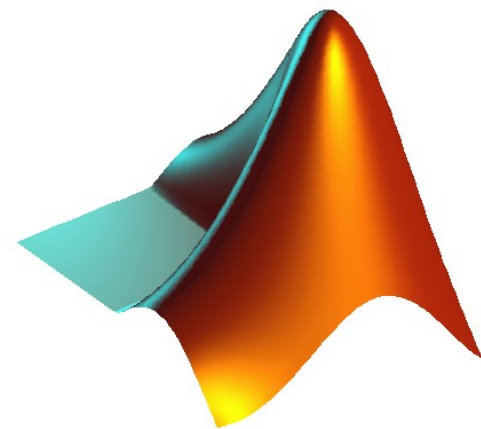
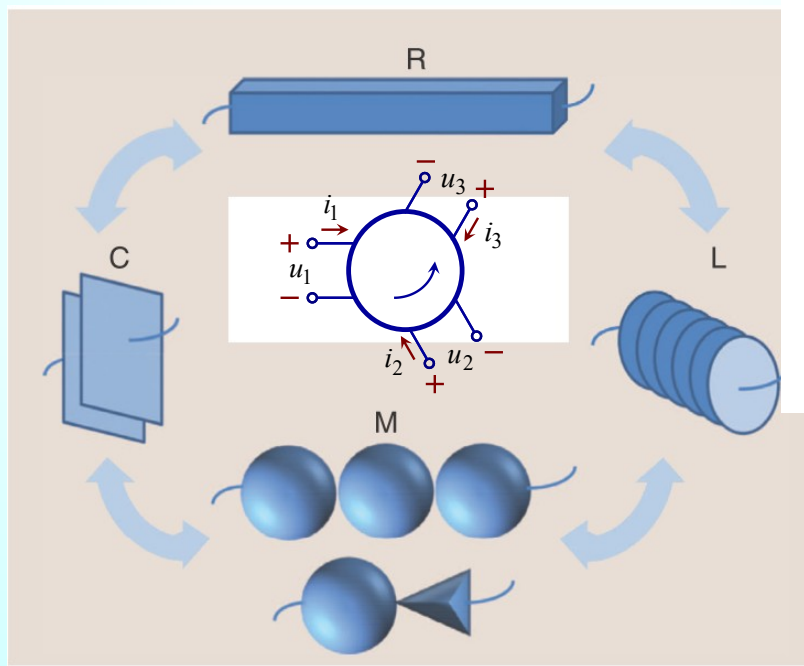


Практикум из рачунарске анализе трофазних кола

1. Увод



Милка Потребих Иваниш
Никола Баста



Практикум из рачунарске анализе трофазних кола 19E072ПРТК

- **Настава се изводи према текућем студијском програму који је Електротехнички факултет акредитовао** <http://www.etf.bg.ac.rs>
- Студијски програм: **Електротехника и рачунарство**
- Изборно подручје (модул): **Енергетика**
- **Циљ.** Излагање основних концепата анализе помоћу рачунара који се користе у софтверским пакетима за симулацију трофазних електричних кола са гледишта инжењера и истраживача електроенергетике. Увођење основа за анализу модела електроенергетских система и направа за нумеричку и симболичку анализу у временском и фреквенцијском домену. Изградња основних знања из теорије и примене симетричних компоненти.
- **Исход.** Разумевање основних концепата анализе трофазних електричних кола помоћу рачунара и одговарајућих математичких алгоритама. Решавање линеарних трофазних електричних кола у временском и фреквенцијском домену помоћу система компјутерске алгебре.
- **Садржај.** Основни концепти рачунарске симулације трофазних електричних кола. Симболичко и нумеричко израчунавање симетричних компоненти. Моделовање и симулација електроенергетских водова, компоненти и система помоћу савремених софтверских алата.

Практикум из рачунарске анализе трофазних кола 19E072ПРТК

- **Наставници:** Др **Милка Потребих Иваниш**, редовни професор, соба 64 (milka_potrebic@etf.rs);
Др **Никола Баста**, доцент, лаб. 63 (nbasta@etf.rs).
- **Асистенткиња:** **Ања Ковачевић**, мс. инж., соба 95 (anja@etf.rs)
- **Сарадница у настави:** **Анђела Нинковић**, дипл. инж., лаб. 63 (andjela@etf.rs)
- Белешке са предавања и лаб. вежби, примери испитних задатака и питања, <http://tek.etf.rs> ;
- Додатни материјали и консултације биће организовани преко канала на MS Teams платформи (линкови за канале су на адреси <http://tek.etf.rs>);
- Начин одржавања наставе. Рачунарска учионица (70, РЦ, 64б);
- Пратити званично представљање предмета и објављивање општих службених порука на адреси <http://tek.etf.rs/> ;

Начин полагања испита

- Структура оцене

Активност	Максимални број поена	Опис / Напомена
Предиспитне обавезе	70 поена	Кратки задаци на часовима и/или домаћи задаци (максимално 3 домаћа задатка).
Испит (пројекат)	30 поена	Израда на рачунару уз коришћење литературе у сваком испитном року.
Бонус	10 поена	MATLAB Onramp сертификат (добровољно).

За израду домаћих задатака јавити се и-мејлом Милки Потребих Иваниш.

Рачунарски (софтверски) алати

- *Mathematica*, WolframAlpha, MuPAD, Maxima, SymPy, SymPy Live, SymPy Gamma
- MATLAB, Scilab, Octave, FreeMat, Julia
- LTspice, QucsStudio, ngspice, Xyce, XCircuit
- Python, MathCAD, MAPLE, GeoGebra, Sage
- Symbolab, SpeQ Mathematics, meta-calculator, Desmos, ...
- Android apk: NCalc+, CalcES, SymJa CAS, Scientific Calculator Pro, CYZSoft Scientific Calculator Plus

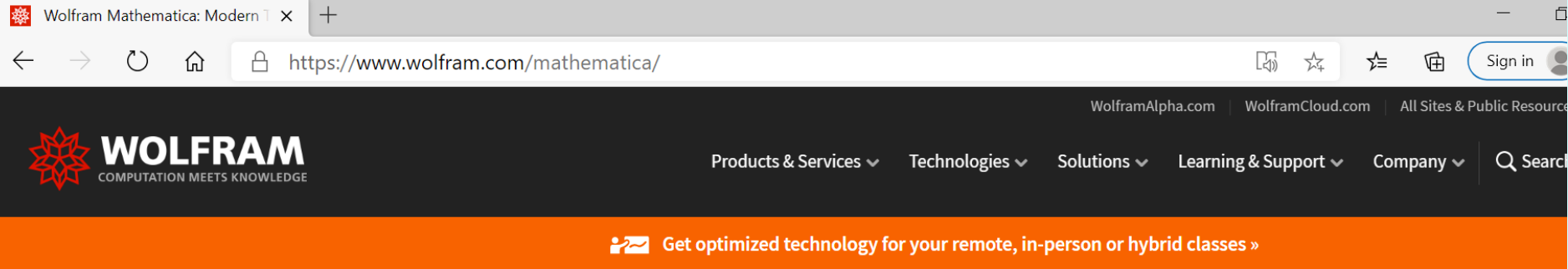
Free/Libre Open Source Software (FLOSS)

Моделовање електричних кола

- PSpice, ngSpice, Xcos, SciLab, TINA-TI, Micro-Cap, Plexim, 5Spice...
- **Online Circuit Simulator:** DoCircuits, Circuit Lab, PartSim, EasyEda, Multisim, PartQuest (SystemVision Cloud)...
- **Android circuit simulator:** Electric Circuit Studio, Circuit Safari SPICE Simulator...

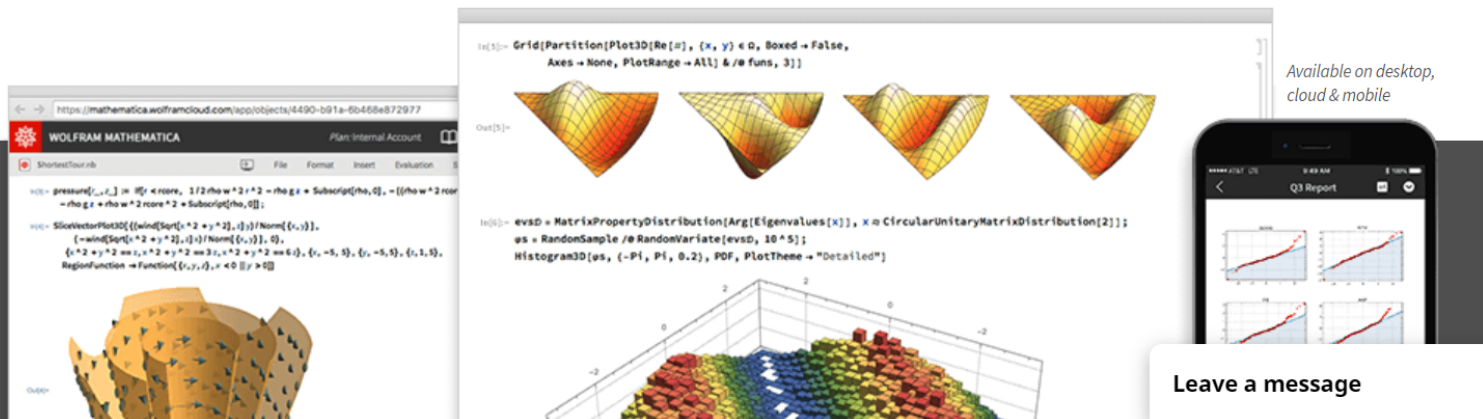
Mathematica

<https://www.wolfram.com/mathematica/>



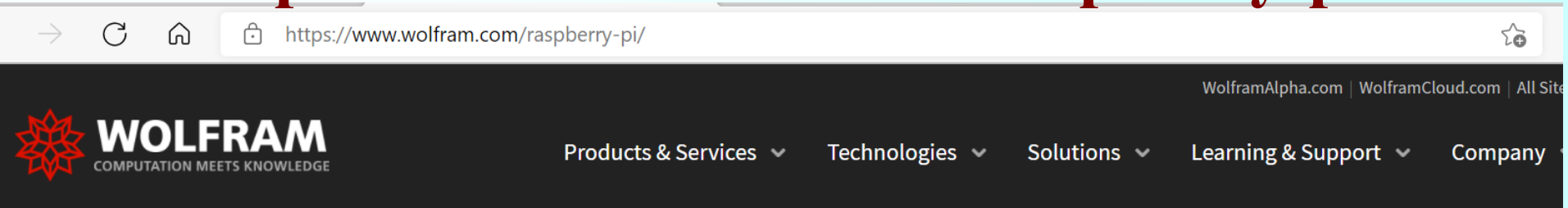
WOLFRAM MATHEMATICA

The world's definitive system for modern technical computing



Mathematica free on every Raspberry Pi!

<https://www.wolfram.com/raspberry-pi/>



www.wolframalpha.com

Give your brain a quick workout with Wolfram Problem Generator.



$y' + 1000 * y = 10000, y(0) = -10$



Examples Random

Input:

$$\{y'(x) + 1000 y(x) = 10\,000, y(0) = -10\}$$

ODE classification:

first-order linear ordinary differential equation

Alternate forms:

$$\{y'(x) = 10\,000 - 1000 y(x), y(0) = -10\}$$

$$\{y'(x) + 1000 (y(x) - 10) = 0, y(0) = -10\}$$

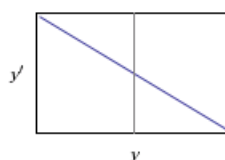
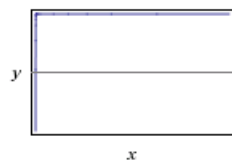
Differential equation solution:

Approximate form

Step-by-step solution

$$y(x) = 10 - 20 e^{-1000 x}$$

Plots of the solution:



New to
Wolfram|Alpha?



Take the Tour >>

New!
Wolfram Problem
Generator

Need a hint?

Maxima

<http://maxima.sourceforge.net/>

The screenshot shows the Maxima website interface. At the top, there's a navigation bar with links for Downloads, Documentation, Project, and Page Top. Below this, a large black box displays a complex mathematical result for the integral of $1/(1+x^4)$. The result is a sum of logarithmic and arctangent functions, with terms involving $\sqrt{2}$ and x . Below the result, there are three small circles, with the first one being filled, indicating the current step in a sequence. At the bottom of the page, there's a section titled "Maxima, a Computer Algebra System" with a paragraph describing its capabilities. To the right of this section, there's a "Recent Releases" box listing several versions with their release dates. The versions listed are 5.44.0 (June 8, 2020), 5.43.2 (Jan 27, 2020), 5.43.1 (Jan 20, 2020), 5.43.0 (May 31, 2019), and 5.42.2 (January 22, 2019). A link is provided to click on a version number to see the list of main changes.

Maxima, a Computer Algebra System

Maxima is a system for the manipulation of symbolic and numerical expressions, including differentiation, integration, Taylor series, Laplace transforms, ordinary differential equations, systems of linear equations, polynomials, sets, lists, vectors, matrices and tensors. Maxima yields high precision numerical results by using exact fractions, arbitrary-precision integers and variable-precision floating-point numbers. Maxima can plot functions and data in two and three dimensions.

The Maxima source code can be compiled on many systems, including Windows, Linux, and MacOS X. The source code for all systems and precompiled binaries for Windows and Linux are available at the [SourceForge file manager](#).

Maxima is a descendant of Macsyma, the legendary computer algebra system developed in the late 1960s at the [Massachusetts Institute of Technology](#). It is the only system based on that effort still publicly available and with an active user community, thanks to its open source nature. Macsyma was revolutionary in its day, and many later systems, such as Maple and Mathematica, were

Recent Releases

Click on a version number to see the list of main changes.

- [5.44.0](#): June 8, 2020
- [5.43.2](#): Jan 27, 2020
- [5.43.1](#): Jan 20, 2020
- [5.43.0](#): May 31, 2019
- [5.42.2](#): January 22, 2019

Maxima on line

<http://maxima.cesga.es/>

Servicio proporcionado por  CESGA

Maxima on line

Help: [Español](#), [English](#), [Galego](#)

```
expand((x-2)^3*(x+1/3)^2);  
solve(x^2-x+2=0);  
invert(matrix([2,3,1], [a,0,0], [1,4,8]));  
integrate(x * sin(x), x);  
draw3d(explicit(x^2+y^2,x,-1,1,y,-1,1));
```

Clic

Clear

[AVISO LEGAL](#)

(%i1) `expand((x-2)^3*(x+1/3)^2);`

(%o1) $x^5 - \frac{16x^4}{3} + \frac{73x^3}{9} - \frac{2x^2}{3} - 4x - \frac{8}{9}$

(%i2) `solve(x^2-x+2=0);`

(%o2) $\left[x = -\frac{\sqrt{7}i-1}{2}, x = \frac{\sqrt{7}i+1}{2} \right]$

(%i3) `invert(matrix([2,3,1], [a,0,0], [1,4,8]));`

(%o3) $\begin{pmatrix} 0 & \frac{1}{a} & 0 \\ \frac{2}{5} & -\frac{3}{4a} & -\frac{1}{20} \\ -\frac{1}{5} & \frac{1}{4a} & \frac{3}{20} \end{pmatrix}$

Maxima on Android

https://play.google.com/store/apps/details?id=jp.yhonda&hl=en_US&gl=US

The screenshot shows the Google Play Store interface. At the top, the navigation bar includes the Google Play logo, a search bar, and tabs for 'Apps', 'Categories', 'Home', 'Top charts', and 'New releases'. The left sidebar contains links to 'My apps', 'Shop', 'Games', 'Kids', 'Editors' Choice', 'Account', 'Payment methods', 'Play Points', 'My subscriptions', 'Redeem', 'Buy gift card', 'My wishlist', 'My Play activity', and 'Parent Guide'. The main content area displays the app 'Maxima on Android' by 'Yasuaki Honda' in the 'Education' category. It has a rating of 4.5 stars from 2,100 reviews and is rated 'Everyone'. A message states 'You don't have any devices'. Below this is an 'Add to Wishlist' button and an 'Install' button. The app icon features a green Android robot with a blue and red stylized 'M' and a dollar sign. Three preview images are shown at the bottom: the first shows the app's version history and license information; the second shows a 3D surface plot of the function $2^{*(v^2-u^2)}$; the third shows the 'Maxima 5.28.0 Manual' with a table of contents and a brief description of the system.

← → ↺ 🏠 🔒 https://play.google.com/store/apps/details?id=jp.yhonda&hl=en_US&gl=US

Google Play Search 🔍

Apps Categories ▾ Home Top charts New releases

My apps
Shop
Games
Kids
Editors' Choice

Account
Payment methods
Play Points **New**
My subscriptions
Redeem
Buy gift card
My wishlist
My Play activity
Parent Guide

Maxima on Android

Yasuaki Honda Education ★★★★★ 2,100

E Everyone

⚠ You don't have any devices

🔖 Add to Wishlist **Install**

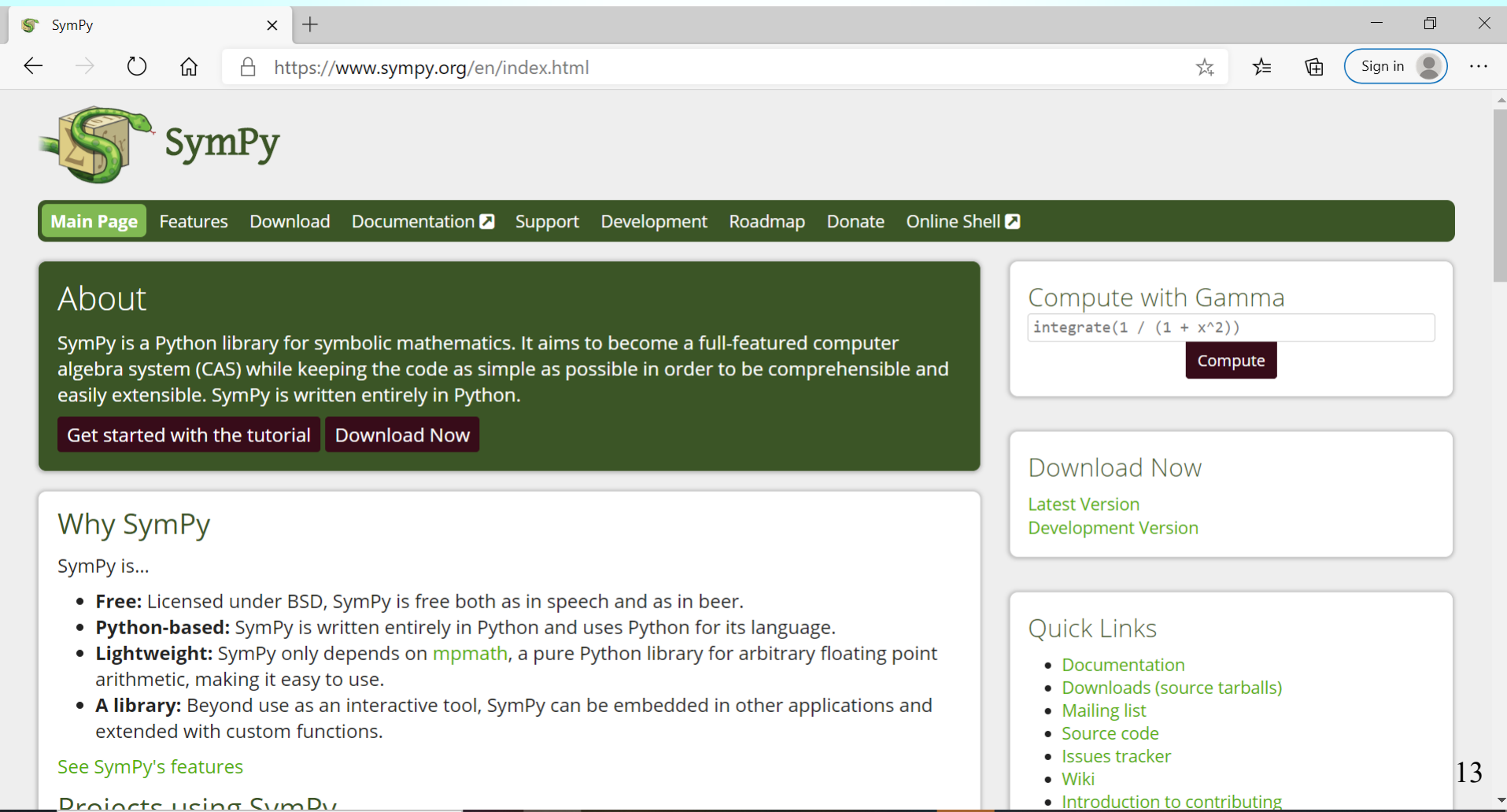
MaximaOnAndroid 9:58
Maxima on Android 1.1, Aug 11, 2012
powered by MathJax 2.0 for math rendering
written by Yasuaki Honda
Maxima 5.27 post <http://maxima.sourceforge.net>
using Lisp ECL 11.1.1
Distributed under the GNU Public License.
See the file COPYING.
Dedicated to the memory of William Schelter.
The function `bug_report()` provides bug reporting information.
(%i1) 'integrate(exp(-x)*sin(x),x,0,inf)=

MaximaOnAndroid 10:38 PM
2^{*(v²-u²)}

MaximaOnAndroid 9:44
[\[Top\]](#) [\[Contents\]](#) [\[Index\]](#) [\[2\]](#)
Maxima 5.28.0 Manual
Maxima is a computer algebra system, implemented in Lisp.
Maxima is derived from the Macsyma system, developed at MIT in the years 1968 through 1982 as part of Project MAC. MIT turned over a copy of the Macsyma source code to the Department of Energy in 1982; that version is now known as DOE Macsyma. A copy of DOE Macsyma was maintained by Professor William F. Schelter of the University of Texas from 1982 until his death in 2001. In 1996, Schelter obtained permission from the Department of Energy to release the DOE Macsyma source code under the GNU Public License, and in 2000 he

SymPy

www.sympy.org/



The screenshot shows the SymPy website homepage in a web browser. The browser's address bar displays the URL `https://www.sympy.org/en/index.html`. The website features a dark green header with navigation links: Main Page, Features, Download, Documentation, Support, Development, Roadmap, Donate, and Online Shell. The main content area is divided into several sections: 'About' (describing SymPy as a Python library for symbolic mathematics), 'Why SymPy' (listing features like being free, Python-based, lightweight, and a library), 'Compute with Gamma' (a calculator interface), 'Download Now' (with links for latest and development versions), and 'Quick Links' (a list of useful resources). The SymPy logo, a green snake coiled around a yellow cube, is visible in the top left corner of the website.

SymPy

https://www.sympy.org/en/index.html

Sign in

SymPy

Main Page Features Download Documentation Support Development Roadmap Donate Online Shell

About

SymPy is a Python library for symbolic mathematics. It aims to become a full-featured computer algebra system (CAS) while keeping the code as simple as possible in order to be comprehensible and easily extensible. SymPy is written entirely in Python.

Get started with the tutorial Download Now

Why SymPy

SymPy is...

- **Free:** Licensed under BSD, SymPy is free both as in speech and as in beer.
- **Python-based:** SymPy is written entirely in Python and uses Python for its language.
- **Lightweight:** SymPy only depends on `mpmath`, a pure Python library for arbitrary floating point arithmetic, making it easy to use.
- **A library:** Beyond use as an interactive tool, SymPy can be embedded in other applications and extended with custom functions.

See SymPy's features

Projects using SymPy

Compute with Gamma

`integrate(1 / (1 + x^2))`

Compute

Download Now

Latest Version
Development Version

Quick Links

- Documentation
- Downloads (source tarballs)
- Mailing list
- Source code
- Issues tracker
- Wiki
- Introduction to contributing

SymPy live

live.sympy.org

live.sympy.org



Main Page Download Documentation Support Development Donate **Online Shell**

These commands were executed:

```
>>> from __future__ import division
>>> from sympy import *
>>> x, y, z, t = symbols('x y z t')
>>> k, m, n = symbols('k m n', integer=True)
>>> f, g, h = symbols('f g h', cls=Function)
```

Documentation can be found at <http://docs.sympy.org/>.

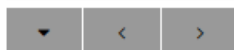
```
>>> diffeq = Eq(f(x).diff(x) + 1000*f(x), 10000)
>>> diffeq
```

$$1000f(x) + \frac{d}{dx} f(x) = 10000$$

```
>>> dsolve(diffeq, f(x))
```

$$f(x) = C_1 e^{-1000x} + 10$$

```
>>>
```



Evaluate

Clear

Fullscreen

Log In

About this page

SymPy Live is SymPy running on the SymPy Engine.

