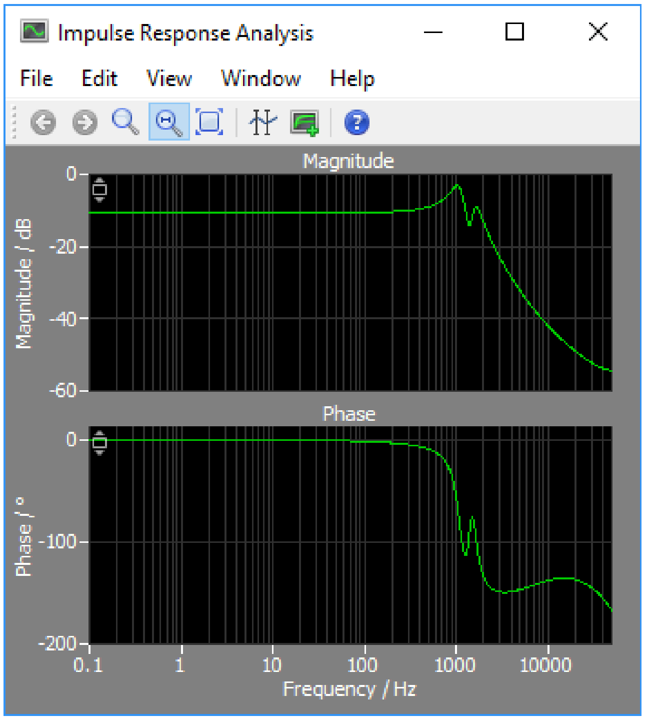


Fig. 19. Bode diagram of audio susceptibility $\left(\frac{\hat{v}_o(s)}{\hat{v}_g(s)}\right)$ transfer function for studied Zeta converter.

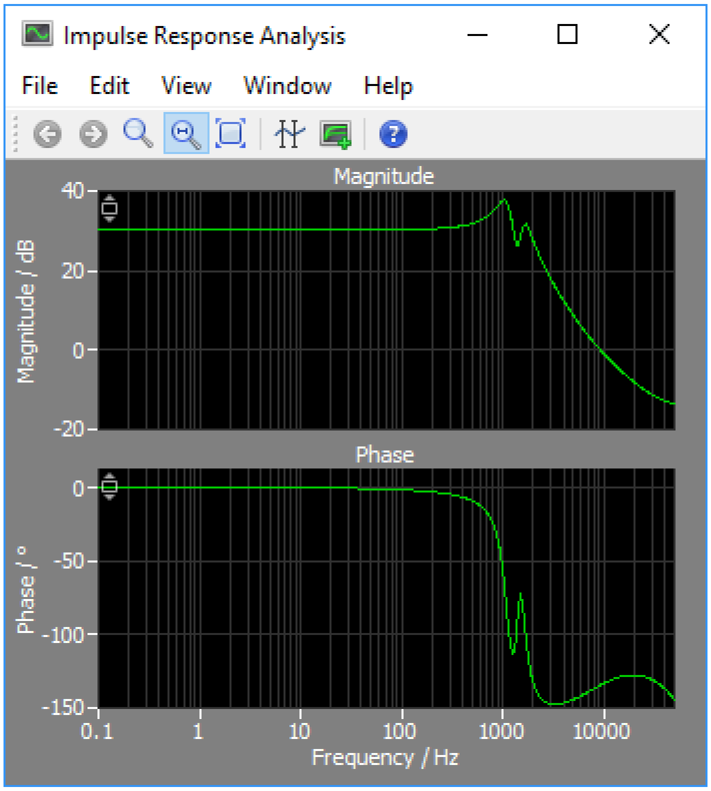


Fig. 20. Bode diagram of control-to-output $\left(\frac{\hat{v}_o(s)}{\hat{d}(s)}\right)$ transfer function for studied Zeta converter.

V. DEVELOPED TOOLBOX

Figure 21, shows the main window of developed toolbox. Current version of this toolbox supports the buck, boost, buck-boost, Cuk, SEPIC, fly back, forward and full bridge topologies.

Buck converter and SEPIC converter analysis section of the developed software is shown in Figure 22 and 23, respectively.

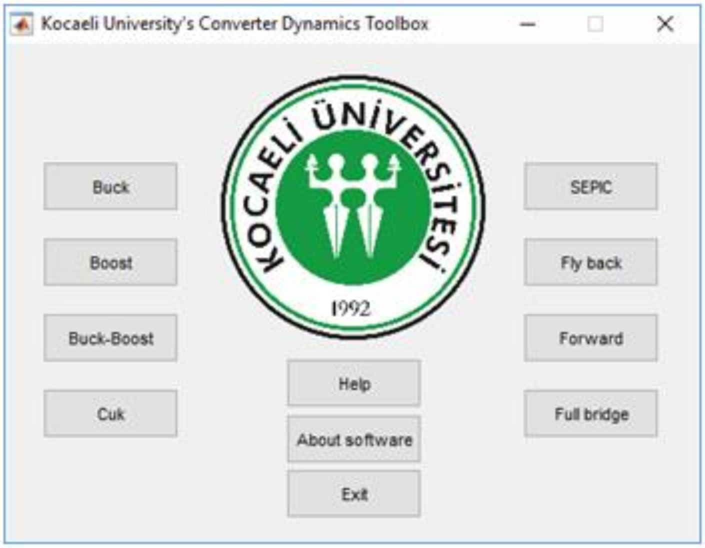
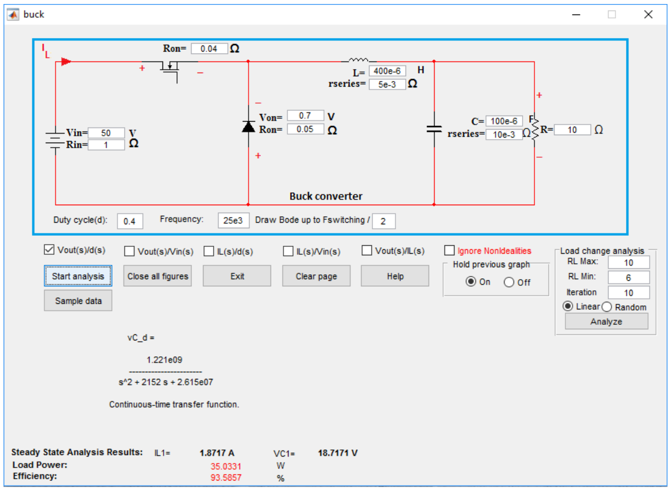


Fig. 21. Main menu of developed toolbox.



Duty cycle(d): 0.4   Frequency: 25e3   Draw Bode up to Fswitching / 2

☑ Vout(s)/d(s)   □ Vout(s)/Vin(s)   □ IL(s)/d(s)   □ IL(s)/Vin(s)   □ Vout(s)/IL(s)   □ Ignore NonIdealities

Hold previous graph: ● On ○ Off

Load change analysis: RL Max: 10   RL Min: 6   Iteration: 10   ● Linear ○ Random

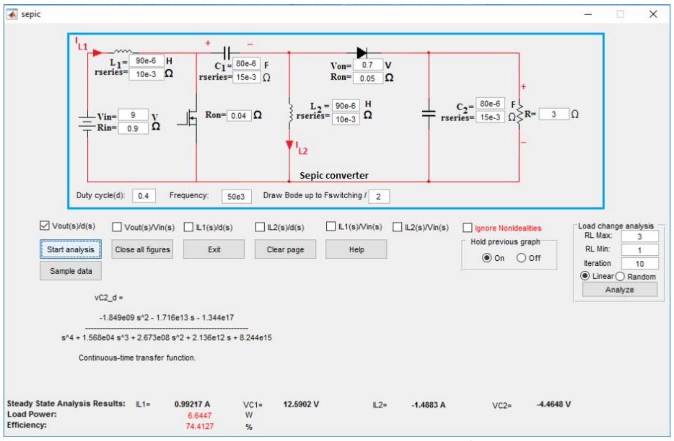
```
vC_d =

        1.221e09
  ----------------------
  s^2 + 2152 s + 2.615e07

Continuous-time transfer function.
```

```
Steady State Analysis Results:   IL1=     1.8717 A     VC1=    18.7171 V
Load Power:                               35.0331      W
Efficiency:                               93.5857      %
```

Fig. 22. Buck converter analysis section of the toolbox.