This is just a regular Python shell with the following commands executed:

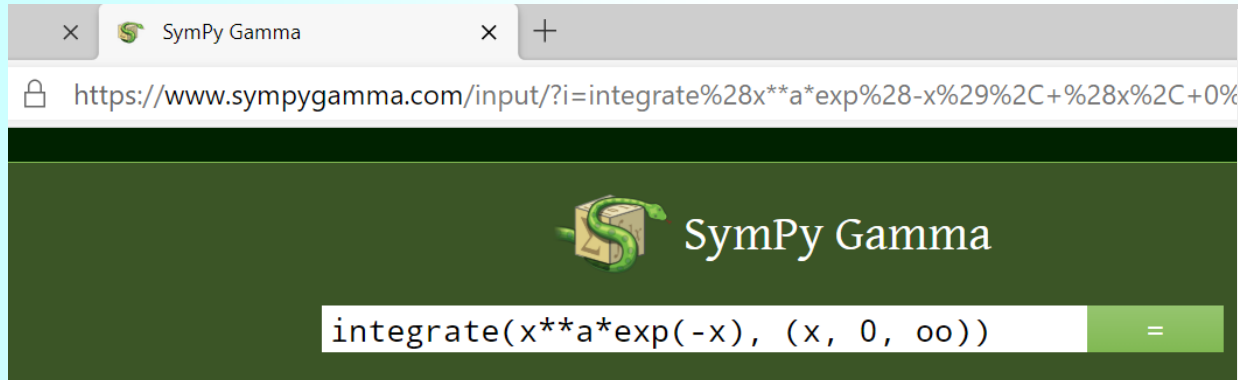
```
>>> from __future__ import division
>>> from sympy import *
>>> x, y, z, t = symbols('x y z t')
>>> k, m, n = symbols('k m n', integer=True)
>>> f, g, h = symbols('f g h', cls=Function)
```

Please note that the Google App Engine has a timeout of 60 seconds for computation due to a quirk in Safari on iOS, which is 58 seconds.

The thumbtack icon is from the Bar Icon Pack and is used under a Creative Commons license. You may use the materials in this presentation without restriction to develop your application.

SymPy Gamma

<https://www.sympygamma.com/>



SymPy:

`integrate (x**a*exp (-x),(x,0,oo ))`

$$\int_0^{\infty} x^a e^{-x} dx$$

Antiderivative forms:

`integrate(x**a*exp(-x), (x, 0, oo))`

$$\begin{cases} \Gamma(a+1) & \text{for } \operatorname{re}(a) > -1 \\ \int_0^{\infty} x^a e^{-x} dx & \text{otherwise} \end{cases}$$

`sympy.integrals.manualintegrate(x**a*exp(-x), (x, 0, oo))`

$$-\Gamma(a+1, x)$$

Examples

Random Example

Arithmetic

Algebra

Trigonometry

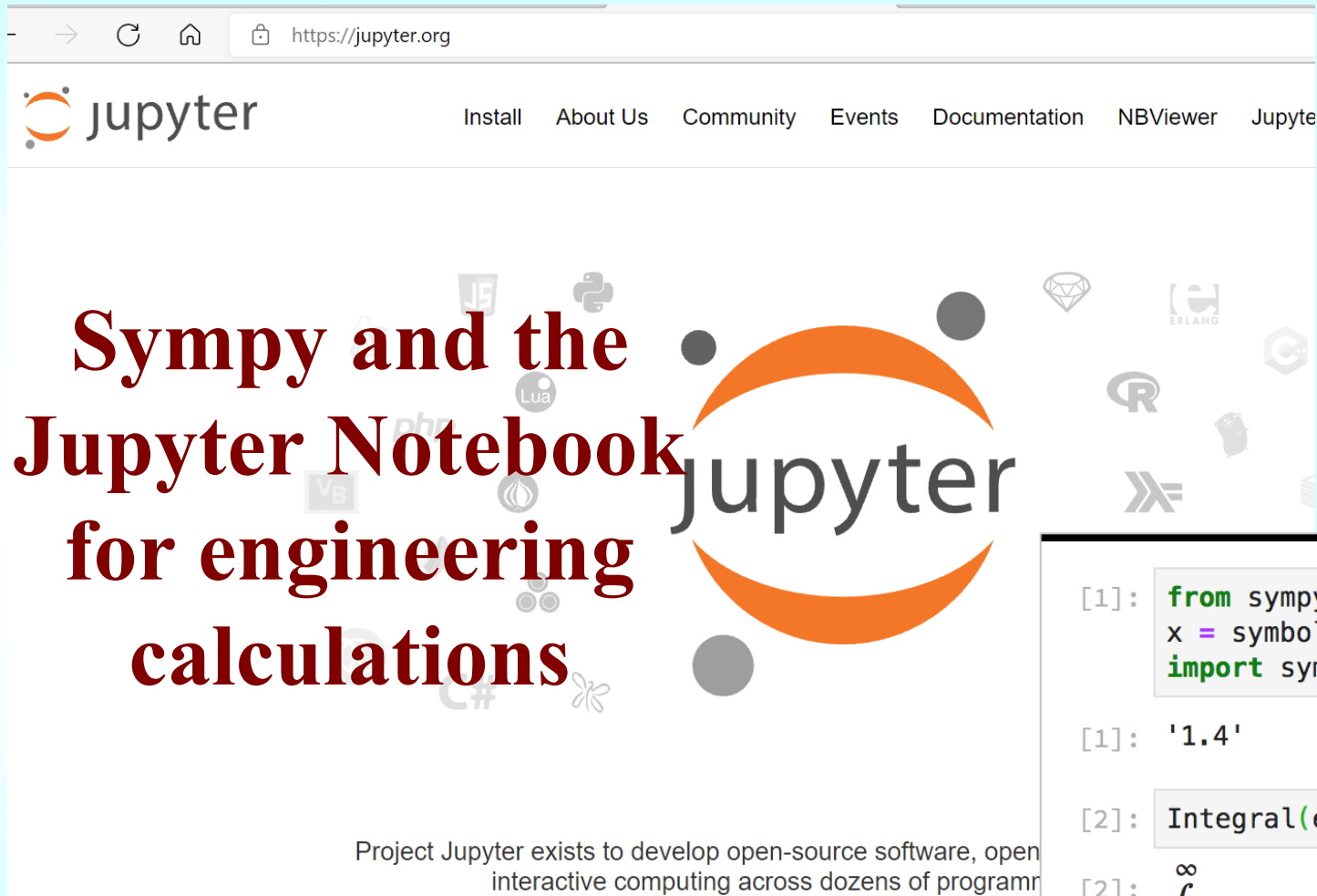
Calculus

Number Theory

Discrete Mathematics

Plotting

Miscellaneous



The image shows the Jupyter website with the URL <https://jupyter.org> in the browser address bar. The Jupyter logo is visible in the top left, and navigation links like 'Install', 'About Us', 'Community', 'Events', 'Documentation', 'NBViewer', and 'Jupyter' are in the top right. A large, semi-transparent text overlay in the center reads 'SymPy and the Jupyter Notebook for engineering calculations'. The Jupyter logo is also visible behind the text. At the bottom, a small text snippet reads: 'Project Jupyter exists to develop open-source software, open interactive computing across dozens of programr'.

Sympy and the Jupyter Notebook for engineering calculations

Project Jupyter exists to develop open-source software, open interactive computing across dozens of programr

```
[1]: from sympy import *  
x = symbols('x')  
import sympy; sympy.__version__  
  
[1]: '1.4'  
  
[2]: Integral(exp(-x**2), (x, -oo, oo))  
[2]: 
$$\int_{-\infty}^{\infty} e^{-x^2} dx$$
  
  
[3]: _.doit()  
[3]: 
$$\sqrt{\pi}$$

```

Google калкулатор

Google

$x \cdot \sin(x)$

Све

Слика

Видео

Мапе

Вести

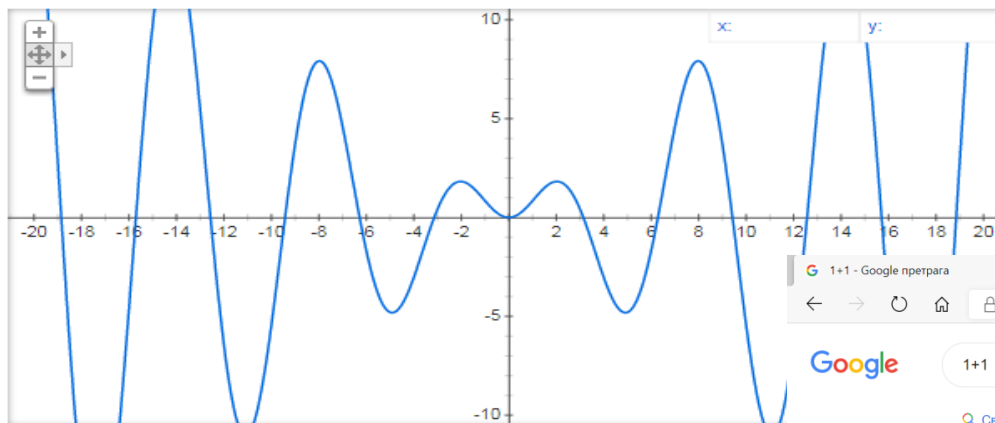
Још

Подешавања

Алатке

Око 3.640.000.000 резултата (0,57 секунде/и)

Графикон за $x \cdot \sin(x)$



www.mytutor.co.uk > Maths > Преведи ову страницу

What is the integral of $x \sin(x) dx$? | MyTutor

Find the following integral: $\int x \sin(x) dx$ This question is a good candidate for parts method, as it is the product of two different 'parts'. R...

www.teachoo.com > category > Преведи ову страницу

Ex 7.6, 1 - Find integration of $x \sin x$ - Chapter 7 Clas

25.09.2018. - Ex 7.6, $\int x \sin x dx$ | Algebraic Trigonometric V
 $g(x) \int dx = f(x) \int 1 \cdot g(x) \dots$

1+1 - Google претпара

← → ↻ 🏠 🔒 https://www.google.com/search?sxsrf=ALeKk02IKn_jfXOK71wFXLI_AsmEQLIWvg%3A16018

Google

1+1

Све Слика Мапе Видео Вести Још Подешавања Алатке

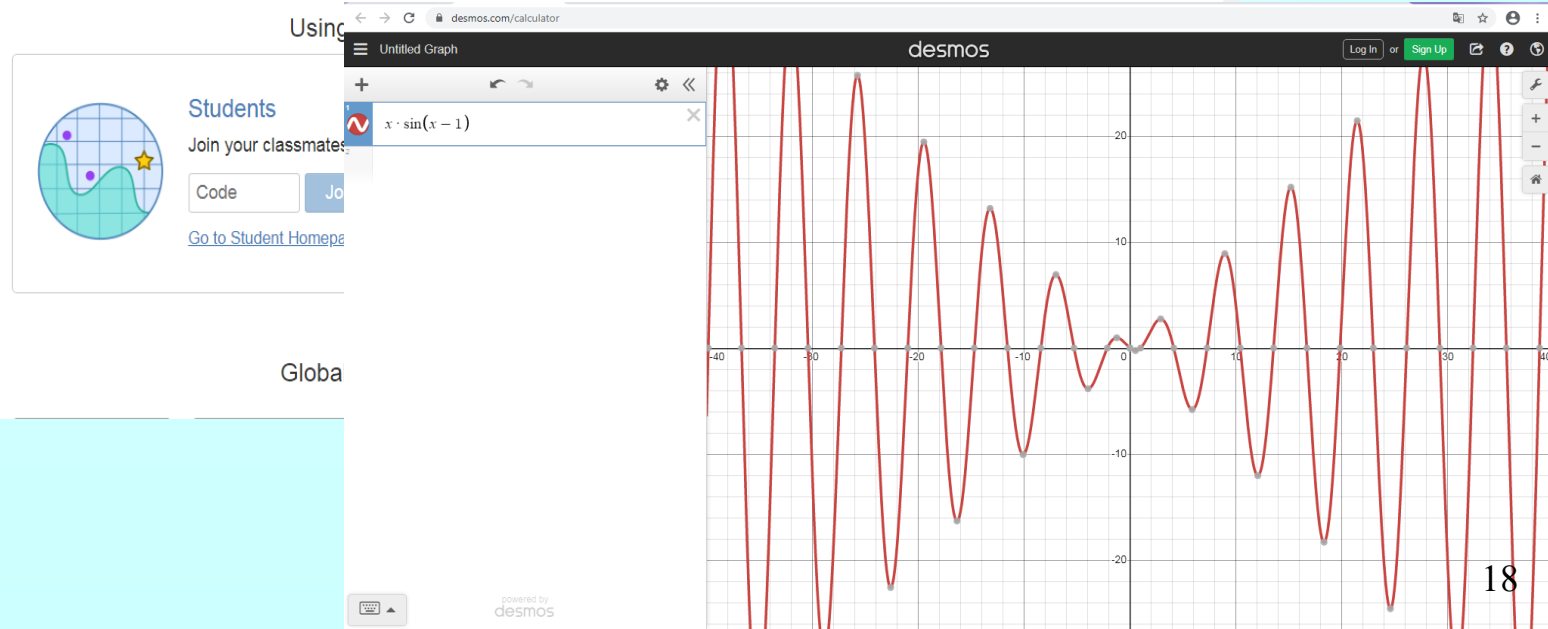
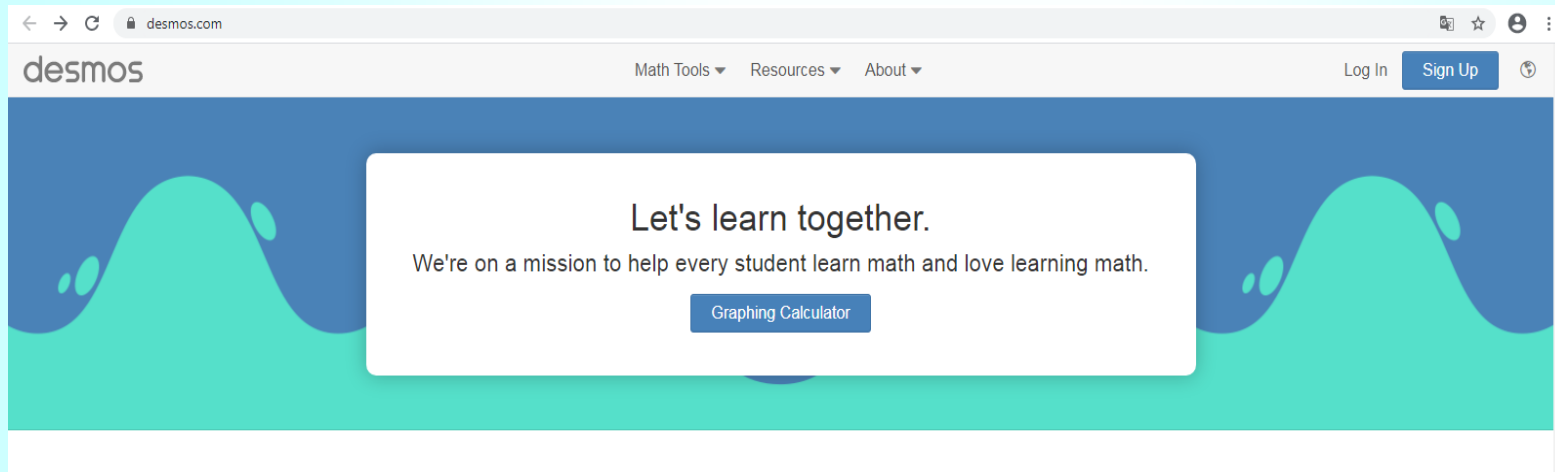
Око 25.270.000.000 резултата (0,72 секунде/и)

1 + 1 = 2

Rad	Deg	x!	(	)	%	AC
Inv	sin	ln	7	8	9	÷
π	cos	log	4	5	6	×
e	tan	√	1	2	3	-
Ans	EXP	x'	0	.	=	+

Graphing Calculator – Desmos

<https://www.desmos.com/calculator>



Symbolab Math Solver - Step by Step calculator

<https://www.symbolab.com/>

Symbolab Math Solver - Step by Step calculator

Try our new Geometry solver! [Got it](#)

Formatting tips »

© EqsQuest 2017

Home What's New Blog About Privacy Terms Popular Problems Help

Most Used Actions

simplify solve for inverse tangent line See All

Symbolab SOLUTIONS GRAPHING PRACTICE GEOMETRY beta NOTEBOOK GROUPS CHEAT SHEETS

Pre Algebra Algebra Pre Calculus Calculus Functions Matrices & Vectors Geometry Trigonometry Statistics Physics

Step-by-Step Calculator Solve problems from Pre Algebra to Calculus step-by-step

Solution [Keep Practicing >](#)

[Show Steps](#)

$\int_0^T x \sin(x) dx = \sin(T) - T \cos(T)$

Steps

$\int_0^T x \sin(x) dx$

Apply Integration By Parts: $u = x, v' = \sin(x)$ [Show Steps](#)

$= [-x \cos(x) - \int -\cos(x) dx]_0^T$

$\int -\cos(x) dx = -\sin(x)$ [Show Steps](#)

$= [-x \cos(x) - (-\sin(x))]_0^T$

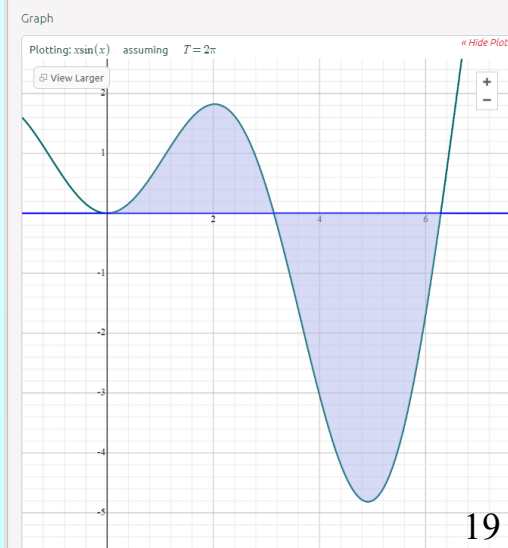
Simplify

$= [-x \cos(x) + \sin(x)]_0^T$

Compute the boundaries: $[-x \cos(x) + \sin(x)]_0^T = \sin(T) - T \cos(T)$ [Show Steps](#)

$= \sin(T) - T \cos(T)$

[Got a different answer? Check if it's correct](#) [Verify](#)



SciLab

<https://www.scilab.org/>

The image is a composite showing the SciLab website and its software interface. The top part is a screenshot of the website <https://www.scilab.org/> in a web browser. The website features the SciLab logo, a search bar, and navigation links: Download, Software, Tutorials, Applications, Services, Cloud, and About. The main content area promotes the download of SciLab 6.1.0 for Windows, Linux, and Mac OS X, highlighting it as open-source software for numerical computation. Below the website screenshot, there are several overlapping images of the SciLab software interface. One image shows the 'Toolboxes' menu with options: Numerical Analysis, Data visualization, Algorithm development, and Application development. Another image shows a 3D surface plot of a function. A third image shows a code editor with a script for a control system simulation. A fourth image shows a 'Control Panel' window with various settings. A fifth image shows a 'My SciLab App' window with a 'Control Panel' and a 'Data Panel'. A sixth image shows a 'SciLab CONFERENCE' logo. A seventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A eighth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A ninth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A tenth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A eleventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twelfth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirteenth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fourteenth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifteenth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixteenth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventeenth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A eighteenth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A nineteenth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twentieth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-first image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-second image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-third image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-fourth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-fifth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-sixth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-seventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-eighth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A twenty-ninth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirtieth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-first image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-second image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-third image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-fourth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-fifth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-sixth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-seventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-eighth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A thirty-ninth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fortieth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-first image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-second image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-third image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-fourth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-fifth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-sixth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-seventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-eighth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A forty-ninth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fiftieth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-first image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-second image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-third image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-fourth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-fifth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-sixth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-seventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-eighth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A fifty-ninth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixtieth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-first image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-second image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-third image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-fourth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-fifth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-sixth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-seventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-eighth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A sixty-ninth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventieth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-first image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-second image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-third image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-fourth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-fifth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-sixth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-seventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-eighth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A seventy-ninth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eightieth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-first image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-second image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-third image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-fourth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-fifth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-sixth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-seventh image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-eighth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. An eighty-ninth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A ninetieth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values. A hundredth image shows a 'SciLab 6.1.0 Console' window with a list of variables and their values.

Download SciLab 6.1.0
Windows, Linux and Mac OS X
Open source software for numerical computation

Toolboxes

- Numerical Analysis
- Data visualization
- Algorithm development
- Application development

Image Processing & Computer Vision
Model Order Reduction
SciLab Code Generator
Signal acquisition & instrument control
Functional Mock-Up Interface (FMI) for Model-Exchange & Co-Simulation

Xcos

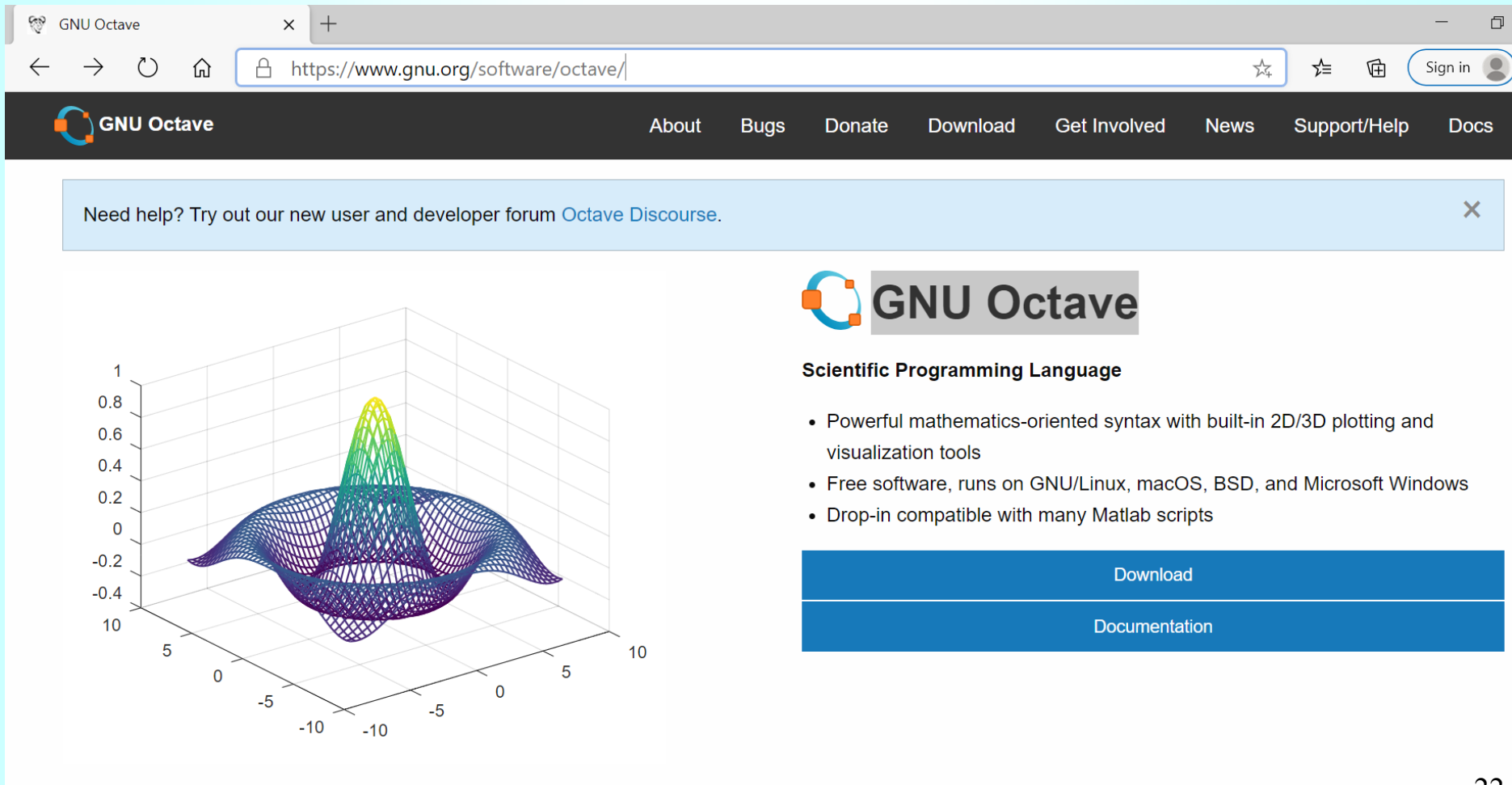
<https://www.scilab.org/software/xcos>

The screenshot shows the Xcos website interface. At the top, the Scilab logo is on the left, and a navigation menu includes 'Download', 'Software', 'Tutorials', 'Applications', 'Services', 'Cloud', and 'About'. A search bar is on the right. Below the navigation bar, the word 'Xcos' is prominently displayed in the center. At the bottom, there are four example diagrams illustrating different applications of Xcos:

- Signal Processing:** A diagram showing three 'Signal Builder' blocks connected to a 'Scope' block. A note below states: 'blocks can also emit continuous or discrete activation signals'.
- Control Systems:** A diagram titled 'Water-Tank System' showing a feedback control loop with a 'Controller' block and a 'Robot-Tank System' block.
- Mechanics/Thermodynamics:** A diagram showing a mechanical system with multiple masses and springs.
- Electronics:** A diagram showing an electronic circuit with various components like resistors, capacitors, and a power source.