Duty cycle(d): 0.4   Frequency: 50e3   Draw Bode up to Fswitching / 2

☑ Vout(s)/d(s)   □ Vout(s)/Vin(s)   □ IL1(s)/d(s)   □ IL2(s)/d(s)   □ IL1(s)/Vin(s)   □ IL2(s)/Vin(s)   □ Ignore NonIdealities

Hold previous graph: ● On ○ Off

Load change analysis: RL Max: 3   RL Min: 1   Iteration: 10   ● Linear ○ Random

```
vC2_d =

      -1.849e09 s^2 - 1.716e13 s - 1.344e17
  ----------------------------------------------------
  s^4 + 1.568e04 s^3 + 2.673e08 s^2 + 2.136e12 s + 8.244e15

Continuous-time transfer function.
```

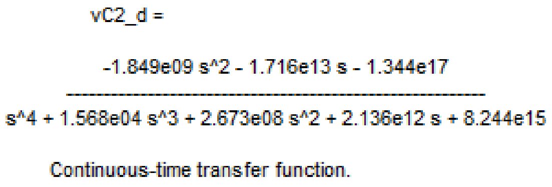
```
Steady State Analysis Results:   IL1=   0.99217 A    VC1=   12.5902 V        IL2=   -1.4883 A       VC2=    -4.4648 V
Load Power:                             6.6447       W
Efficiency:                             74.4127      %
```

Fig. 23. SEPIC converter analysis section of the toolbox.

User only enters the components values. Non idealities such as equivalent series resistance of inductors and capacitors, voltage drop of diodes, on resistance of MOSFET’s and internal resistance of input sources are taken into account. Obtaining the dynamical model of converter in presence of these non-idealities is quite cumbersome for pencil-and-paper analysis.

We want to obtain the $H(s) = \frac{\hat{v}_{c2}(s)}{\hat{d}(s)}$ for a SEPIC converter with the following components values:

V_{in}=9 V, $r_{internal}$=0.9 Ω, L_1=L_2=90 μH, rL_1=rL_2=10 mΩ, C_1=C_2=80 μF, r_{C1}=r_{C2}=15 mΩ, V_{Diode_on}=0.7 V, r_{Diode_on}=0.05 Ω, r_{MOSFET}=40 mΩ, R_{Load}=3 Ω. The software gives the following result (vC2_d= $\frac{\hat{v}_{c2}(s)}{\hat{d}(s)}$).



```
vC2_d =

      -1.849e09 s^2 - 1.716e13 s - 1.344e17
  ----------------------------------------------------
  s^4 + 1.568e04 s^3 + 2.673e08 s^2 + 2.136e12 s + 8.244e15

Continuous-time transfer function.
```

Fig. 24. Algebraic transfer function calculated for $\frac{\hat{v}_{C_2}(s)}{\hat{d}(s)}$.

It draws the frequency response and pole-zero map of obtained transfer function as well.

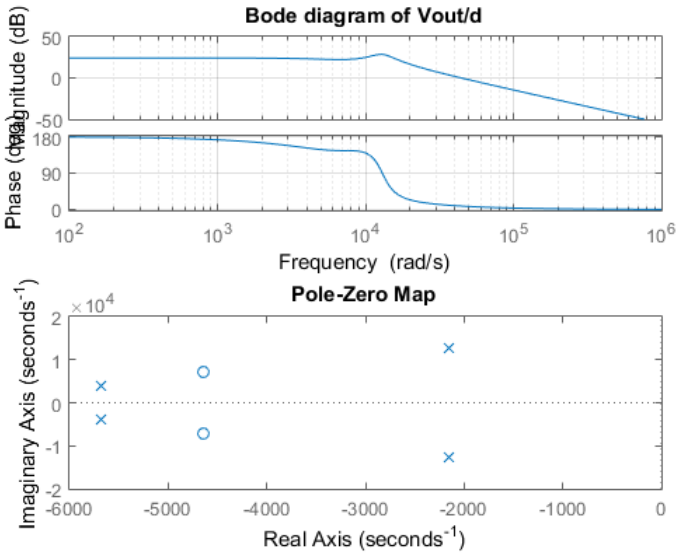


Fig. 25. Bode diagram and pole zero diagram of $\frac{\hat{v}_{C_2}(s)}{\hat{d}(s)}$.