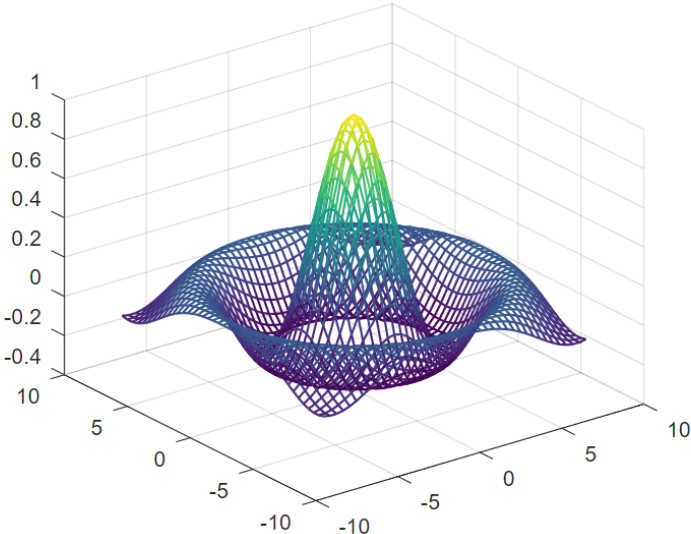
GNU Octave

<https://www.gnu.org/software/octave/>



GNU Octave

Need help? Try out our new user and developer forum [Octave Discourse](#).



GNU Octave

Scientific Programming Language

- Powerful mathematics-oriented syntax with built-in 2D/3D plotting and visualization tools
- Free software, runs on GNU/Linux, macOS, BSD, and Microsoft Windows
- Drop-in compatible with many Matlab scripts

[Download](#)

[Documentation](#)

Micro-Cap

<http://www.spectrum-soft.com/index.shtml>

The screenshot displays the Spectrum Software website's Micro-Cap 12 download page. The browser's address bar shows the URL <http://www.spectrum-soft.com/index.shtml>. The website header includes the Spectrum Software logo and a navigation menu with links to Home, Products, Download, Newsletters, Support, and Contact Us. A banner for "Industrial Strength Simulation" is visible. On the left, a sidebar lists navigation options: MC12 Revision History, Product Information, Features Tour, and New MC12 Features. Below this, a "News" section states that Micro-Cap is now free and that technical support will be available for 90 days via email at [Support](mailto:support@spectrum-soft.com). The main content area features a circuit diagram with various components labeled (e.g., VCC, 15V, V2, QPL, Q2, Q10, R18, 10, 1K, R17, Q21, 30PF, C3, R1, 1K, QNL, Q4, %B, VCC, 1K, R16, Q5, VCC, Q11, Q1, 40K, R10). A prominent gold coin graphic with "SPECTRUM SOFTWARE" and "12" is centered. A red banner above the circuit states: "MC10, MC11, and MC12 are now free and need no key." On the right, two informational boxes are present: "About Spectrum" and "Help".

Spectrum Software

Home Products Download Newsletters Support Contact Us

Industrial Strength Simulation

Select :

- MC12 Revision History
- Product Information
- Features Tour
- New MC12 Features

News:
Effective 7/4/2019, Spectrum Software is closed. Micro-Cap is now free.
Technical support will be available for at least 90 days via email at [Support](mailto:support@spectrum-soft.com).
You can download the latest versions of Micro-Cap here: [Download](#)
You can choose either the executable program or the entire installation CD for MC10, MC11, and MC12. If you have an earlier version, download and use MC12. These new versions do not require

MC10, MC11, and MC12 are now free and need no key.

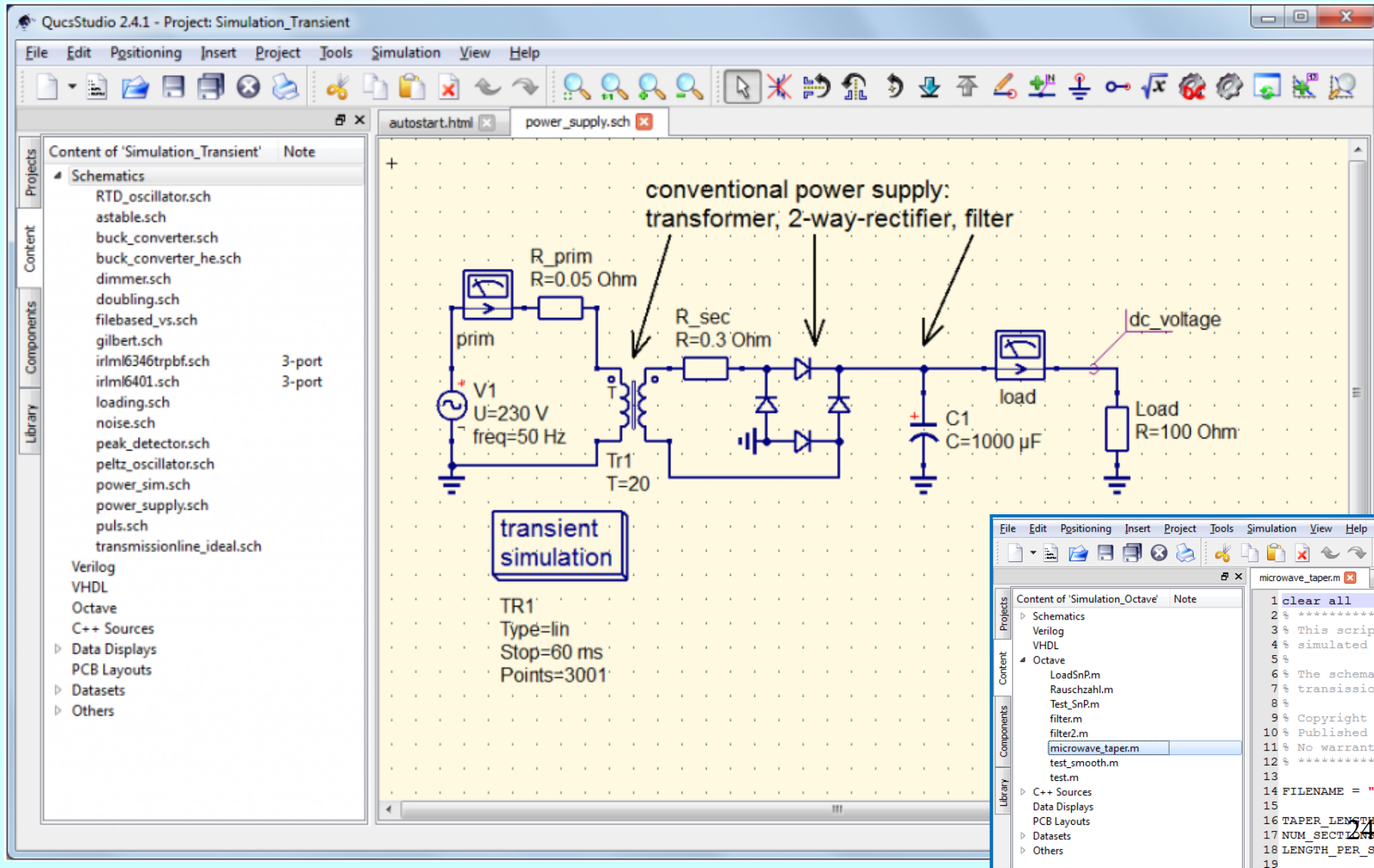
About Spectrum
Spectrum Software was founded in February of 1980 to provide software for personal computers. Initially, the company concentrated on providing software for Apple II systems.
One of the earliest products was Logic Designer and Simulator. Released in June 1980, this product was the first integrated circuit editor and logic simulation system available for personal computers. More...

Help
Search our newsletter issues which contain application notes for Micro-Cap or view our Frequently Asked Questions section for common questions that arise with Micro-Cap use.

QucsStudio

<http://qucsstudio.de>

*Octave engine
inside!*



CircuitLab

<https://www.circuitlab.com/>

The image shows a screenshot of the CircuitLab website and a tutorial circuit diagram. The website header includes the CircuitLab logo, a search bar, and user information for 'milka.potrebic'. The main heading is 'Circuit simulation and schematics.' followed by the subheading 'Build and simulate circuits right in your browser.' A list of features is provided: 'Design with our easy-to-use schematic editor.', 'Analog & digital circuit simulations in seconds.', 'Professional schematic PDFs, wiring diagrams, and plots.', and 'No installation required! Launch it instantly with one click.' A green button labeled 'Launch CircuitLab' is present, along with a link to 'or watch a quick demo video →'. Below this, a video player shows a 'Getting Started with CircuitLab' tutorial. The bottom section displays a circuit diagram for a 1-minute tutorial. The circuit consists of a 1 kHz square wave voltage source (V1) connected in series with a 100 Ω resistor (R1). This is followed by a parallel combination of a 10 μF capacitor (C1) and a 100 Ω resistor (R2). The output is taken from the node between R1 and C1. A list of five steps for the tutorial is provided, along with a link to the 'Circuit Design and Analysis' textbook.

CircuitLab

Search

milka.potrebic
inbox (0) · settings · log out

Circuit simulation and schematics.

Build and simulate circuits right in your browser.

- Design with our easy-to-use schematic editor.
- Analog & digital circuit simulations in seconds.
- Professional schematic PDFs, wiring diagrams, and plots.
- No installation required! Launch it instantly with [one click](#).

► **Launch CircuitLab**

[or watch a quick demo video →](#)

Getting Started with CircuitLab

Гледајте К... Дели

CircuitLab

File Edit Run Help

Press 'F' to search

Essentials

DC Sources

Passive Elements

Voltage Signal Sources

Current Signal Sources

Operational Amplifiers

Diodes

V1 square 1 kHz

R1 100 Ω

C1 10 μF

out

1-minute Tutorial

1. Click and drag a wire from the right of R1 to the top of C1.
2. Click "Simulate."
3. Click "Run Time-domain Simulation" and look at the filter output V(out)
4. Double-click R1, change it to "1k", and simulate again.
5. Run the Frequency-Domain simulation and see the Bode plot.

Textbook **NEW!**

Electronic systems with CircuitLab's free, 1k.

[Circuit Design and Analysis](#)

Powered by CircuitLab > to learn about the full version, please visit www.circuitlab.com/

148%

DoCircuits Circuit Simulator

Do Circuits

Blog Learn Download Login



Experience virtual labs for electronics on browser. Work with real looking components and devices. Build circuits - run, analyse and save them in easy steps.
Reach us at info@docircuits.com

Launch Online Labs Download Free Trial



Get started with some sample circuits !

Colpitts Oscillator

This circuit generates a continuous sinusoidal signal. The tank circuit is made up of two capacitors and an inductor in parallel

Active Low Pass Filter

This circuit made up using op-amp and will allow signals lower than a particular cut off frequency to pass though and attenuates those above it.

Active High Pass Filter

This circuit uses an op-amp as the active element and allows only the input signals with frequency higher than the cut-off frequency and rejects those below it.

Clipper

This is a wave-shaping circuit. It will modify the shape of the signal by "clipping" off a portion of it. When the diode is forward-biased, the voltage across it will be a constant 0.7 V. And when the diode is reverse-biased, it will be open. Thus the output will follow the input when it is forward biased.

Clamper

This circuit is another level shape. The wave will be shifted to a positive or negative level.

Fig. 1: DoCircuits - Home

26

ngSpice

<http://ngspice.sourceforge.net/>

← → ↻ 🏠 ⚠ Not secure | ngspice.sourceforge.net



MIXED MODE - MIXED LEVEL
CIRCUIT SIMULATOR
BASED ON BERKELEY'S SPICE3F5

NGSPICE SUMMARY

Home News Screenshots Download Documentation Tutorials Extras/Options Applications Development Simulation Environments Recipes

Ngspice Home

Home
What is ngspice ?
Features, Extras & Options
F.A.Q.
Tutorials
Sourceforge Developer Pages

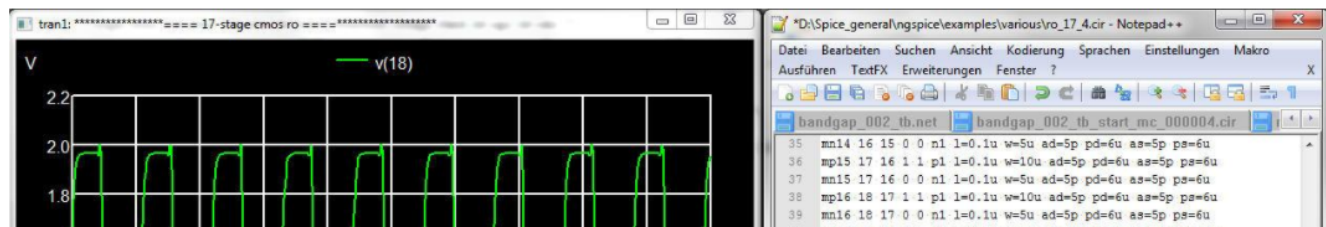
ngspice - open source spice simulator

ngspice is the open source spice simulator for electric and electronic circuits.

Such a circuit may comprise of JFETs, bipolar and MOS transistors, passive elements like R, L, or C, diodes, transmission lines and other devices, all interconnected in a netlist. Digital circuits are simulated as well, event driven and fast, from single gates to complex circuits. And you may enter the combination of both analog and digital as a mixed-signal circuit.

ngspice offers a wealth of device models for active, passive, analog, and digital elements. Model parameters are provided by our [collections](#), by the [semiconductor device manufacturers](#), or from [semiconductor foundries](#). The user adds her circuits as a netlist, and the output is one or more graphs of currents, voltages and other electrical quantities or is saved in a data file.

ngspice does not provide schematic entry. Its input is command line or file based. There are however [third party](#) interfaces available.



<https://www.ti.com/tool/TINA>



Search

Products

Applications

Design resources

Home / Design resources

TINA-TI

SPICE-based analog simulation program

Overview

Downloads

Technical documentation

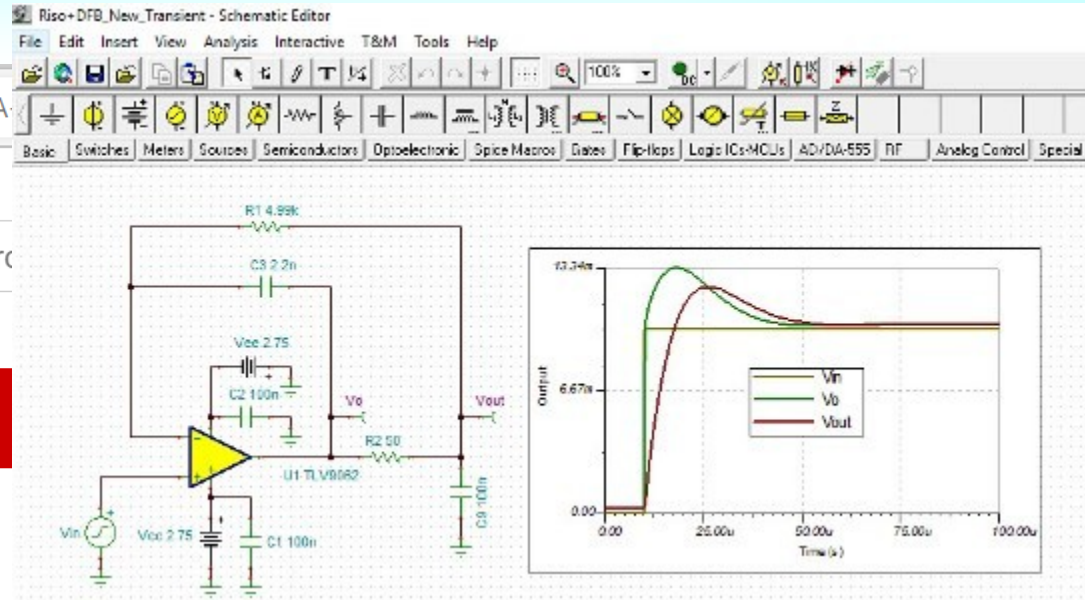
Support & training

TINA-TI

<https://www.ti.com/tool/TINA-TI>

Overview

TINA-TI provides all the conventional DC, transient and frequency domain analysis of SPICE and much more. TINA has extensive post-processing capability that allows you to format results the way you want them. Virtual instruments allow you to



Electric Circuit Studio



https://play.google.com/store/apps/details?id=com.ecstudiosystems.electriccircuitstudio&hl=en_US&gl=US



Google Play

Search



Apps

Categories ▾

Home

Top charts

New releases

My apps

Shop

Games

Kids

Editors' Choice

Account

Payment methods

Play Points

New

My subscriptions

Redeem

Buy gift card

My wishlist

My Play activity

Parent Guide



Electric Circuit Studio

ECStudio Systems Education

★★★★★ 2,709

Everyone

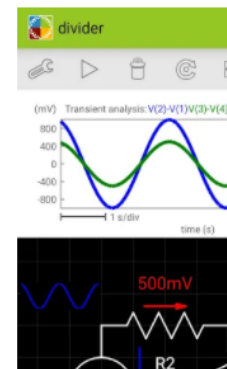
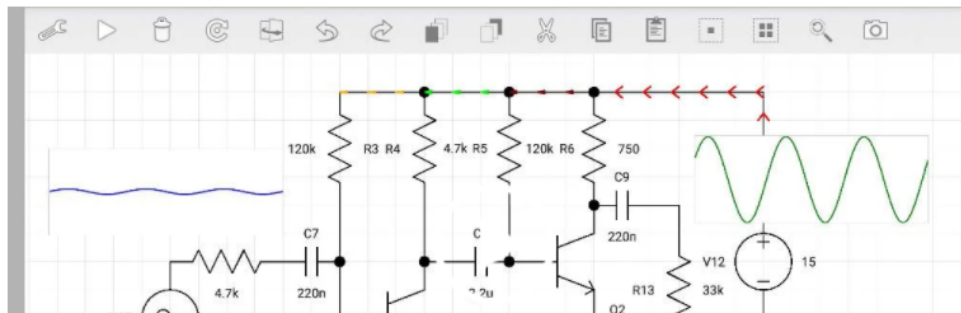
Contains Ads

⚠ You don't have any devices



Add to Wishlist

Install



CircuitSafari SPICE Simulator

← → ↻ 🏠 https://play.google.com/store/apps/details?id=com.logipipe.circuitsafari&hl=en_US&gl=US



Search



Apps

Categories ▾

Home

Top charts

New releases

My apps

Shop

Games

Kids

Editors' Choice

Account

Payment methods

Play Points

New

My subscriptions

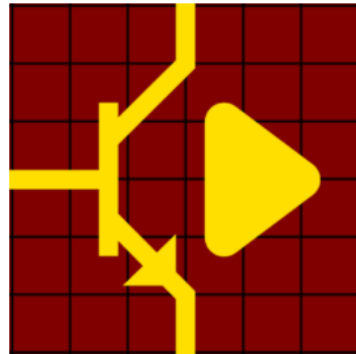
Redeem

Buy gift card

My wishlist

My Play activity

Parent Guide



CircuitSafari SPICE Simulator (Early Access)

Logipipe, LLC Productivity

E Everyone

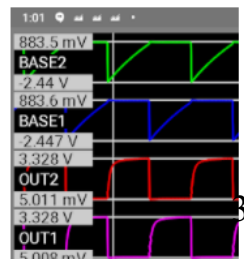
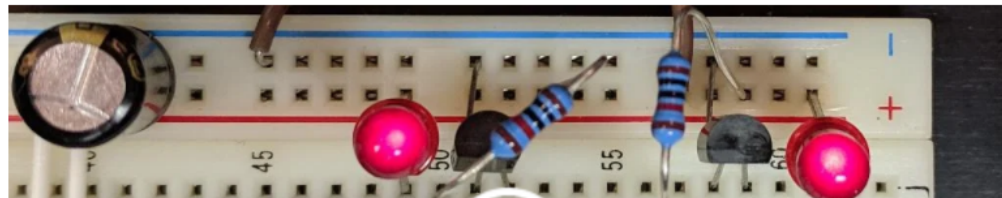
Offers in-app purchases

ⓘ This app is in development. It may be unstable.

⚠ You don't have any devices

🔖 Add to Wishlist

Install



Julia Programming Language

[Download](#)[Documentation](#)[Learn](#)[Blog](#)[Community](#)[Contribute](#)[JSoC](#)

44,131



Sponsor

The Julia Programming Language

[Download](#)[Documentation](#)

44,131

Julia in a Nutshell

Fast

Julia was designed for [high performance](#). Julia programs automatically compile to efficient native code via LLVM, and support [multiple platforms](#).

Dynamic

Julia is [dynamically typed](#), feels like a scripting language, and has good support for [interactive use](#), but can also optionally be separately compiled.

Reproducible

[Reproducible environments](#) make it possible to recreate the same Julia environment every time, across platforms, with [pre-built binaries](#).

Composable

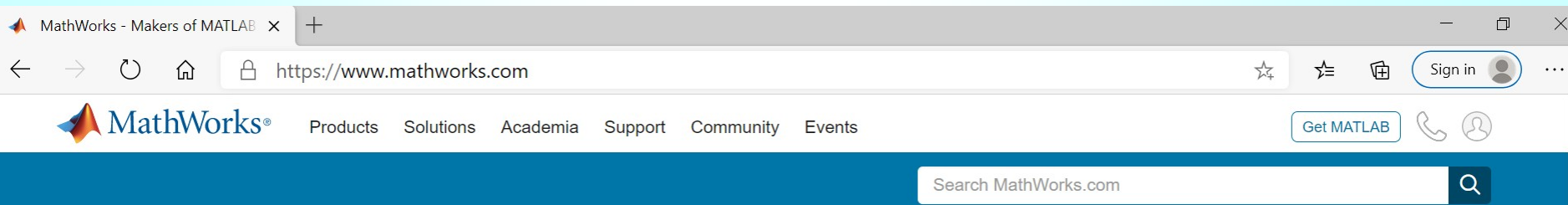
Julia uses [multiple dispatch](#) as a paradigm, making it easy to express many object-oriented and [functional](#) programming patterns. The talk on the [Unreasonable Effectiveness of Multiple Dispatch](#) explains why it works so well.

General

Julia provides [asynchronous I/O](#), [metaprogramming](#), [debugging](#), [logging](#), [profiling](#), a [package manager](#), and more. One can build entire [Applications and Microservices](#) in Julia.

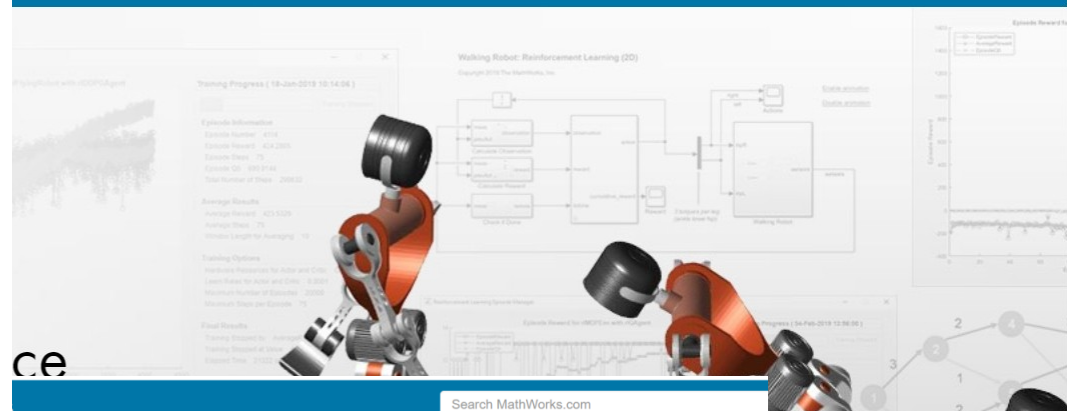
Open source

Julia is an open source project with over 1,000 contributors. It is made available under the [MIT license](#). The [source code](#) is available on GitHub.

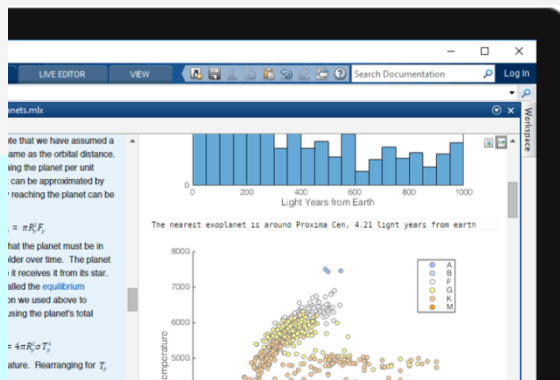


Millions of Engineers and Scientists Trust MATLAB

MATLAB® combines a desktop environment tuned for iterative analysis and design processes with a programming language that expresses matrix and array mathematics directly. It includes the [Live Editor](#) for creating scripts that combine code, output, and formatted text in an executable notebook.



Design AI models and AI-driven systems



Products by Category Alphabetical

Trial software

MATLAB® PRODUCT FAMILY

MATLAB

Parallel Computing

Parallel Computing Toolbox
MATLAB Parallel Server

Math and Optimization

Curve Fitting Toolbox
Optimization Toolbox
Global Optimization Toolbox
Symbolic Math Toolbox
Mapping Toolbox

SIMULINK® PRODUCT FAMILY

Simulink

System Composer

Event-Based Modeling

Stateflow
SimEvents

Physical Modeling

Simscape
Simscape Driveline
Simscape Electrical
Simscape Fluids

SERVICES

Software Maintenance
Training
Consulting

LICENSE TYPES

Industry Use
Student Use
University Use
Startup Use
Primary and Secondary School Use
Home Use



MATLAB

 <http://www.mathworks.com/>

MATLAB R2019a

HOME PLOTS APPS

New Script New Live Script New Open Find Files Import Data Save Workspace New Variable Open Variable Clear Workspace Analyze Code Run and Time Favorites Clear Commands Simulink Layout Preferences Set Path Add-Ons Help Community Request Support Learn MATLAB

FILE VARIABLE CODE ENVIRONMENT RESOURCES

Current Folder: C:\Program Files\Polyspace\R2019a\bin

Command Window: New to MATLAB? See resources for [Getting Started.](#)

Workspace: Name Value

System target file browser: Simulink2DSHplus

System target file: Description: asap2.tlc ASAM-ASAP2 Data Definition Target dsh.tlc Borland C++ DSHplus Real-Time Target with dsh.tlc Visual C/C++ DSHplus Real-Time Target with ert.tlc RTW Embedded Coder (no auto configuration) ert.tlc RTW Embedded Coder (auto configures for C) ert.tlc C++ Project Makefile only for the real-time target ert.tlc C++ Project Makefile only for the real-time target

Full name: Template makefile Makefile

Cancel Help Apply

Simulink2DSHplus

MathWorks

MATLAB®

Professional License

© 1984-2019 The MathWorks, Inc. Protected by U.S. and international patents. See mathworks.com/patents. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

MathWorks® R2019a

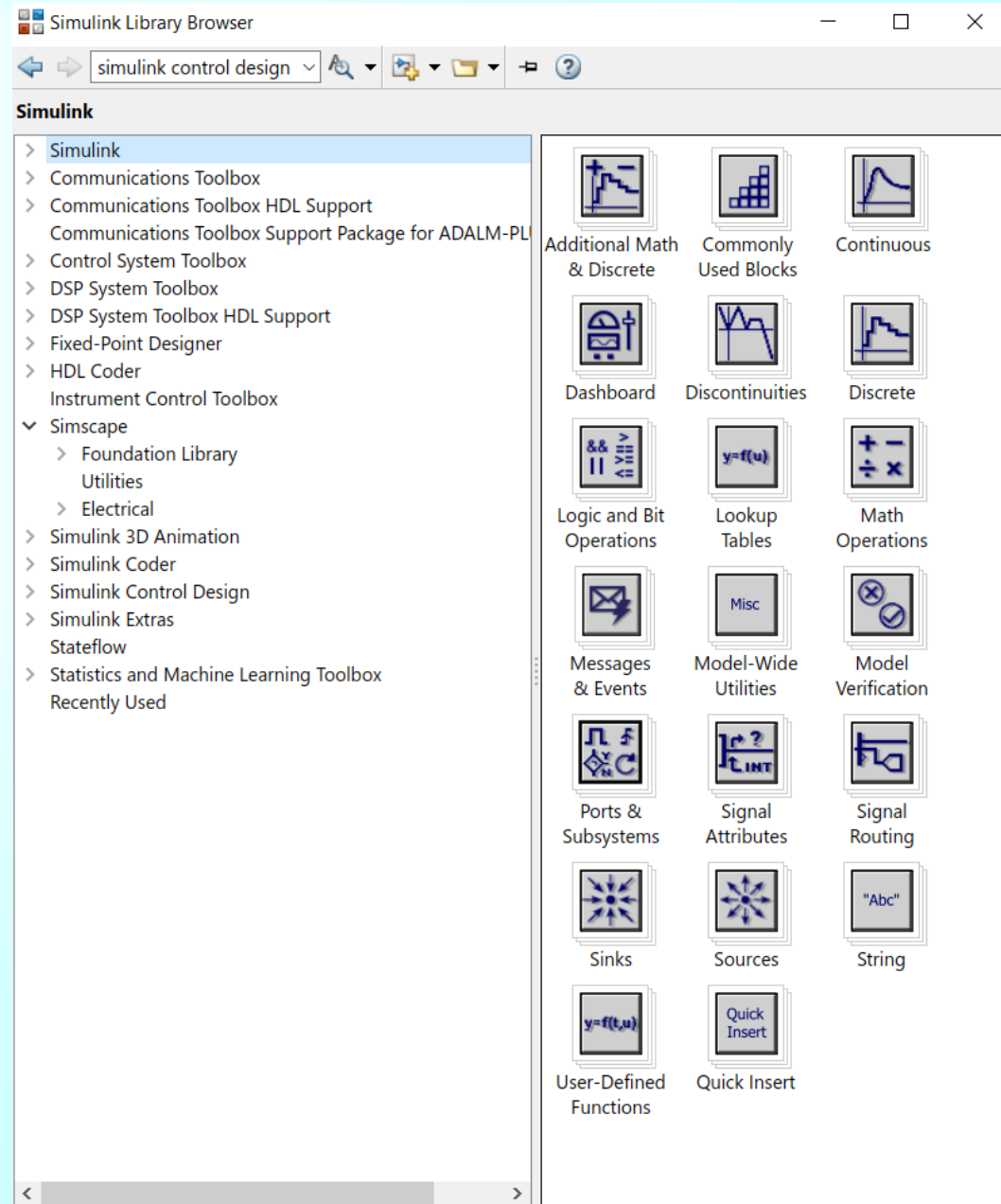
Ready

Toolboxes...

MATLAB

Simulink

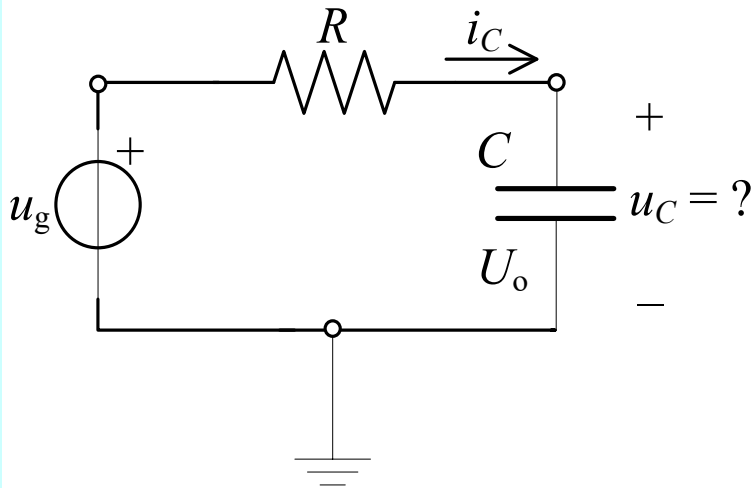
- Алатка за нумеричку анализу математичких модела
- Систем се дефинише графички, уметањем блокова и њиховим повезивањем у мрежу



Simulink > Simscape

- Библиотека за симулацију и нумеричку анализу мултидоменских проблема (механика, електродинамика, оптика)
- Пример једноставног електричног кола:

коло је образовано у тренутку $t_0 = 0$

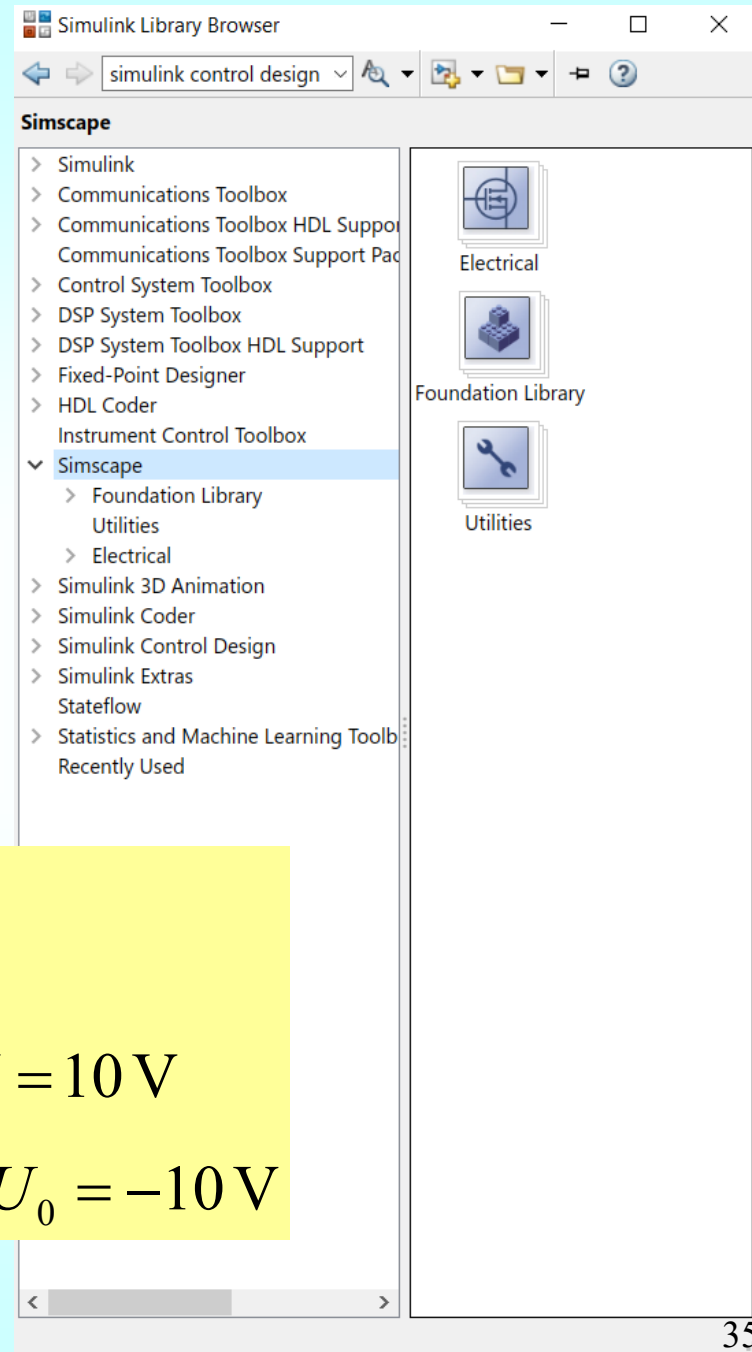


$$R = 1 \text{ k}\Omega$$

$$C = 1 \mu\text{F}$$

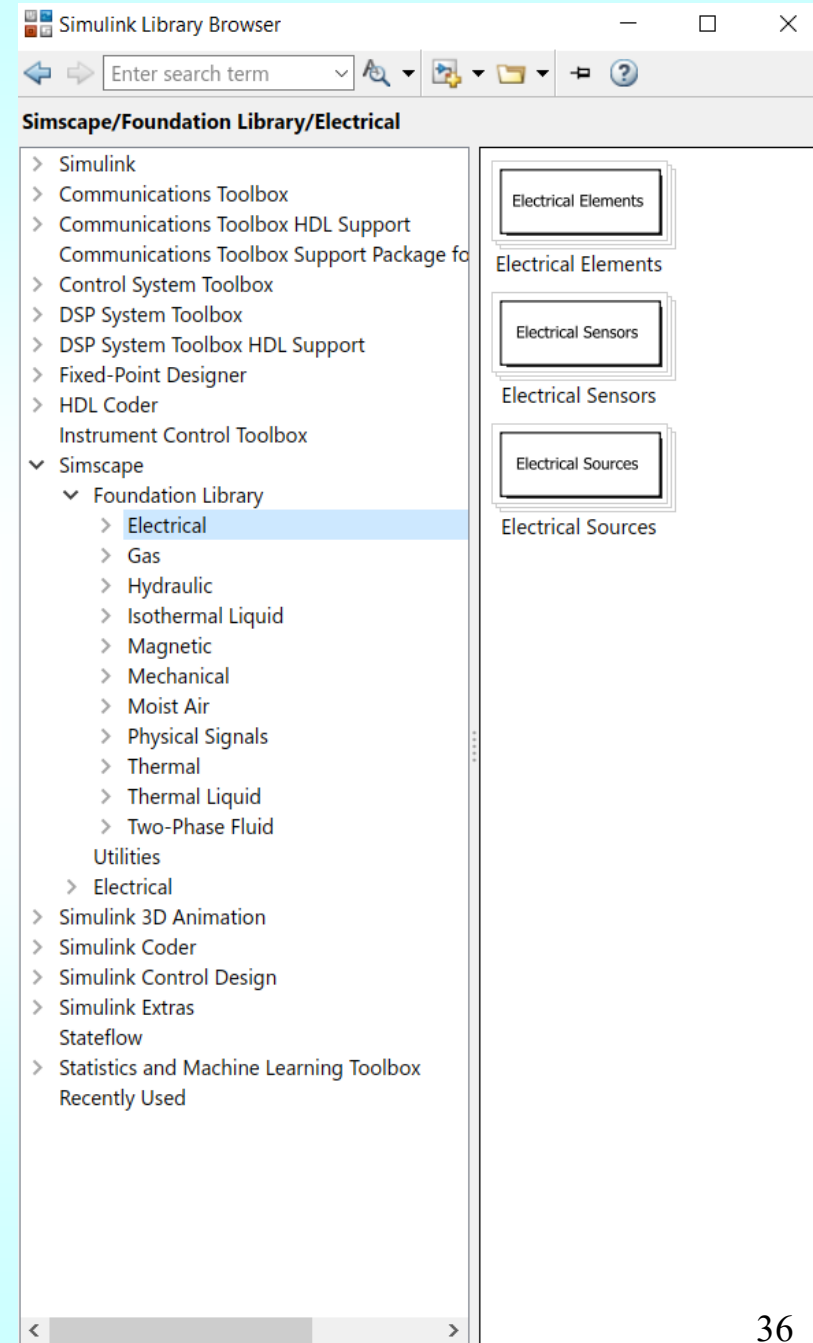
$$u_g(t) = U = 10 \text{ V}$$

$$u_C(t_0^-) = U_0 = -10 \text{ V}$$



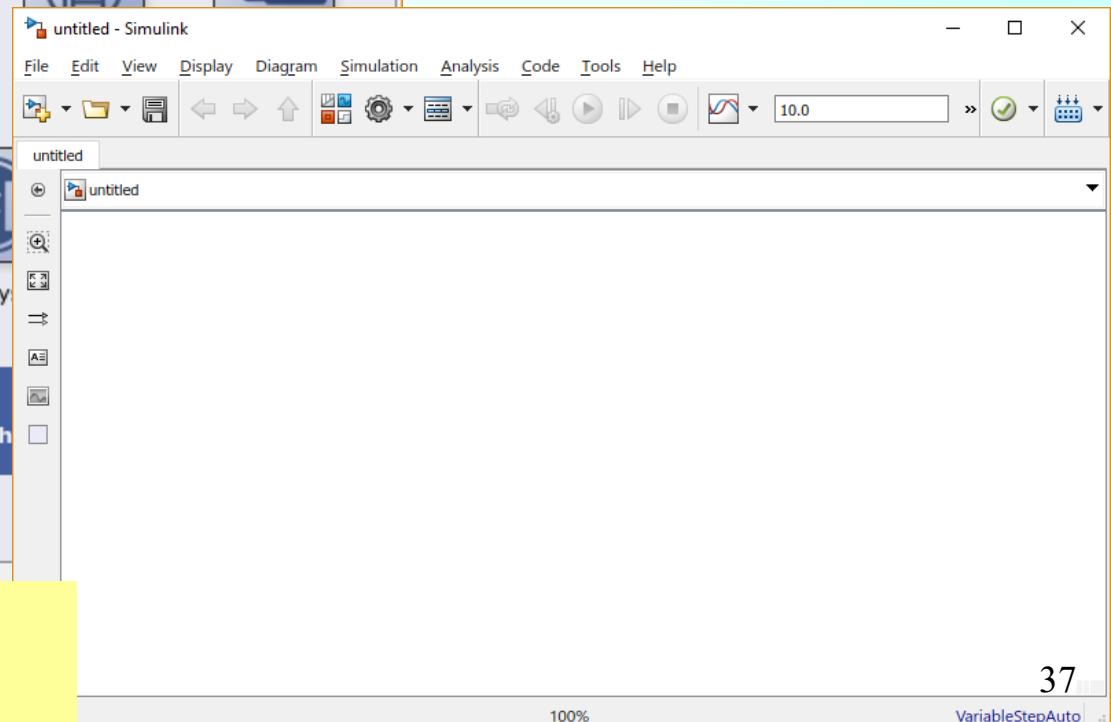
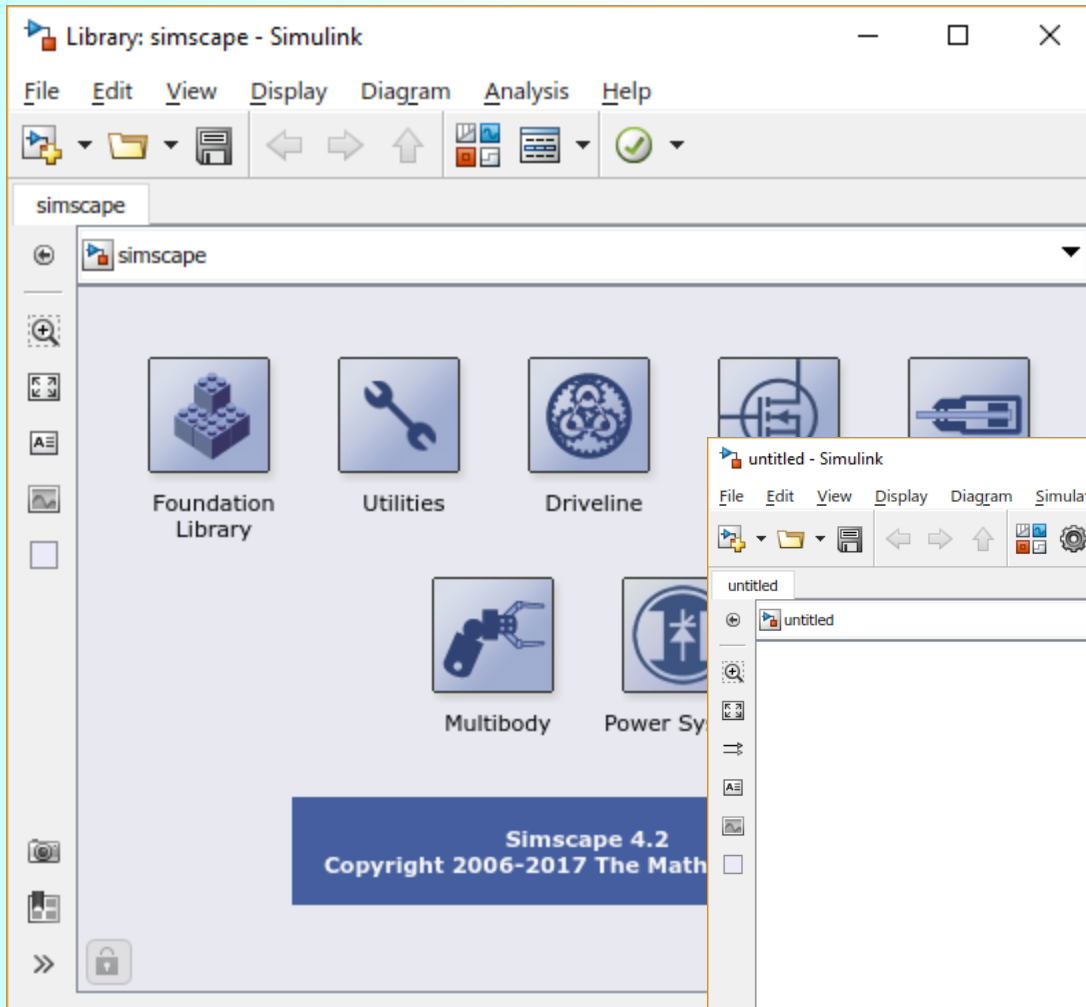
Simulink > Simscape > Foundation Library > Electrical