You can study the effect of changes in component values on the converter dynamics. Figure 26, shows the effect of change in load on the control-to-output voltage ($\frac{\hat{v}_{c2}(s)}{\hat{d}(s)}$) transfer function. This option helps you to understand the effect of uncertainties on the system dynamics.

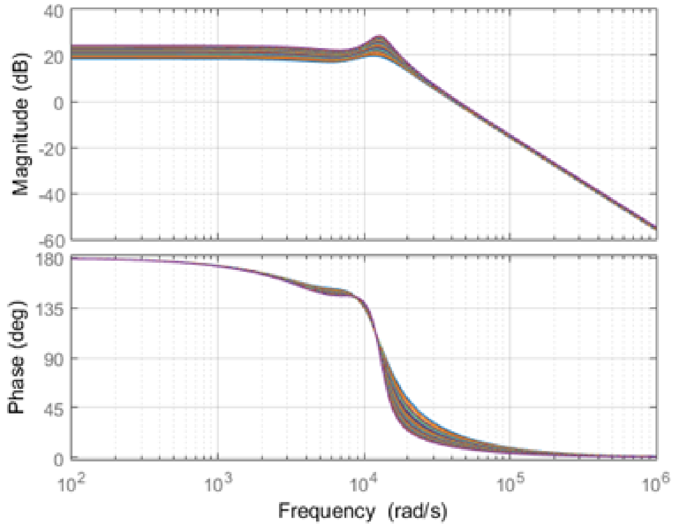


Fig. 26. Effect of load changes on the $\frac{\hat{v}_{C_2}(s)}{\hat{d}(s)}$ transfer function.

VI. CONCLUSION

Switched DC-DC converters need feedback control to provide the required output voltage or current for the load. Obtaining the required output voltage or current in presence of disturbances such as input voltage changes and/or output load changes seems difficult without some form of control.

Obtaining the mathematical model of the switched DC-DC converter is the first step of controller design procedure (in model base controller design techniques). First part of this paper studied the different techniques to extract the converter dynamics. Both ready to use software packages and MATLAB programming are used to extract the converter dynamics. Second part of this paper introduced a MATLAB toolbox to do the modelling job of DC-DC converters automatically. Developed toolbox can be used for educational or industrial purposes.

This paper can be a good reference for researchers interested in dynamics and control of DC-DC converters.

REFERENCES

[1] R. Ericson , D. Maksimovic, *Fundamentals of Power Electronics,* Norwell: Kluwer Academic Publisher 2001. pp. 161-165.

[2] SA. Akbarabadi, H. Atighechi, J. Jatskevich, “Circuit-averaged and state-space-averaged-value modeling of second order flyback converter in CCM and DCM including conduction losses”. 4th International Conference on Power Engineering, Energy and Electrical Drives; 2013; Istanbul. pp. 995-1000.

[3] F. Asadi, K. Eguchi, *Dynamics and control of DC-DC converters,* Morgan and Claypool; 2018. pp. 89-145.

[4] T. Suntio, *Dynamic profile of switched mode converter: modeling, analysis and control.* John Wiley & Sons. 2009. pp. 17-37.

[5] RD. Middlebrook, S. Cuk, “A general unified approach to modeling switching-converter power stages”. Int. J. Electronics Theoretical and Experimental.1977. pp. 521–550.

[6] R. Tymerski, V. Vorperian, “Generation, classification and analysis of switched-mode DC-DC converters by the use of converter cells”. Telecommunications Energy Conference. 1986. pp. 181-195.

[7] J. Chen, T. Ngo, “Alternate forms of the PWM switch model in discontinuous conduction mode”. IEEE Trans. Aerosp. Electron. Syst. 2001.pp. 754-758.

[8] J. Sun, M. Mitchell, M. Greuel, T. Krein, R. Bass, “Average modeling of PWM converters in discontinuous modes”. IEEE Trans. Power Electron. 2001.pp. 482–492.

[9] S. Cuk, R. Middlebrook, “A general unified approach in modeling switching DC-to-DC converters in discontinuous conduction mode”. in Proc. IEEE Power Electronics Special Conf.1977. pp. 36–57.

[10] T. Suntio, “Unified average and small-signal modeling of direct on-time control”, IEEE Trans. Indust. Electron. 2006. pp. 287–295.

[11] D. Maksimovic, A. Stankovic, J. Tottuvelil, C. Verghese, “Modeling and simulation of power electronic converters”, Proc. IEEE, vol. 89, no. 6. 2001. pp. 898–912.

[12] J. Allmeling, P. Hammer, “PLECS - piece-wise linear electrical circuit simulation for Simulink”. Proceedings of the IEEE 1999 International Conference on Power Electronics and Drive Systems.1999. pp. 355–360.

BIOGRAPHIES

FARZIN ASADI received his BSc in Electronics Engineering, MSc in Control Engineering and Phd in Mechatronics Engineering.

Currently he is with the Department of Electrical and Electronics Engineering at the Maltepe University, Istanbul, Turkey. Dr. Asadi has published more than 40 international papers and 12 books. He is on the editorial board of 7 scientific journals as well. His research interests include switching converters, control theory, robust control of power electronics converters, and robotics.

Appendix

Program 1

```
%This program extracts the small signal transfer functions
% of a Buck converter
clc
clear all

%converter components values
%fsw= 20 KHz
VG=50;      %input DC source voltage
rg=0.5;     %input DC source internal resistance
rds=0.04;   %MOSFET drain-source resistance
rD=0.01;    %Diode series resistance
VD=0.7;     %Diode voltage drop
rL=10e-3;   %Inductor Equivalent Series Resistance(ESR)
L=400e-6;   %Inductor value
rC=0.05;    %Capacitor ESR
C=100e-6;   %Capacitor value
R=20;       %Load resistor
D=0.4;      %Duty ratio
IO=0;       %Average value of output current source

syms iL vC io vg vD d
% iL : Inductor L1 current
% vC : Capacitor C1 voltage
% io : Output current source
% vg : Input DC source
% vD : Diode voltage drop
% d  : Duty cycle

%Closed MOSFET Equations
diL_dt_MOSFET_close=(-(rg+rds+rL+R*rC/(R+rC))*iL-R/(R+rC)*vC+R*rC/(R+rC)*io+vg)/L;
dvC_dt_MOSFET_close=(R/(R+rC)*iL-1/(R+rC)*vC-R/(R+rC)*io)/C;
vo_MOSFET_close=R*rC/(R+rC)*iL+R/(R+rC)*vC-R*rC/(R+rC)*io;

%Opened MOSFET Equations
diL_dt_MOSFET_open=(-(rD+rL+rC*R/(R+rC))*iL-R/(R+rC)*vC+R*rC/(R+rC)*io-vD)/L;
dvC_dt_MOSFET_open=(R/(R+rC)*iL-1/(R+rC)*vC-R/(R+rC)*io)/C;
vo_MOSFET_open=R*rC/(R+rC)*iL+R/(R+rC)*vC-R*rC/(R+rC)*io;

%Averaging
averaged_diL_dt=simplify(d*diL_dt_MOSFET_close+(1-d)*diL_dt_MOSFET_open);
averaged_dvC_dt=simplify(d*dvC_dt_MOSFET_close+(1-d)*dvC_dt_MOSFET_open);
averaged_vo=simplify(d*vo_MOSFET_close+(1-d)*vo_MOSFET_open);

%Substituting the steady values of: input DC voltage source, Diode voltage
%drop, Duty cycle and output current source and calculating the DC
%operating point(IL and VC)
right_side_of_averaged_diL_dt=subs(averaged_diL_dt,[vg vD d io],[VG VD D IO]);
right_side_of_averaged_dvC_dt=subs(averaged_dvC_dt,[vg vD d io],[VG VD D IO]);

DC_OPERATING_POINT=
solve(right_side_of_averaged_diL_dt==0,right_side_of_averaged_dvC_dt==0,'iL','vC');