- Симулација линеарних и нелинеарних електричних кола

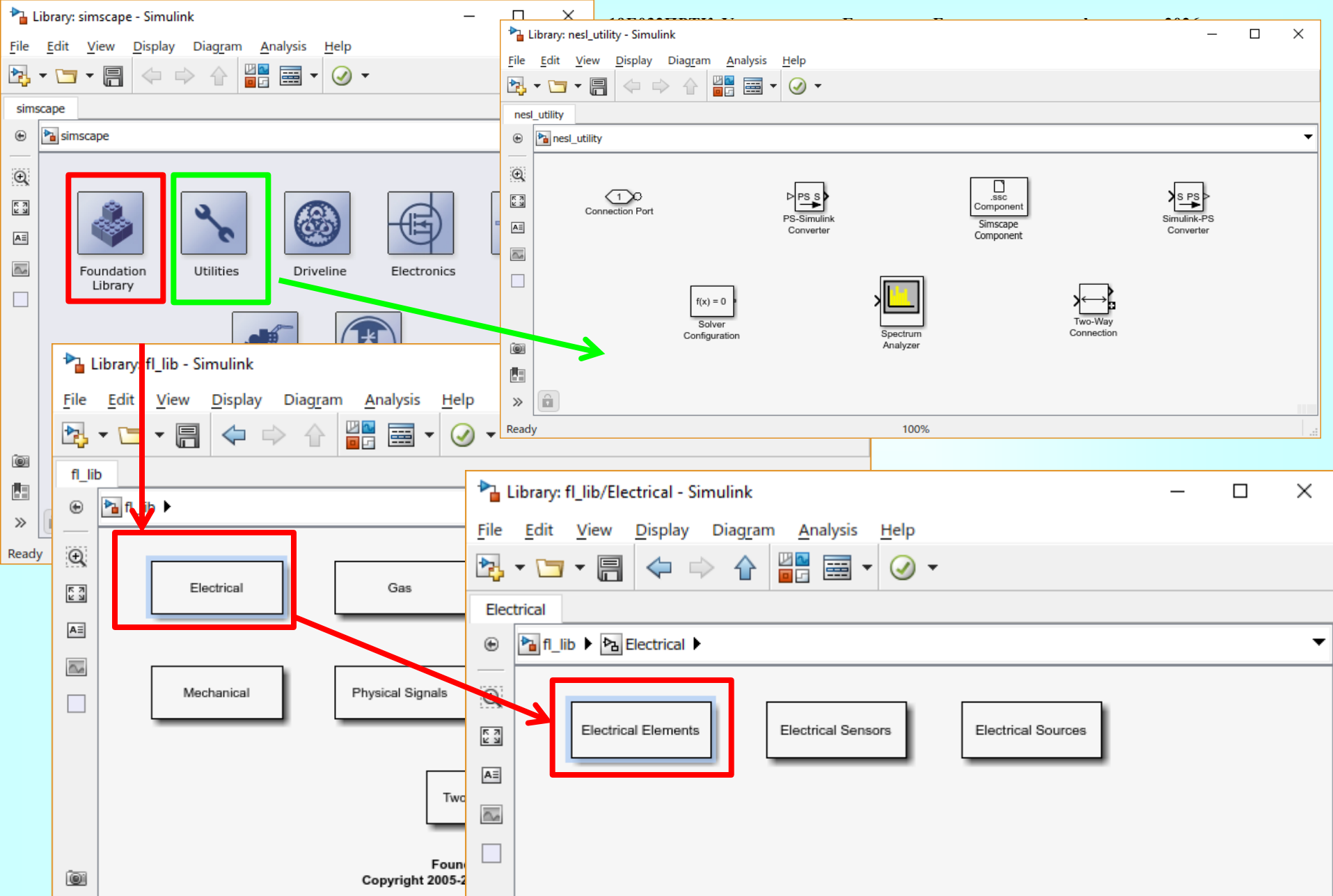


**MATLAB > Simscape >
Foundation Library, Utilities**

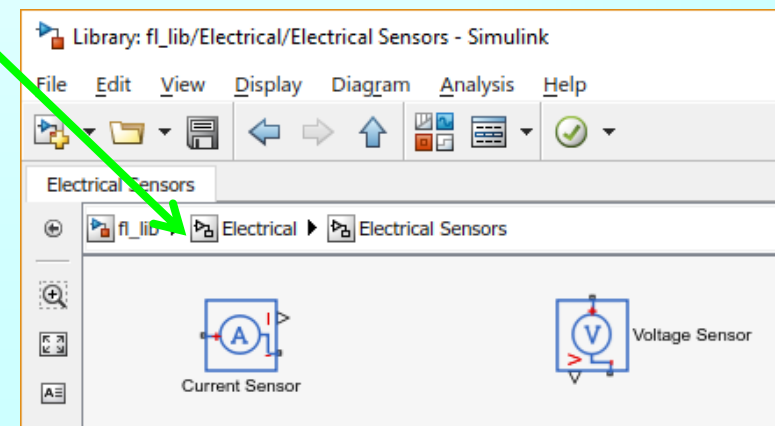
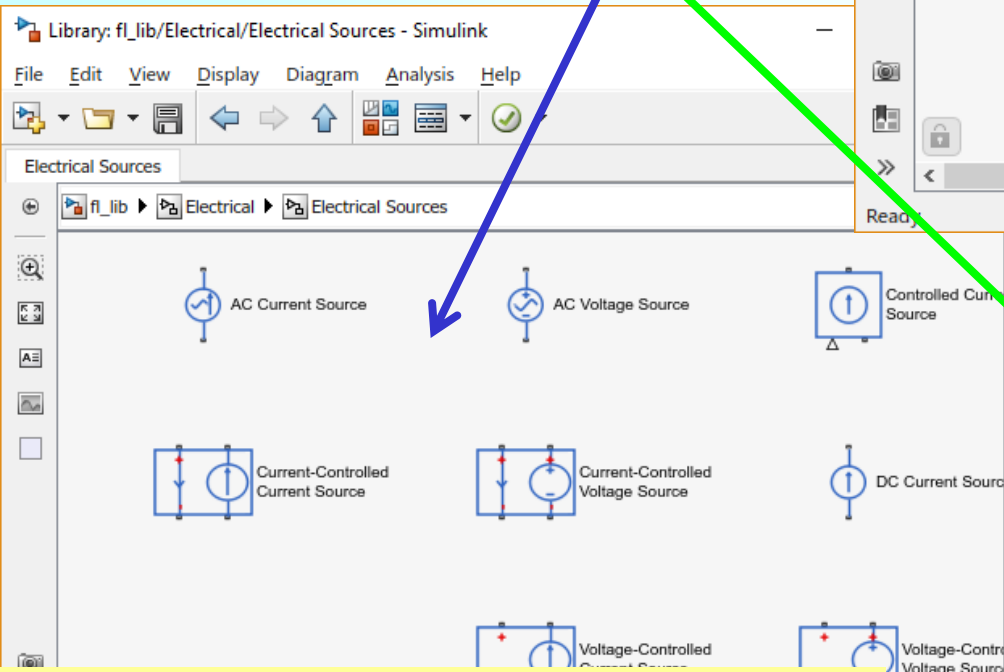
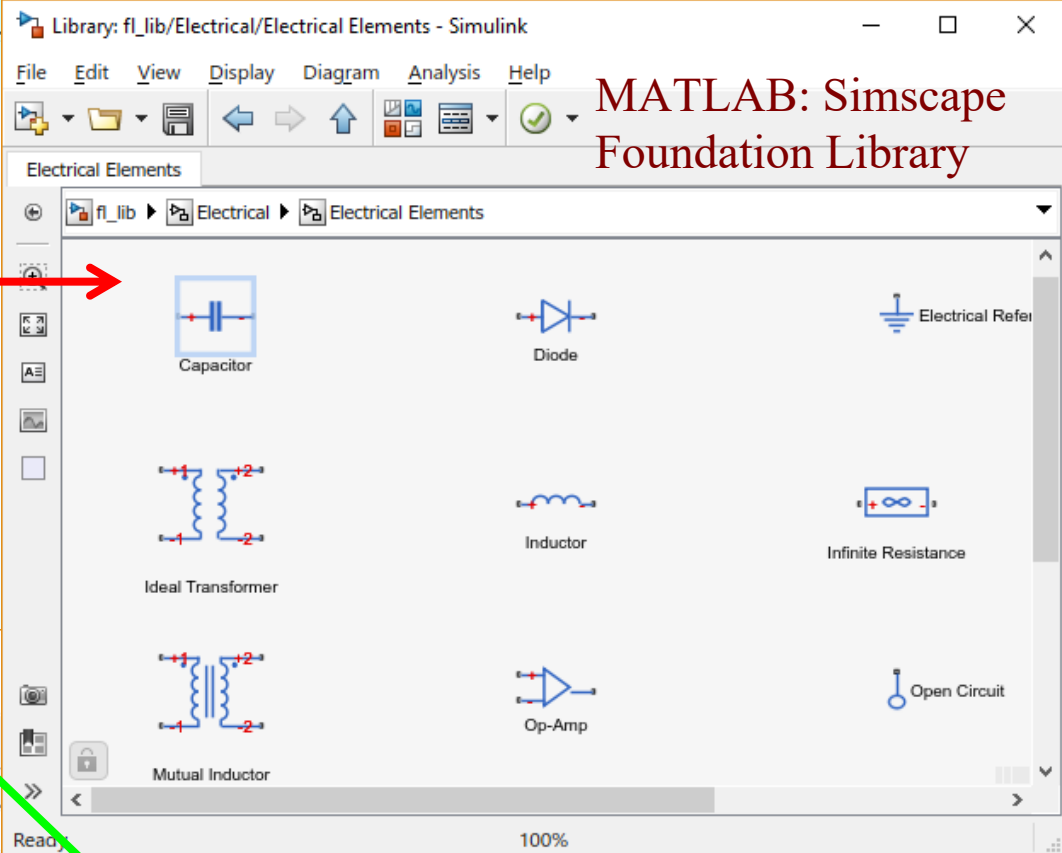
Симулација електричног кола коришћењем **Simscape > Foundation Library** библиотеке



**MATLAB > Simscape >
Foundation Library, Utilities**

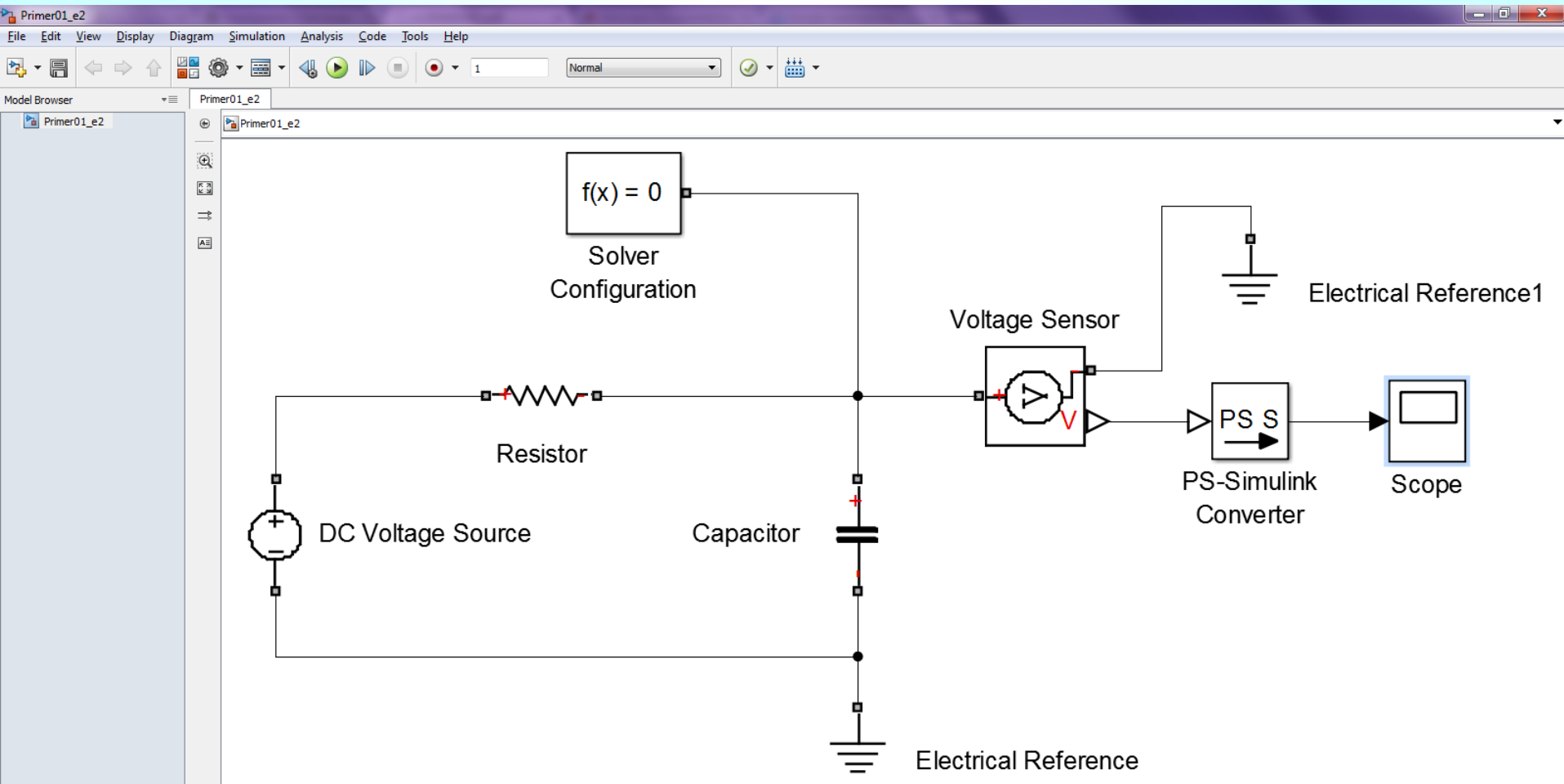


**MATLAB > Simscape >
Foundation Library, Utilities**



**MATLAB > Simscape >
Foundation Library, Utilities**

Симулација електричног кола коришћењем **Simscape > Foundation Library** библиотеке



**MATLAB > Simscape >
Foundation Library, Utilities**

Block Parameters: Solver Configuration

Solver Configuration
Defines solver settings to use for simulation.

Parameters

☐ Start simulation from steady state

Consistency tolerance

☐ Use local solver

Solver type

Sample time

☐ Use fixed-cost runtime consistency iterations

Nonlinear iterations

Mode iterations

Linear Algebra

Delay memory budget [kB]

☒ Apply filtering at 1-D/3-D connections when needed

Filtering time constant

Block Parameters: DC Voltage Source

DC Voltage Source

The ideal voltage source maintains a constant voltage across its output terminals, independent of the current flowing through the source. The output voltage is defined by the Constant voltage parameter, and can be any real value.

[Source code](#)

Settings

Parameters

Constant voltage:

Block Parameters: Resistor

Resistor

The voltage-current (V-I) relationship for a linear resistor is $V=I \cdot R$, where R is the constant resistance in ohms.

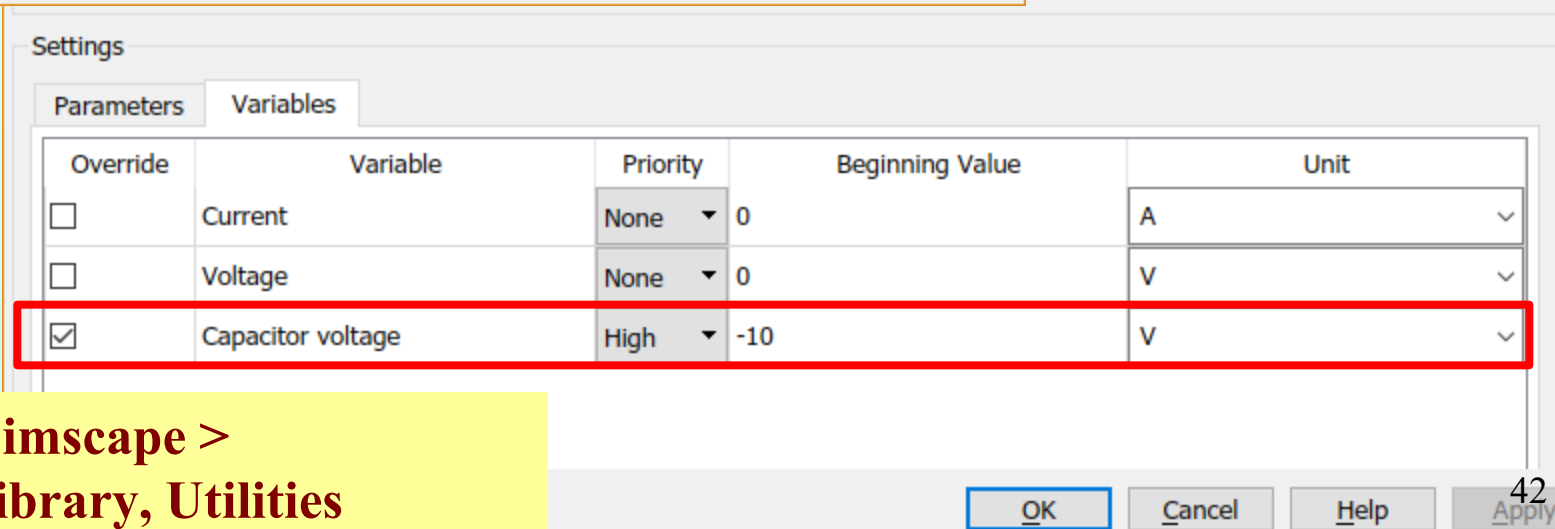
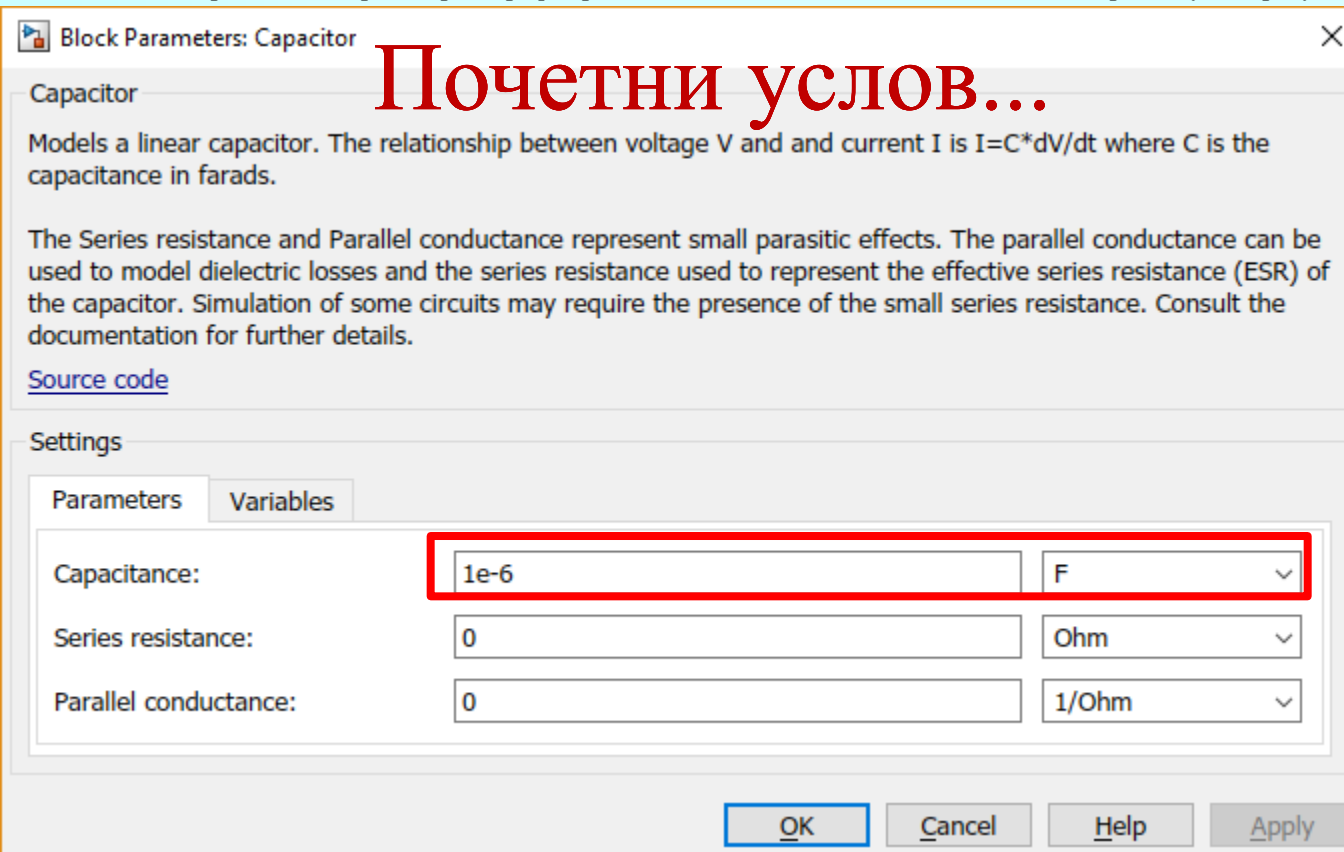
The positive and negative terminals of the resistor are denoted by the + and - signs respectively. By convention, the voltage across the resistor is given by $V(+)-V(-)$, and the sign of the current is positive when flowing through the device from the positive to the negative terminal. This convention ensures that the power absorbed by a resistor is always positive.

[Source code](#)

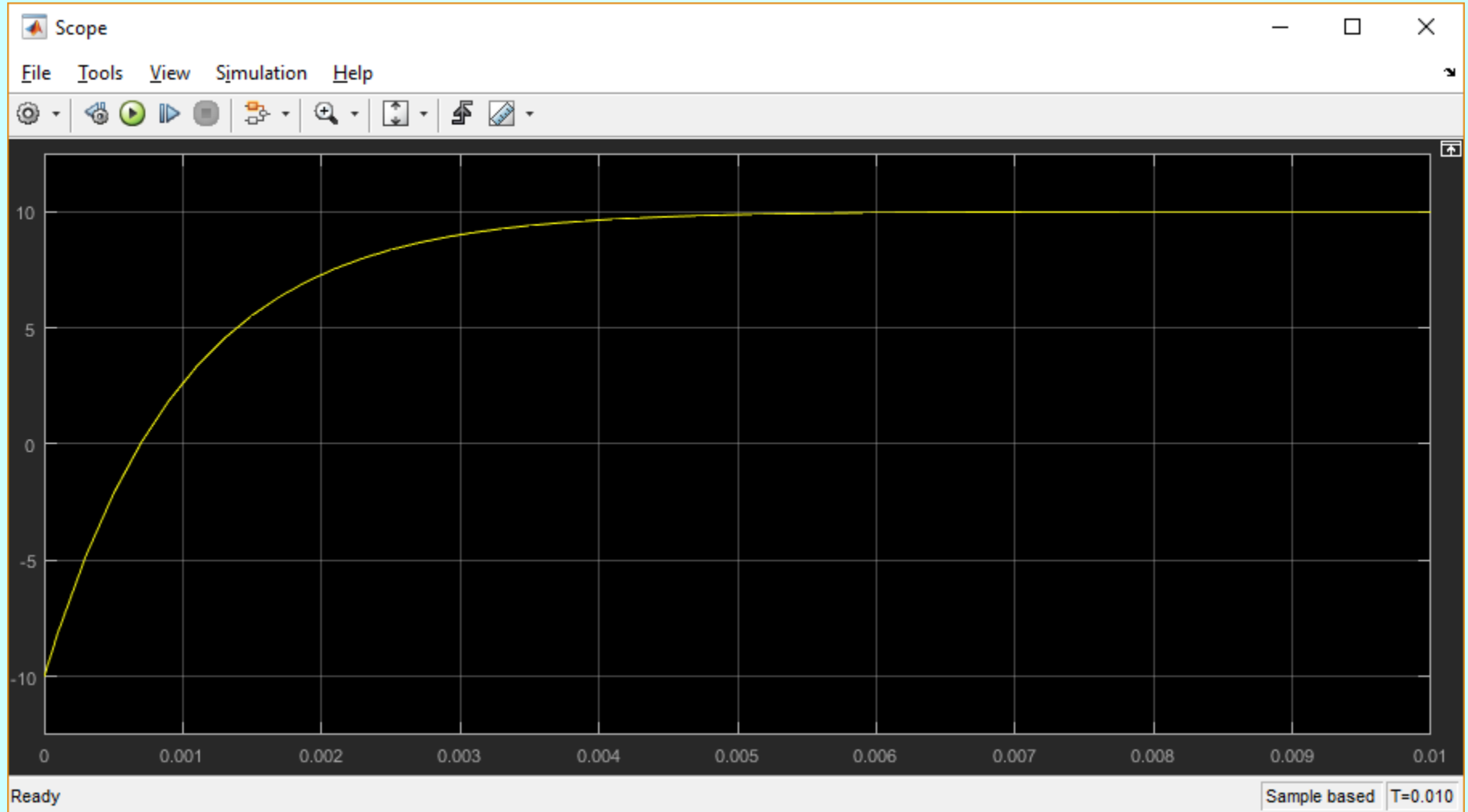
Settings

Parameters Variables

Resistance:



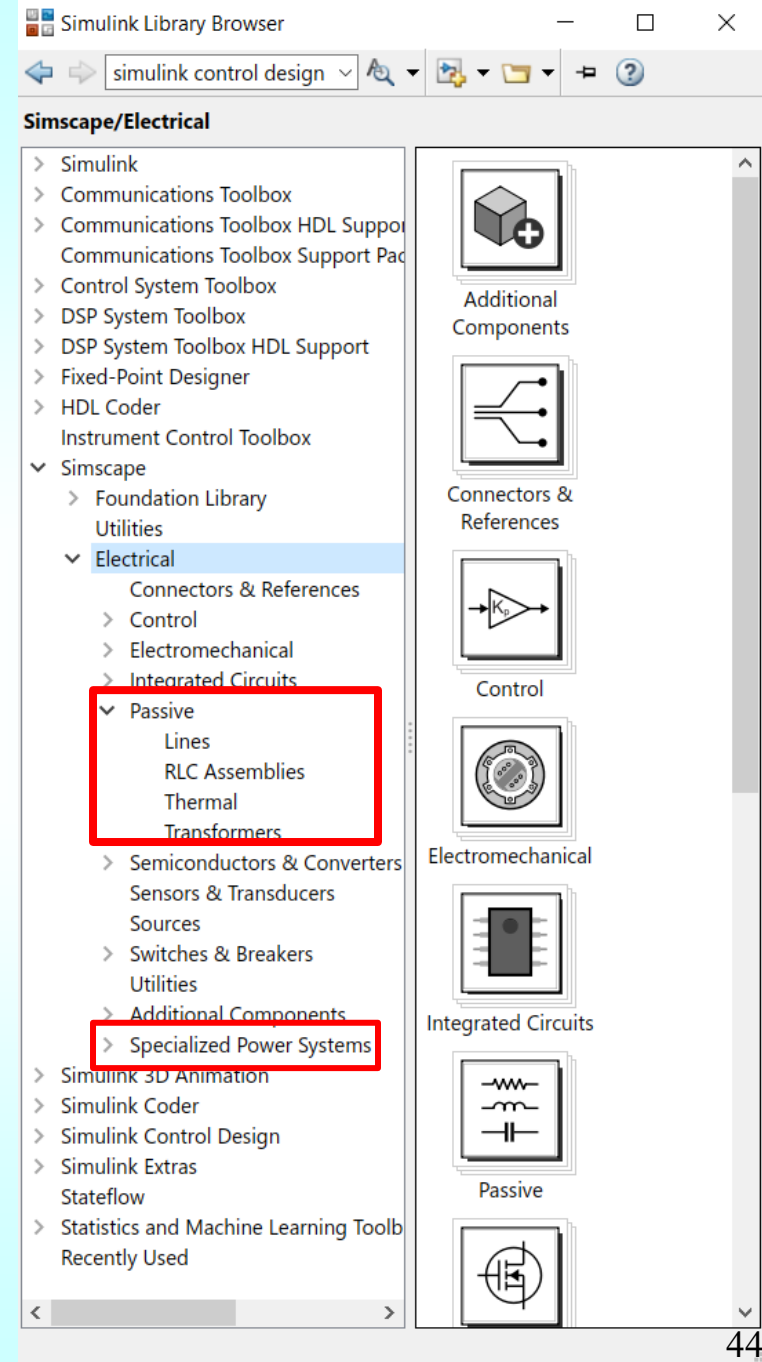
Одређивање одзива...