IL=eval(DC_OPERATING_POINT.iL);
VC=eval(DC_OPERATING_POINT.vC);
VO=eval(subs(averaged_vo,[iL vC io],[IL VC IO]));

disp('Operating point of converter')
disp('----------------------------')
disp('IL(A)=')
disp(IL)
disp('VC(V)=')
disp(VC)
disp('VO(V)=')
disp(VO)
disp('----------------------------')

%Linearizing the averaged equations around the DC operating point.
%We want to obtain the matrix A,B,C and D
%     .
%     x=Ax+Bu
%     y=Cx+Du
%
%where,
%     x=[iL vC]'
%     u=[io vg d]'
%since we used the variables D for steady state duty ratio and C to
%show the capacitors values we use AA, BB, CC and DD instead of A,
%B, C and D.

% Calculating the matrix A
A11=subs(simplify(diff(averaged_diL_dt,iL)),[iL vC d io],[IL VC D IO]);
A12=subs(simplify(diff(averaged_diL_dt,vC)),[iL vC d io],[IL VC D IO]);

A21=subs(simplify(diff(averaged_dvC_dt,iL)),[iL vC d io],[IL VC D IO]);
A22=subs(simplify(diff(averaged_dvC_dt,vC)),[iL vC d io],[IL VC D IO]);


AA=eval([A11 A12;
         A21 A22]);

% Calculating the matrix B
B11=subs(simplify(diff(averaged_diL_dt,io)),[iL vC d vD io vg],[IL VC D VD IO VG]);
B12=subs(simplify(diff(averaged_diL_dt,vg)),[iL vC d vD io vg],[IL VC D VD IO VG]);
B13=subs(simplify(diff(averaged_diL_dt,d)),[iL  vC d vD io vg],[IL VC D VD IO VG]);

B21=subs(simplify(diff(averaged_dvC_dt,io)),[iL vC d vD io vg],[IL VC D VD IO VG]);
B22=subs(simplify(diff(averaged_dvC_dt,vg)),[iL vC d vD io vg],[IL VC D VD IO VG]);
B23=subs(simplify(diff(averaged_dvC_dt,d)),[iL  vC d vD io vg],[IL VC D VD IO VG]);

BB=eval([B11 B12 B13;
         B21 B22 B23]);

% Calculating the matrix C
C11=subs(simplify(diff(averaged_vo,iL)),[iL vC d io],[IL VC D IO]);
C12=subs(simplify(diff(averaged_vo,vC)),[iL vC d io],[IL VC D IO]);

CC=eval([C11 C12]);

D11=subs(simplify(diff(averaged_vo,io)),[iL vC d vD io vg],[IL VC D VD IO VG]);
D12=subs(simplify(diff(averaged_vo,vg)),[iL vC d vD io vg],[IL VC D VD IO VG]);
D13=subs(simplify(diff(averaged_vo,d)),[iL  vC d vD io vg],[IL VC D VD IO VG]);

% Calculating the matrix D
DD=eval([D11 D12 D13]);

% Producing the State Space Model and obtaining the small signal transfer
% functions
```

```
sys=ss(AA,BB,CC,DD);
sys.inputname={'io';'vg';'d'};
sys.outputname={'vo'};

vo_io=tf(sys(1,1)); % Output impedance transfer function vo(s)/io(s)
vo_vg=tf(sys(1,2)); % vo(s)/vg(s)
vo_d=tf(sys(1,3));  % Control-to-output(vo(s)/d(s))

%drawing the Bode diagrams
figure(1)
bode(vo_io),grid minor,title('vo(s)/io(s)')

figure(2)
bode(vo_vg),grid minor,title('vo(s)/vg(s)')

figure(3)
bode(vo_d),grid minor,title('vo(s)/d(s)')
```

Program 2

```
% This program calculates the small signal transfer functions of Zeta converter
clc
clear all

VG=20;      % Average value of input DC source
rg=0;       % Internal resistance of input DC source
rds=.01;    % MOSFET on resistance
C1=100e-6;  % Capacitor C1 value
C2=220e-6;  % Capacitor C2 value
rC1=.19;    % Capacitor C1 Equivalent Series Resistance(ESR)
rC2=.095;   % Capacitor C2 Equivalent Series Resistance(ESR)
L1=100e-6;  % Inductor L1 value
L2=55e-6;   % Inductor L2 value
rL1=1e-3;   % Inductor L1 Equivalent Series Resistance(ESR)
rL2=.55e-3; % Inductor L2 Equivalent Series Resistance(ESR)
rD=.01;     % Diode series resistance
VD=.7;      % Diode voltage drop
R=6;       % Load resistance
D=.23;       % Duty cylcle
IO=0;       % Average value of output current source
fsw=100e3;  % Switching frequency

syms iL1 iL2 vC1 vC2 io vg vD d
% iL1: Inductor L1 current
% iL2: Inductor L2 current
% vC1: Capacitor C1 voltage
% vC2: Capacitor C2 voltage
% io : Output current source
% vg : Input DC source
% vD : Diode voltage drop
% d  : Duty cycle

%Closed MOSFET Equations
diL1_dt_MOSFET_close=(-(rL1+rg+rds)*iL1-(rg+rds)*iL2+vg)/L1;
diL2_dt_MOSFET_close=(-(rg+rds)*iL1-(rg+rds+rC1+rL2+R*rC2/(R+rC2))*iL2+vC1-
R/(R+rC2)*vC2+R*rC2/(R+rC2)*io+vg)/L2;
dvC1_dt_MOSFET_close=(-iL2)/C1;
dvC2_dt_MOSFET_close=(R/(R+rC2)*iL2-1/(R+rC2)*vC2-R/(R+rC2)*io)/C2;
vo_MOSFET_close=R*rC2/(R+rC2)*iL2+R/(R+rC2)*vC2-R*rC2/(R+rC2)*io;

%Opened MOSFET Equations
diL1_dt_MOSFET_open=(-(rL1+rC1+rD)*iL1-rD*iL2-vC1-vD)/L1;
diL2_dt_MOSFET_open=(-rD*iL1-(rD+rL2+R*rC2/(R+rC2))*iL2-
R/(R+rC2)*vC2+R*rC2/(R+rC2)*io-vD)/L2;
dvC1_dt_MOSFET_open=(iL1)/C1;
dvC2_dt_MOSFET_open=(R/(R+rC2)*iL2-1/(R+rC2)*vC2-R/(R+rC2)*io)/C2;
vo_MOSFET_open=R*rC2/(R+rC2)*iL2+R/(R+rC2)*vC2-R*rC2/(R+rC2)*io;

%Averaging
averaged_diL1_dt=simplify(d*diL1_dt_MOSFET_close+(1-d)*diL1_dt_MOSFET_open);
averaged_diL2_dt=simplify(d*diL2_dt_MOSFET_close+(1-d)*diL2_dt_MOSFET_open);
averaged_dvC1_dt=simplify(d*dvC1_dt_MOSFET_close+(1-d)*dvC1_dt_MOSFET_open);
averaged_dvC2_dt=simplify(d*dvC2_dt_MOSFET_close+(1-d)*dvC2_dt_MOSFET_open);
averaged_vo=simplify(d*vo_MOSFET_close+(1-d)*vo_MOSFET_open);

%Substituting the steady values of input DC voltage source, Diode voltage
%drop, Duty cycle and output current source and calculating the DC
%operating point
```

```
right_side_of_averaged_diL1_dt=subs(averaged_diL1_dt,[vg vD d io],[VG VD D IO]);
right_side_of_averaged_diL2_dt=subs(averaged_diL2_dt,[vg vD d io],[VG VD D IO]);
right_side_of_averaged_dvC1_dt=subs(averaged_dvC1_dt,[vg vD d io],[VG VD D IO]);
right_side_of_averaged_dvC2_dt=subs(averaged_dvC2_dt,[vg vD d io],[VG VD D IO]);

DC_OPERATING_POINT=
solve(right_side_of_averaged_diL1_dt==0,right_side_of_averaged_diL2_dt==0,right_side_o
f_averaged_dvC1_dt==0,right_side_of_averaged_dvC2_dt==0,'iL1','iL2','vC1','vC2');

IL1=eval(DC_OPERATING_POINT.iL1);
IL2=eval(DC_OPERATING_POINT.iL2);
VC1=eval(DC_OPERATING_POINT.vC1);
VC2=eval(DC_OPERATING_POINT.vC2);
VO=eval(subs(averaged_vo,[iL1 iL2 vC1 vC2 io],[IL1 IL2 VC1 VC2 IO]));

disp('Operating point of converter')
disp('-----------------------------')
disp('IL1(A)=')
disp(IL1)
disp('IL2(A)=')
disp(IL2)
disp('VC1(V)=')
disp(VC1)
disp('VC2(V)=')
disp(VC2)
disp('VO(V)=')
disp(VO)
disp('-----------------------------')

%Linearizing the averaged equations around the DC operating point.
%We want to obtain the matrix A,B,C and D
%      .
%     x=Ax+Bu
%     y=Cx+Du
%
%where,
%     x=[iL1 iL2 vC1 vC2]'
%     u=[io vg d]'
%Since we used the variables D for steady state duty ratio and C to
%show the capacitors values we use AA, BB, CC and DD instead of A,
%B, C and D.

% Calculating the matrix A
A11=subs(simplify(diff(averaged_diL1_dt,iL1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A12=subs(simplify(diff(averaged_diL1_dt,iL2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A13=subs(simplify(diff(averaged_diL1_dt,vC1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A14=subs(simplify(diff(averaged_diL1_dt,vC2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);