**MATLAB > Simscape >
Foundation Library, Utilities**

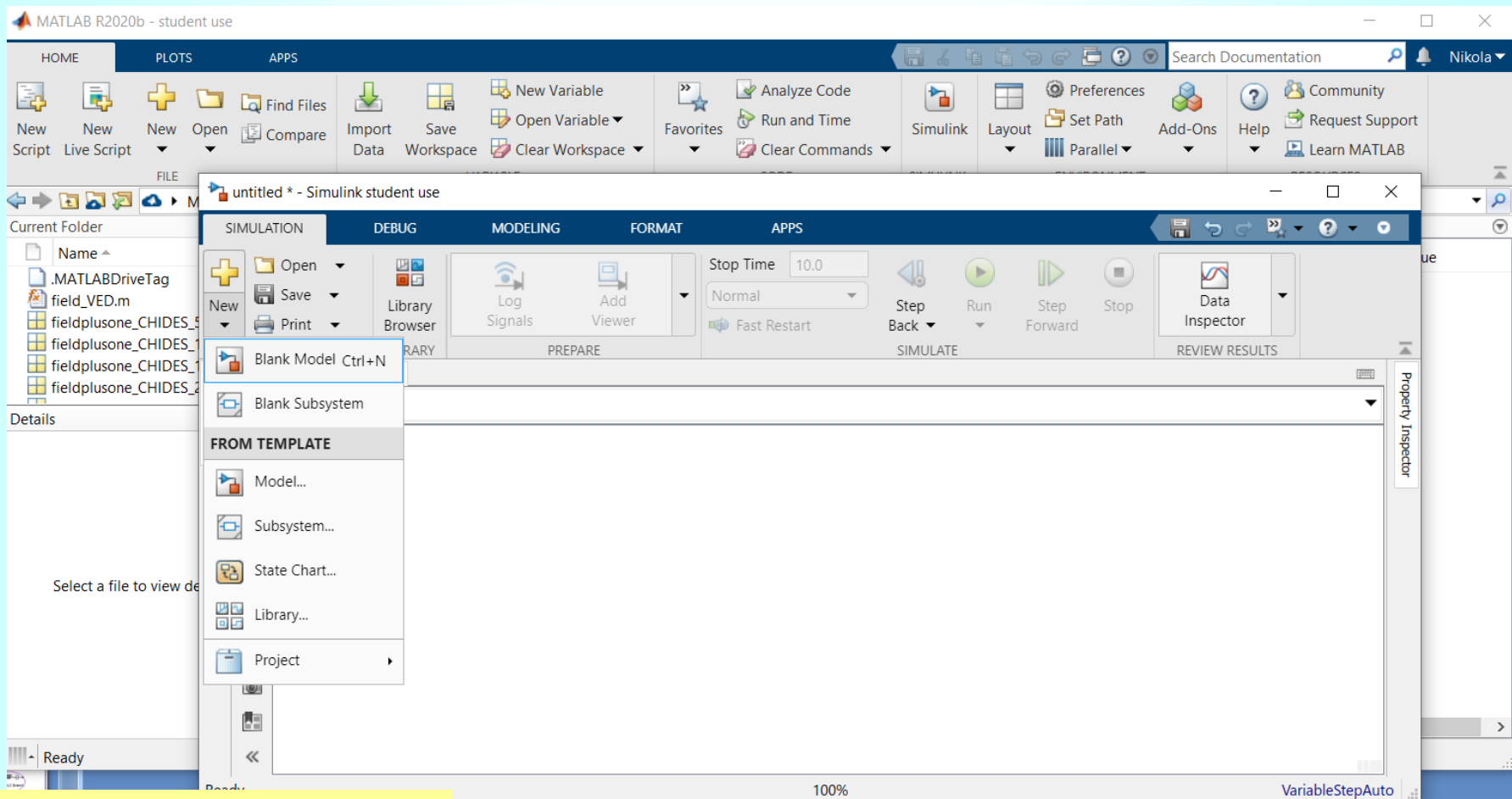
Simulink > Simscape > Electrical

- Библиотека за симулацију и нумеричку анализу електроенергетских система
- Трофазни извори, водови, трансформатори, потрошачи
- Термална анализа

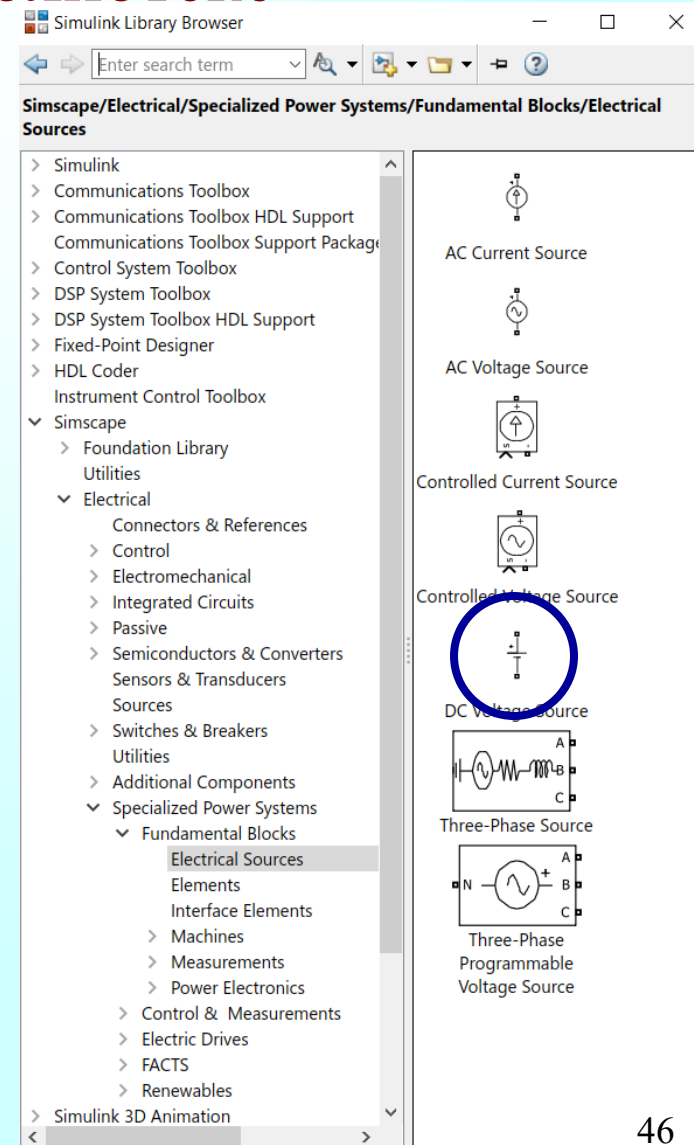
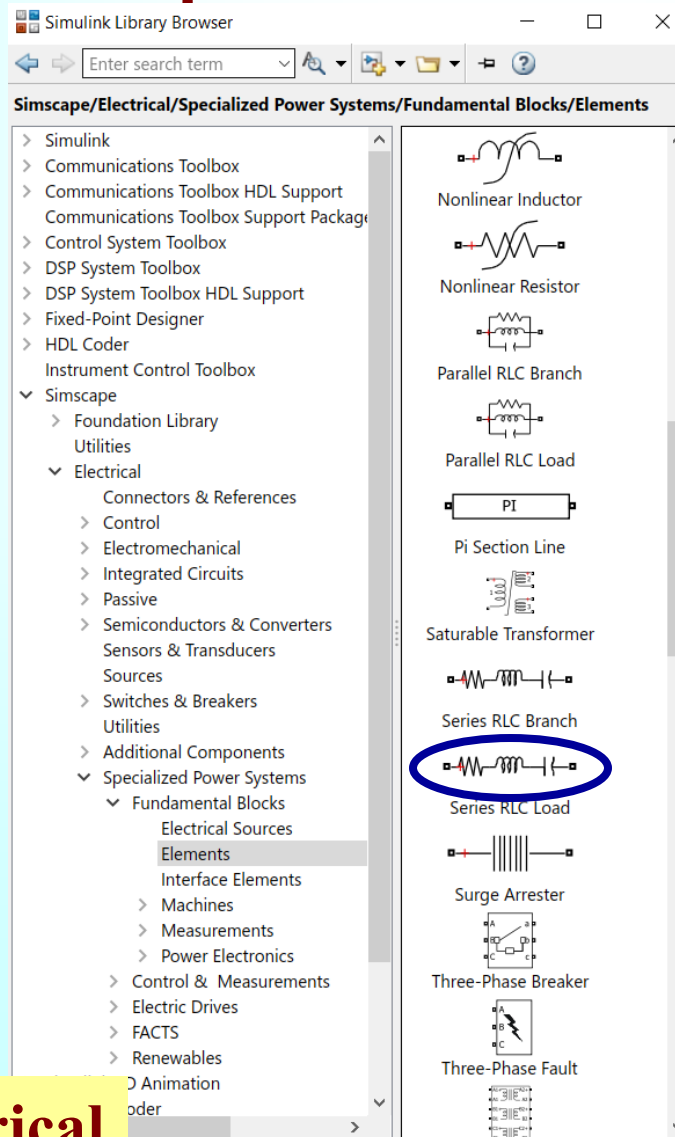


Simscape > Electrical

Симулација електричног кола коришћењем **Simscape>Electrical** библиотеке

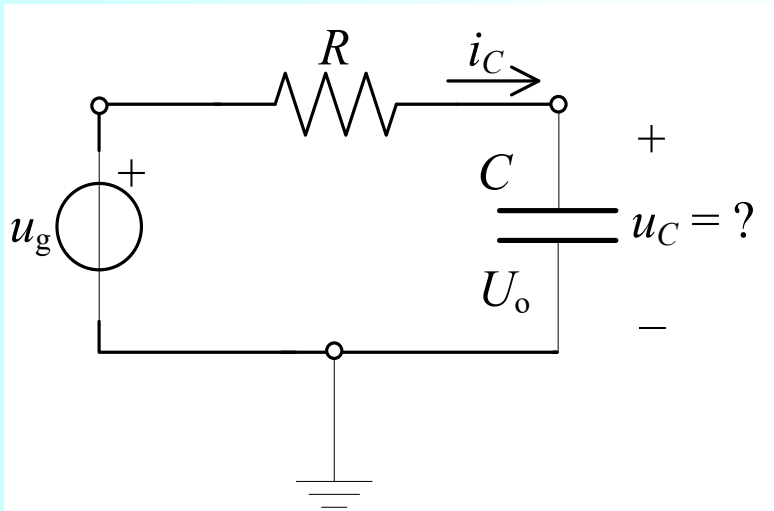


Симулација електричног кола коришћењем **Simscape>Electrical** библиотеке



Симулација електричног кола коришћењем **Simscape>Electrical** библиотеке

коло је образовано у тренутку $t_0 = 0$

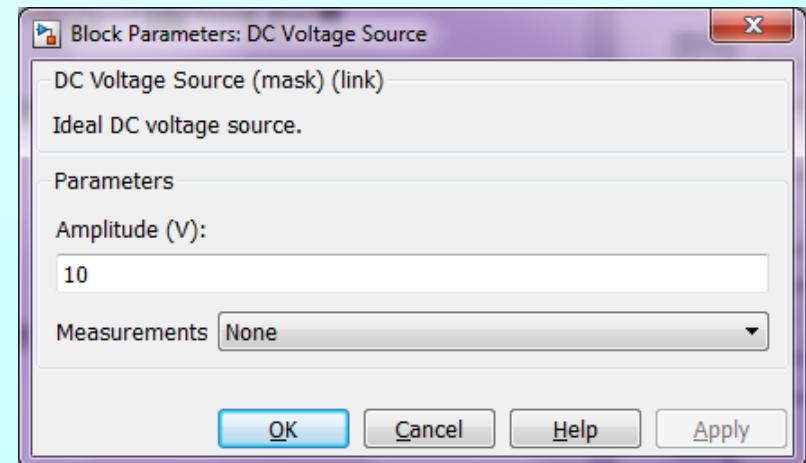
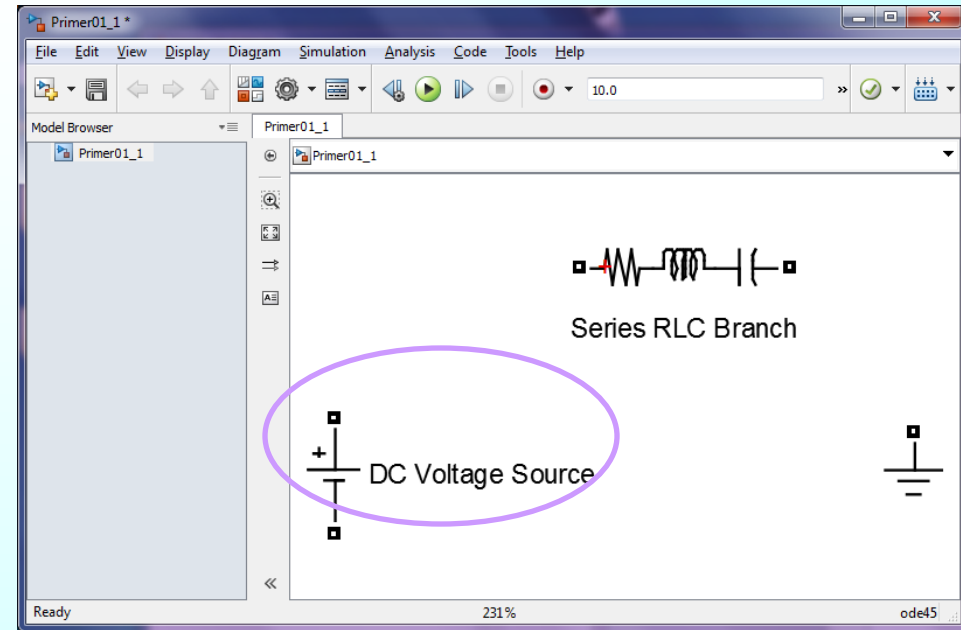