A21=subs(simplify(diff(averaged_diL2_dt,iL1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A22=subs(simplify(diff(averaged_diL2_dt,iL2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A23=subs(simplify(diff(averaged_diL2_dt,vC1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
```

```
A24=subs(simplify(diff(averaged_diL2_dt,vC2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);

A31=subs(simplify(diff(averaged_dvC1_dt,iL1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A32=subs(simplify(diff(averaged_dvC1_dt,iL2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A33=subs(simplify(diff(averaged_dvC1_dt,vC1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A34=subs(simplify(diff(averaged_dvC1_dt,vC2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);

A41=subs(simplify(diff(averaged_dvC2_dt,iL1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A42=subs(simplify(diff(averaged_dvC2_dt,iL2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A43=subs(simplify(diff(averaged_dvC2_dt,vC1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);
A44=subs(simplify(diff(averaged_dvC2_dt,vC2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2
D IO]);

AA=eval([A11 A12 A13 A14;
         A21 A22 A23 A24;
         A31 A32 A33 A34;
         A41 A42 A43 A44]);

% Calculating the matrix B
B11=subs(simplify(diff(averaged_diL1_dt,io)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);
B12=subs(simplify(diff(averaged_diL1_dt,vg)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);
B13=subs(simplify(diff(averaged_diL1_dt,d)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);

B21=subs(simplify(diff(averaged_diL2_dt,io)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);
B22=subs(simplify(diff(averaged_diL2_dt,vg)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);
B23=subs(simplify(diff(averaged_diL2_dt,d)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);

B31=subs(simplify(diff(averaged_dvC1_dt,io)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);
B32=subs(simplify(diff(averaged_dvC1_dt,vg)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);
B33=subs(simplify(diff(averaged_dvC1_dt,d)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);

B41=subs(simplify(diff(averaged_dvC2_dt,io)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);
B42=subs(simplify(diff(averaged_dvC2_dt,vg)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);
B43=subs(simplify(diff(averaged_dvC2_dt,d)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1
VC2 D VD IO VG]);

BB=eval([B11 B12 B13;
         B21 B22 B23;
         B31 B32 B33;
         B41 B42 B43]);

% Calculating the matrix C
C11=subs(simplify(diff(averaged_vo,iL1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2 D
IO]);
C12=subs(simplify(diff(averaged_vo,iL2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2 D
IO]);
C13=subs(simplify(diff(averaged_vo,vC1)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2 D
IO]);
C14=subs(simplify(diff(averaged_vo,vC2)),[iL1 iL2 vC1 vC2 d io],[IL1 IL2 VC1 VC2 D
IO]);

CC=eval([C11 C12 C13 C14]);

D11=subs(simplify(diff(averaged_vo,io)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1 VC2
D VD IO VG]);
D12=subs(simplify(diff(averaged_vo,vg)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1 VC2
D VD IO VG]);
D13=subs(simplify(diff(averaged_vo,d)),[iL1 iL2 vC1 vC2 d vD io vg],[IL1 IL2 VC1 VC2 D
VD IO VG]);

% Calculating the matrix D
DD=eval([D11 D12 D13]);

% Producing the State Space Model and obtaining the small signal transfer
% functions
sys=ss(AA,BB,CC,DD);
sys.inputname={'io';'vg';'d'};
sys.outputname={'vo'};

vo_io=tf(sys(1,1)); % Output impedance transfer function vo(s)/io(s)
vo_vg=tf(sys(1,2)); % vo(s)/vg(s)
vo_d=tf(sys(1,3));  % Control-to-output(vo(s)/d(s))

%drawing the Bode diagrams
figure(1)
bode(vo_io),grid minor,title('vo(s)/io(s)')

figure(2)
bode(vo_vg),grid minor,title('vo(s)/vg(s)')

figure(3)
bode(vo_d),grid minor,title('vo(s)/d(s)')
```