$$R = 1\text{ k}\Omega$$

$$C = 1\text{ }\mu\text{F}$$

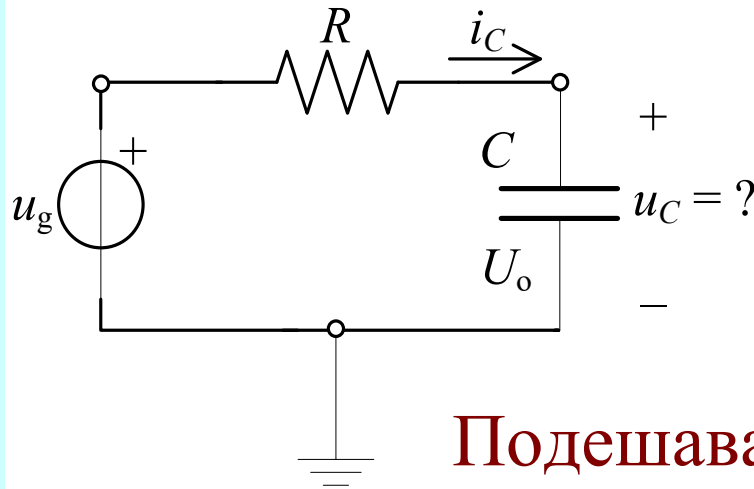
$$u_g(t) = U h(t), U = 10\text{ V}$$

$$u(t_0^-) = U_0 = -10\text{ V}$$



Симулација електричног кола коришћењем **Simscape>Electrical** библиотеке

коло је образовано у тренутку $t_0 = 0$



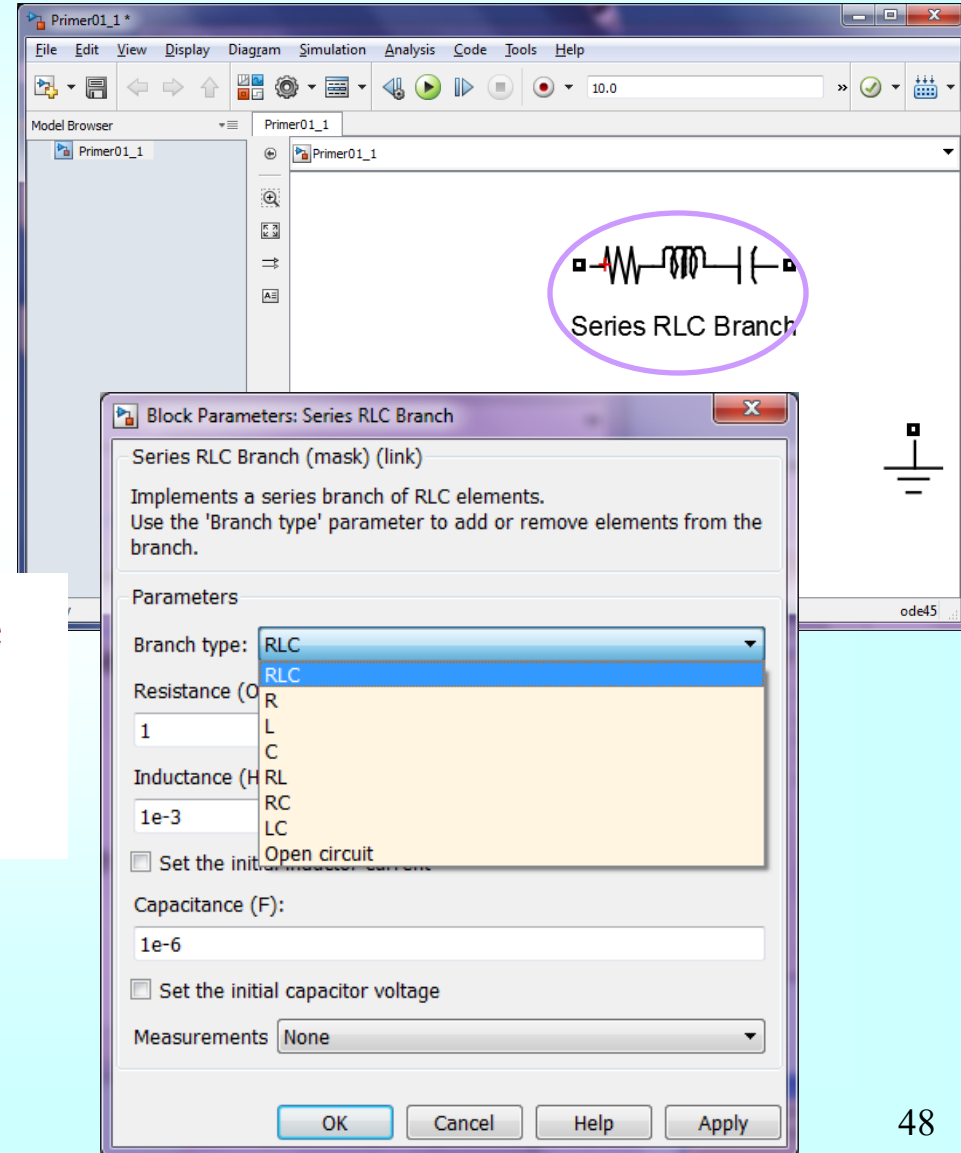
$$R = 1 \text{ k}\Omega$$

$$C = 1 \mu\text{F}$$

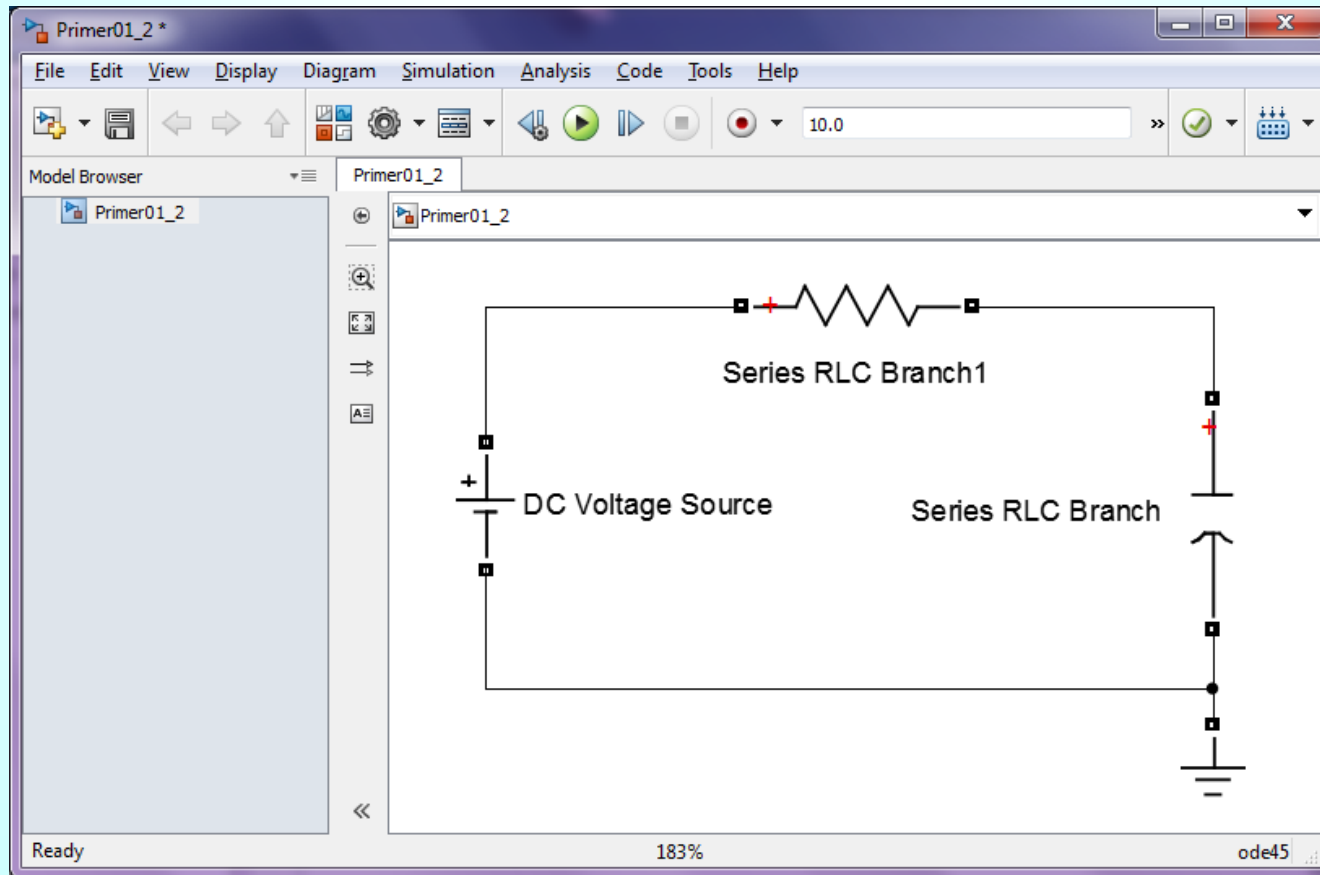
$$u_g(t) = U h(t), U = 10 \text{ V}$$

$$u(t_0^-) = U_0 = -10 \text{ V}$$

Подешавање
параметара
симулације

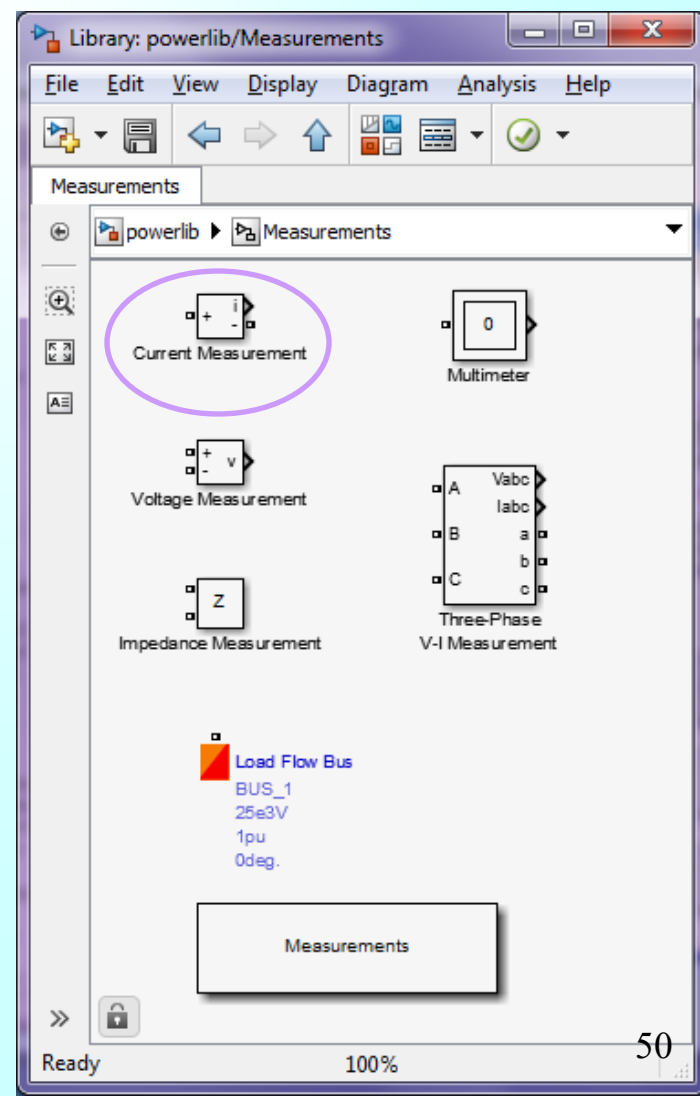
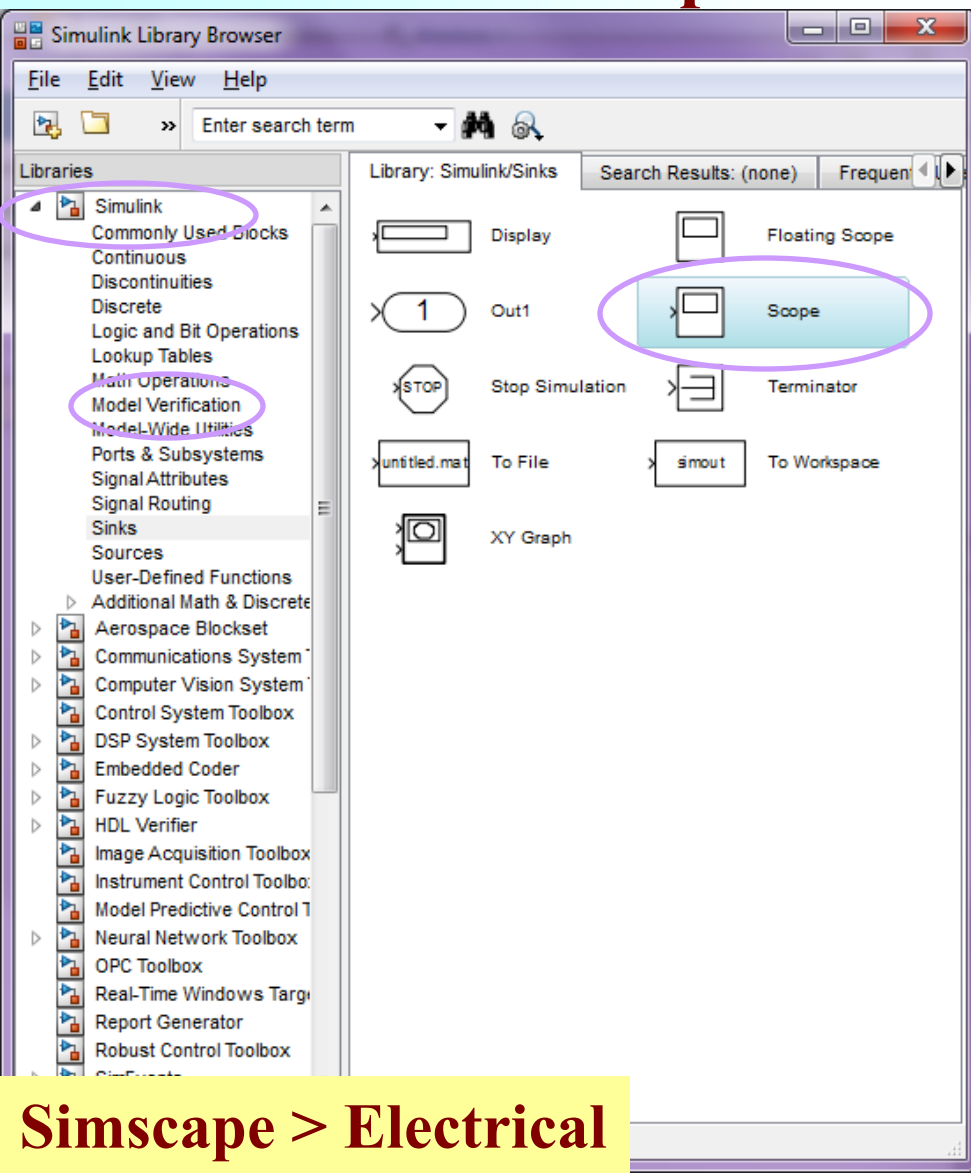


Симулација електричног кола коришћењем **Simscape>Electrical** библиотеке



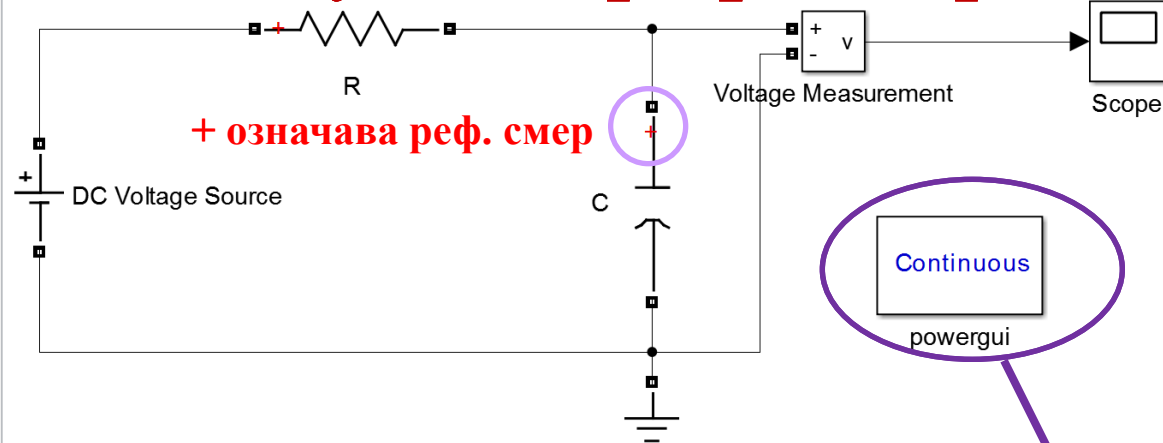
Како се мери напон?

Симулација електричног кола коришћењем **Simscape>Electrical** библиотеке



Почетни услов, реф. смер...

+ означава реф. смер



Block Parameters: C

Series RLC Branch (mask) (link)

Implements a series branch of RLC elements. Use the 'Branch type' parameter to add or remove elements from the branch.

Parameters

Branch type: C

Capacitance (F): 1e-6

☒ Set the initial capacitor voltage

Capacitor initial voltage (V): -10

Measurements: None

OK **Cancel** **Help**

Плоче кондензатора се разликују!!!

Simscape > Electrical

Block Parameters: powergui

PSB option menu block (mask)

Set simulation type, simulation parameters, and preferences.

Solver **Tools** **Preferences**

Steady-State **Initial State** Machine Initialization Impedance Measurement

FFT Analysis Use Linear System Analyzer

Hysteresis Design RLC Line Parameters

Generate Report Customize SPS blocks

Load Flow

► Load flow settings

OK **Cancel** **Help**

Powergui Initial States Setting Tool. model: untitled

Initial electrical state values for simulation:

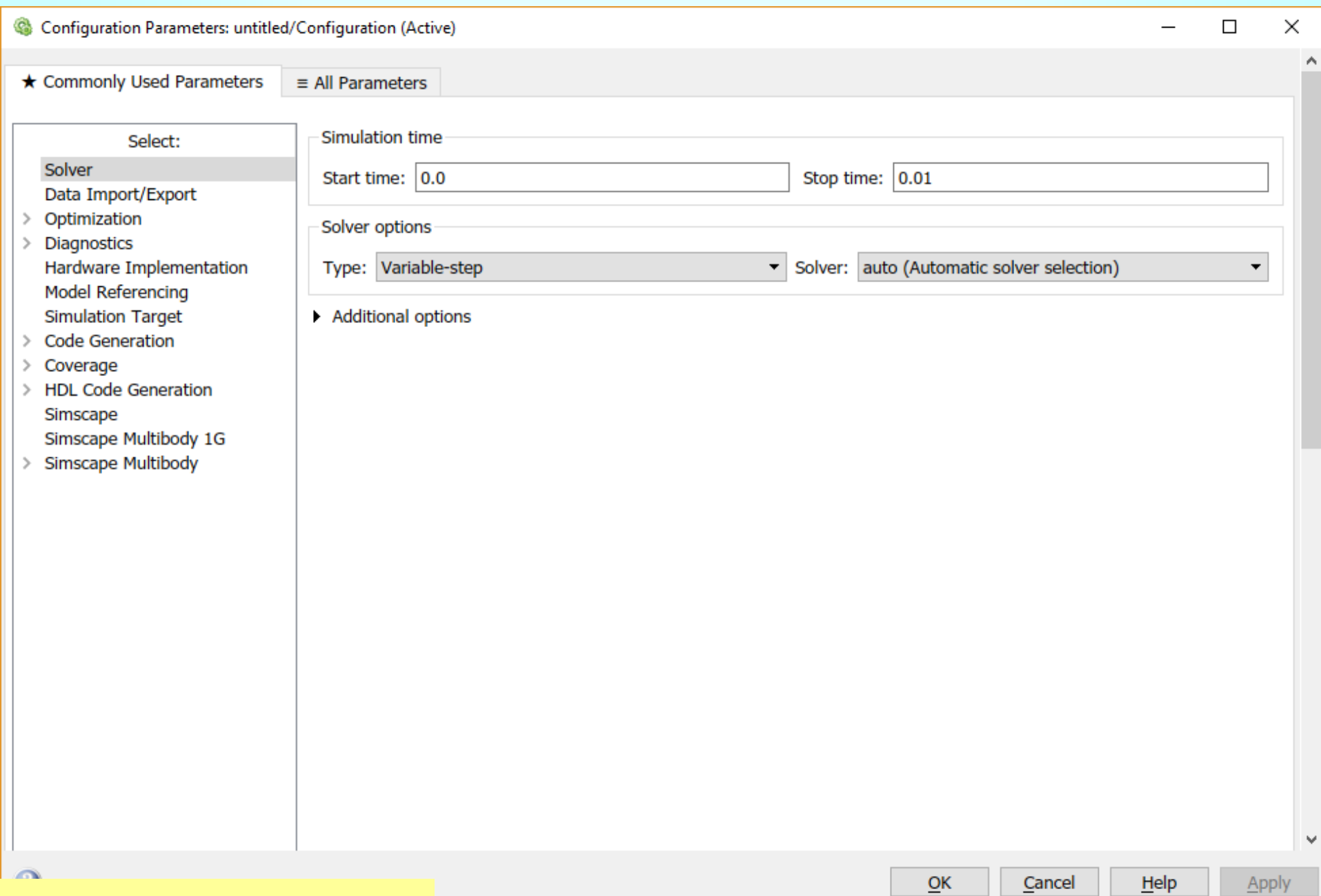
1	'Uc' C =	-10 V

Set selected electrical state values:

-10

Force initial electrical state:

☐ To Steady State ☐ To Zero



Configuration Parameters: untitled/Configuration (Active)

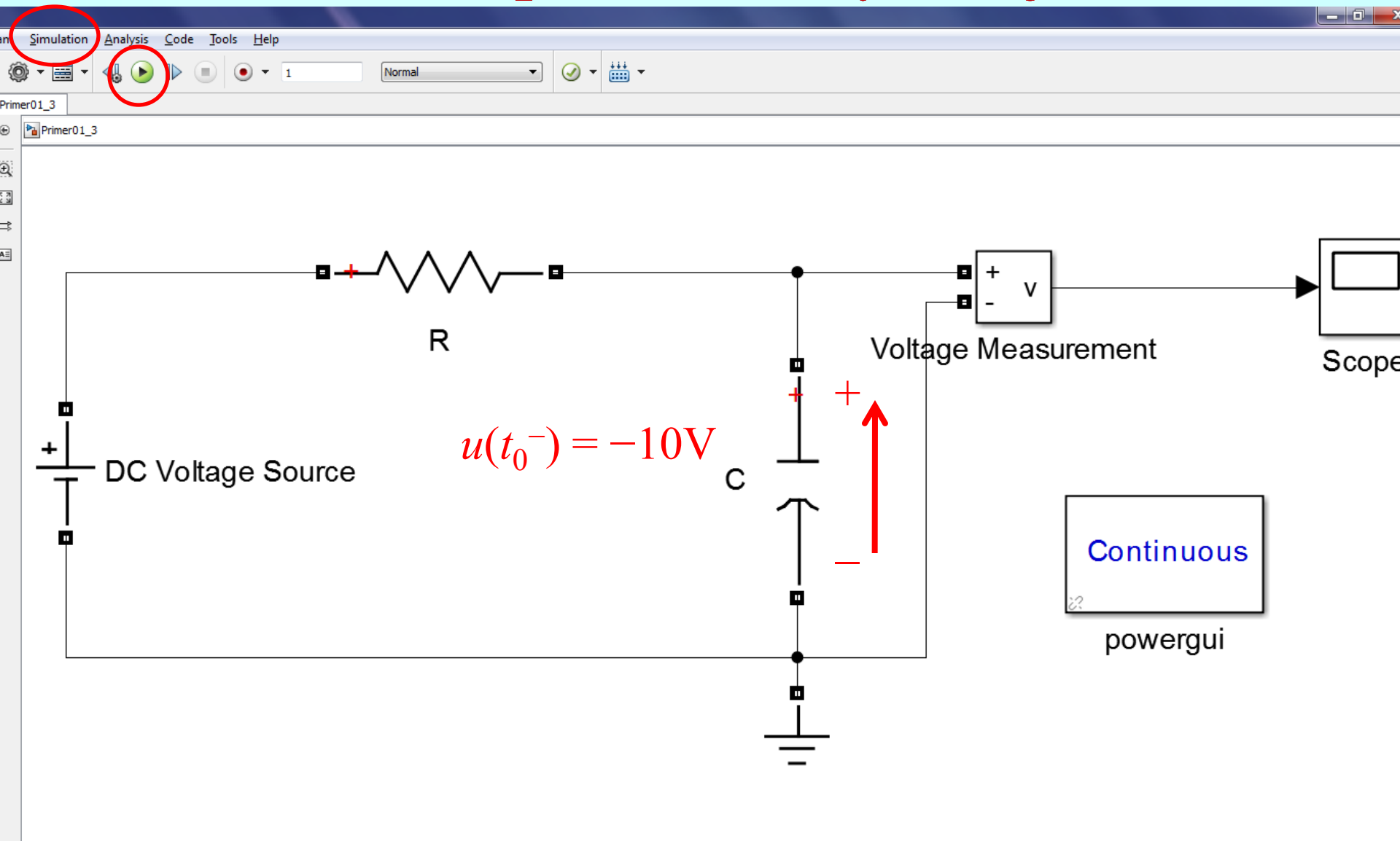
★ Commonly Used Parameters ≡ All Parameters

Category:

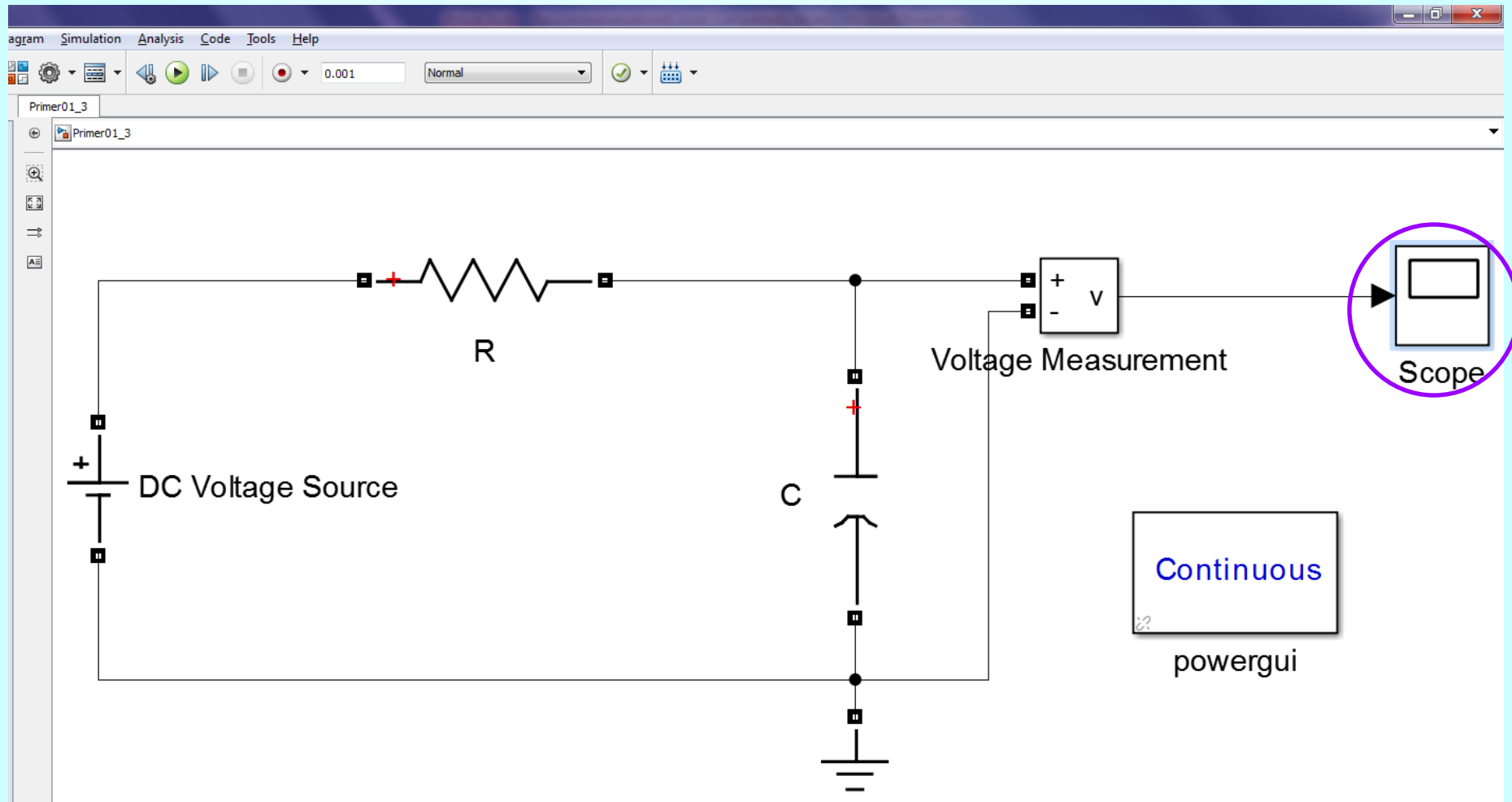
Parameter	Value	Command-Line Name
Solver ▶ Simulation time		
Start time Simulation start time. Note that the values that you specify as block i...	0.0	StartTime
Stop time Simulation stop time.	0.01	StopTime
Solver ▶ Solver options		
Type Choose a variable or fixed-step solver.	Variable-step	SolverType
Solver Choose a solver. If disabled, enable in "Additional options" by changi...	auto (Automatic solver selection)	Solver
Solver ▶ Additional options		
Max step size Maximum step size for a variable-step solver.	1e-4	MaxStep
Relative tolerance Specify the largest acceptable solver error, relative to the size of eac...	1e-3	RelTol
Min step size Minimum step size for a variable-step solver.	auto	MinStep
Absolute tolerance Specify the largest acceptable solver error, as the value of the meas...	auto	AbsTol
Initial step size Specify the size of the first time step that the solver takes.	auto	InitialStep
Shape preservation Improve the integration accuracy by preserving the shape of states b...	Disable All	ShapePreserveControl

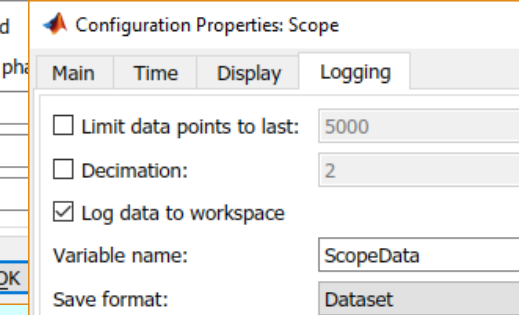
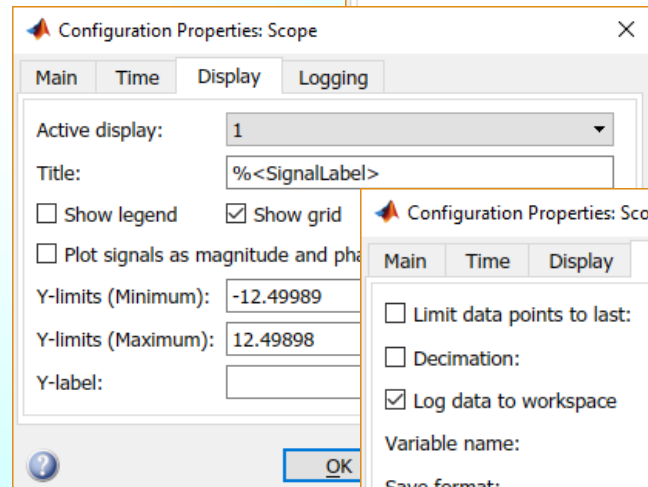
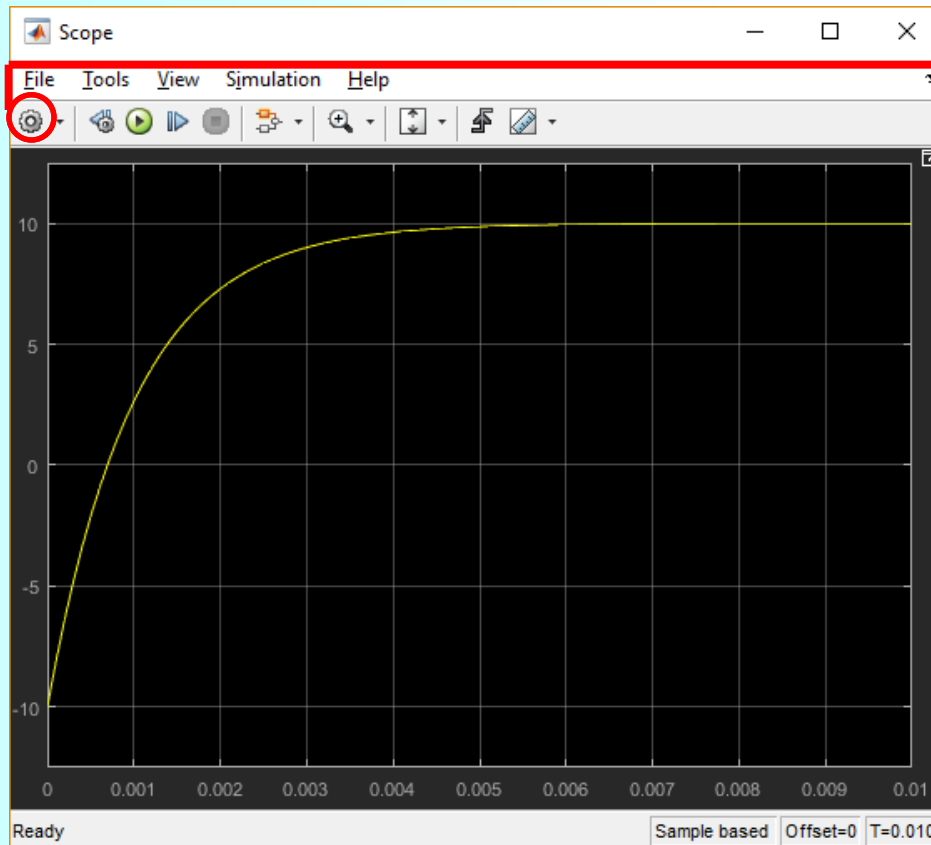
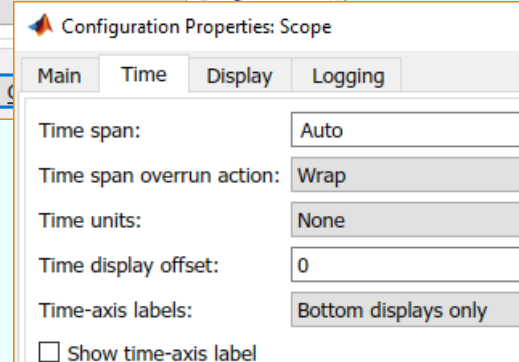
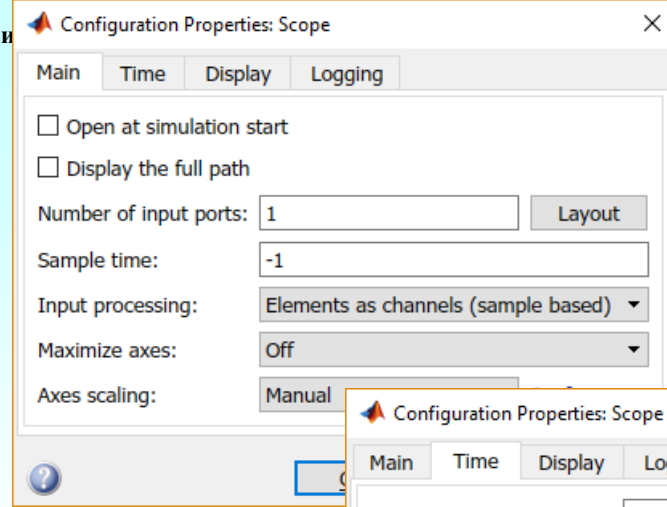
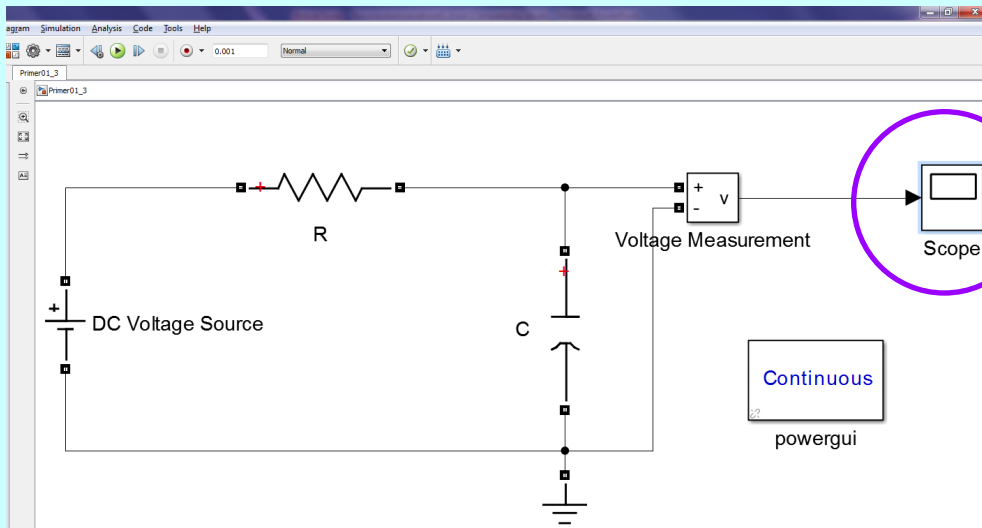
OK Cancel Help Apply

Покретање симулације...



Налажење одзива... Осцилоскоп





Simscape > Electrical

Configuration Parameters: untitled/Configuration (Active)

★ Commonly Used Parameters ≡ All Parameters

Category: All

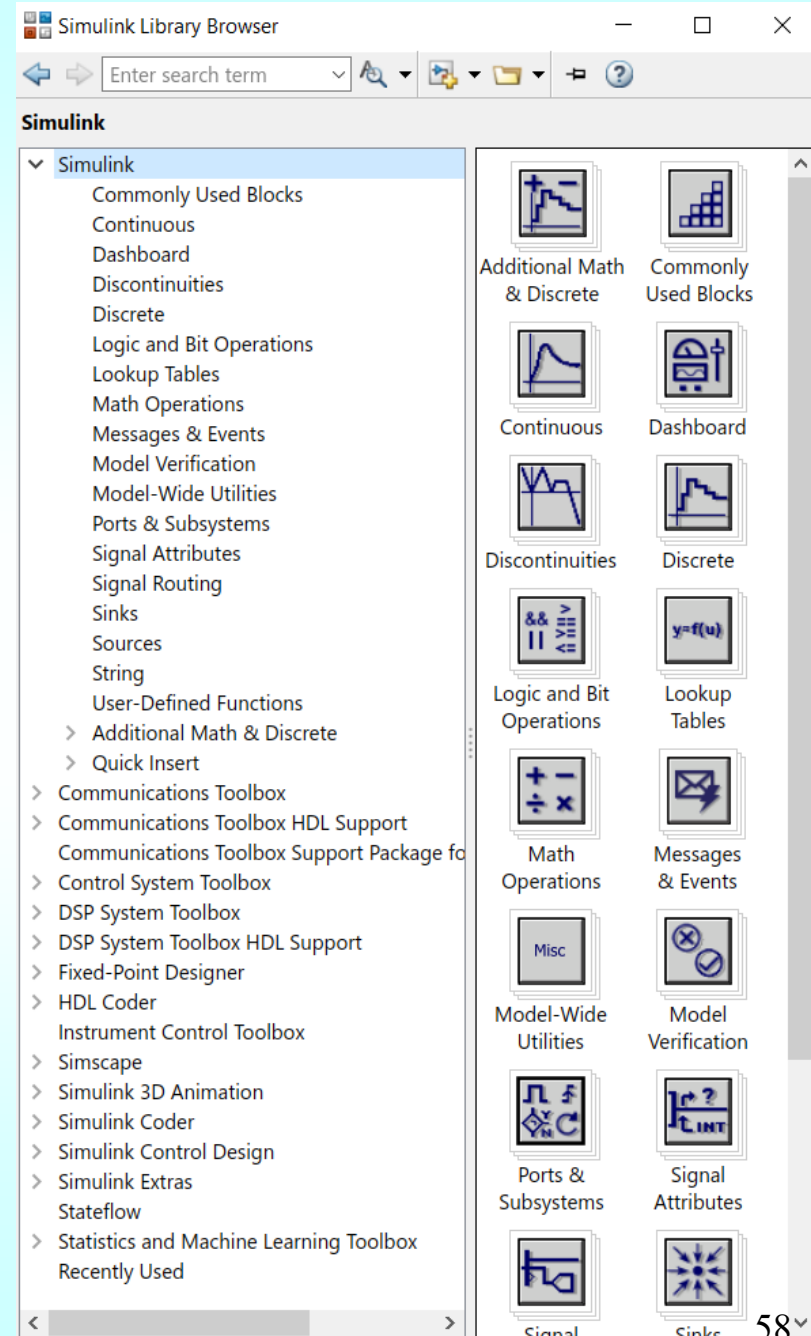
Parameter	Value	Command-Line Name
Solver ▶ Simulation time		
Start time Simulation start time. Note that the values that you specify as block i...	0.0	StartTime
Stop time Simulation stop time.	0.01	StopTime
Solver ▶ Solver options		
Type Choose a variable or fixed-step solver.	Variable-step	SolverType
Solver Choose a solver. If disabled, enable in "Additional options" by changi...	auto (Automatic solver selection)	Solver
Solver ▶ Additional options		
Max step size Maximum step size for a variable-step solver.	1e-4	MaxStep
Relative tolerance Specify the largest acceptable solver error, relative to the size of eac...	1e-3	RelTol
Min step size Minimum step size for a variable-step solver.	auto	MinStep
Absolute tolerance Specify the largest acceptable solver error, as the value of the meas...	auto	AbsTol
Initial step size Specify the size of the first time step that the solver takes.	auto	InitialStep
Shape preservation Improve the integration accuracy by preserving the shape of states b...	Disable All	ShapePreserveControl

$\tau = RC = 1 \text{ ms}$
 $\tau/10 = 0.1 \text{ ms}$

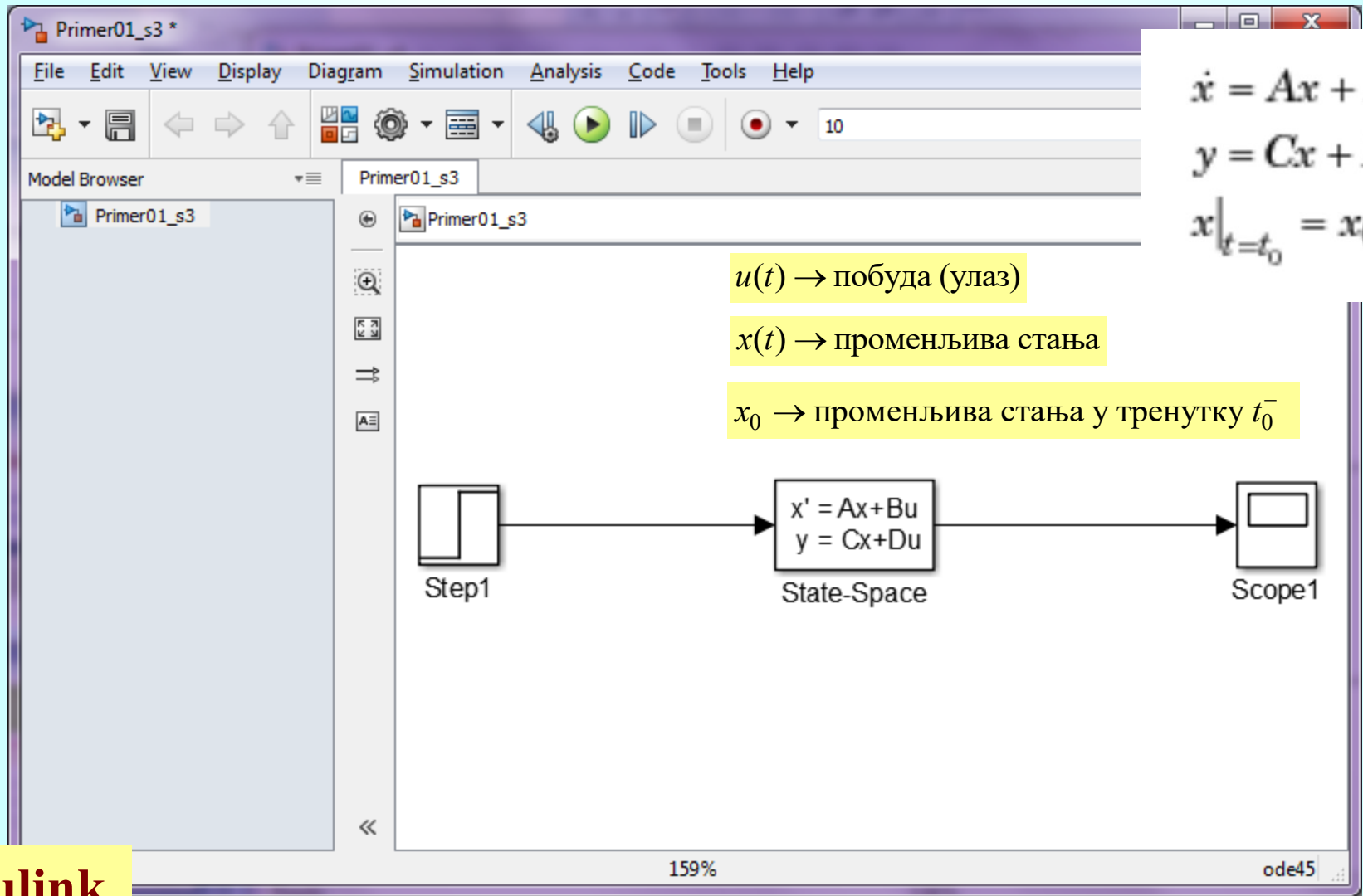
OK Cancel Help Apply

Simulink

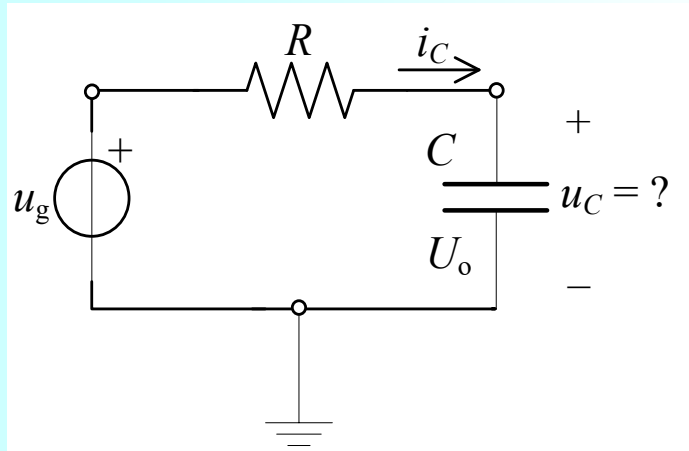
- Решавање кола може се извршити његовим свођењем на систем линеарних диференцијалних једначина
- Дати систем може се представити једноставним блоком са више улаза и излаза



Симулација електричног кола као уопштеног система коришћењем Simulink-а



Симулација електричног кола као уопштеног система коришћењем Simulink-а



$u = u_g(t) \rightarrow$ побуда (улаз)

$x = u_C(t) \rightarrow$ напон кондензатора (променљива стања и одзив - излаз)

$x(t_0^-) = U_0 \rightarrow$ напон кондензатора у тренутку t_0^-

$$u_g(t) = Ri_C + u_C(t)$$

$$i_C = C \frac{du_C}{dt}$$

$$u_C(t_0^-) = U_0 = -10 \text{ V}$$

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \\ x|_{t=t_0} &= x_0, \end{aligned}$$

коло је образовано у тренутку $t_0 = 0$

$$R = 1 \text{ k}\Omega$$

$$C = 1 \mu\text{F}$$

$$u_g(t) = U h(t), U = 10 \text{ V}$$

$$u_C(t_0^-) = U_0 = -10 \text{ V}$$

$$\begin{aligned} \frac{du_C(t)}{dt} &= \frac{i_C}{C} = \frac{u_g(t) - u_C(t)}{RC} = -\frac{u_C(t)}{RC} + \frac{u_g(t)}{RC} \\ u_C(t_0^-) &= U_0 = -10 \text{ V} \end{aligned}$$

$$\frac{du_C(t)}{dt} = Au_C(t) + Bu_g(t) \Rightarrow A = -\frac{1}{RC} = -1000, B = 1000$$

$$u_C(t) = Cu_C(t) \Rightarrow C = 1, D = 0$$

$$u_C(t_0^-) = U_0 = -10 \text{ V}$$

Баста, доцент, 19Е032ПРТК, Универзитет

Симулација електричног кола као уопштеног система коришћењем Simulink-а

Source Block Parameters: Step1

Step

Output a step.

Parameters

Step time:

Initial value:

Final value:

Sample time:

☒ Interpret vector parameters as 1-D

☒ Enable zero-crossing detection

OK Cancel Help Apply

Function Block Parameters: State-Space

State Space

State-space model:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned}$$

Parameters

A:

B:

C:

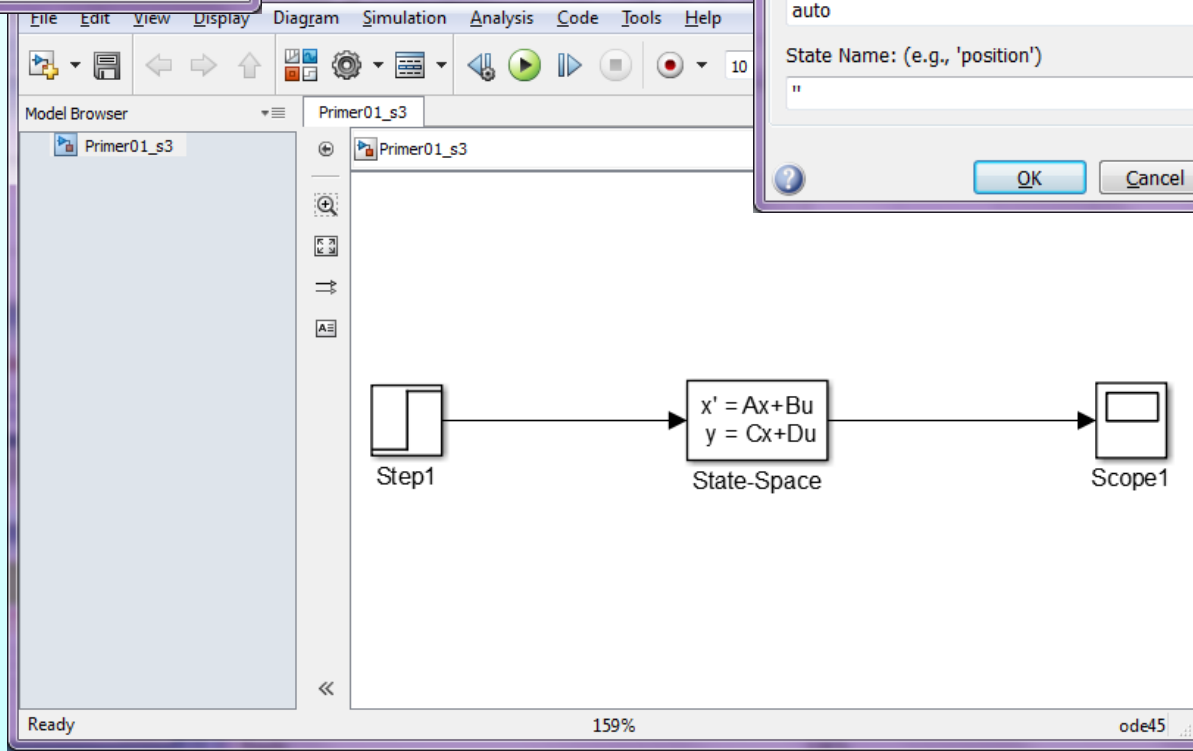
D:

Initial conditions:

Absolute tolerance:

State Name: (e.g., 'position')

OK Cancel Help Apply



Function Block Parameters: State-Space

State Space

State-space model:
 $\dot{x}/dt = Ax + Bu$
 $y = Cx + Du$

Parameters

A:

B:

C:

D:

Initial conditions:

Absolute tolerance:

State Name: (e.g., 'position')

OK Cancel

Баста, доцент, 19Е032ПРТК, Универзитет у И

Симулација електричног кола као уопштеног система коришћењем Simulink-а

Source Block Parameters: Step1

Step

Output a step.

Parameters

Step time:

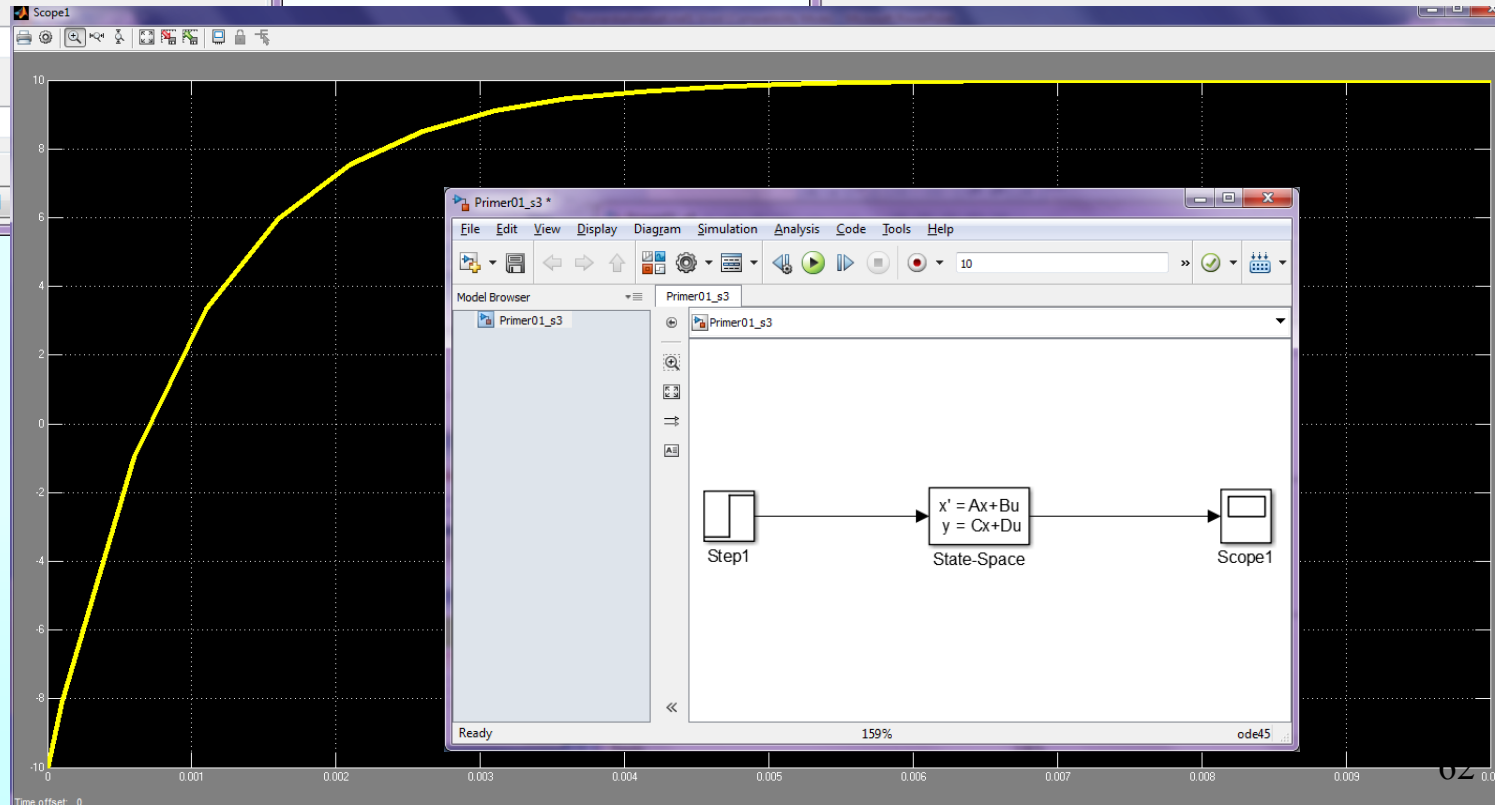
Initial value:

Final value:

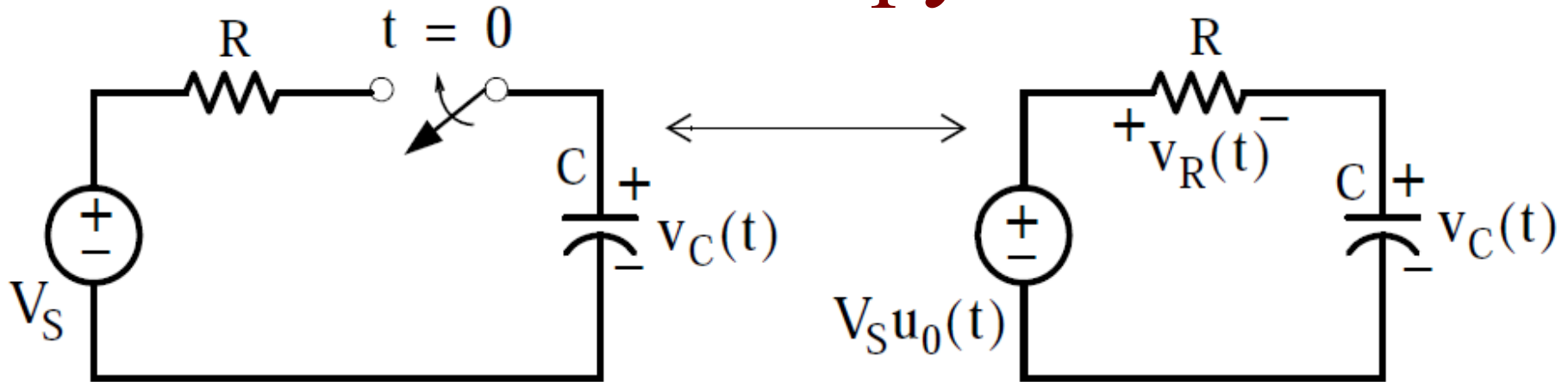
Sample time:

☒ Interpret vector parameters as 1-D

☒ Enable zero-crossing detection



Решимо “ручно”



$$v_R + v_C = V_S u_0(t)$$

$$i = i_C = C \frac{dv_C}{dt}$$

$$v_R = Ri = RC \frac{dv_C}{dt}$$

$$RC \frac{dv_C}{dt} + v_C = V_S u_0(t)$$

$$t > 0$$

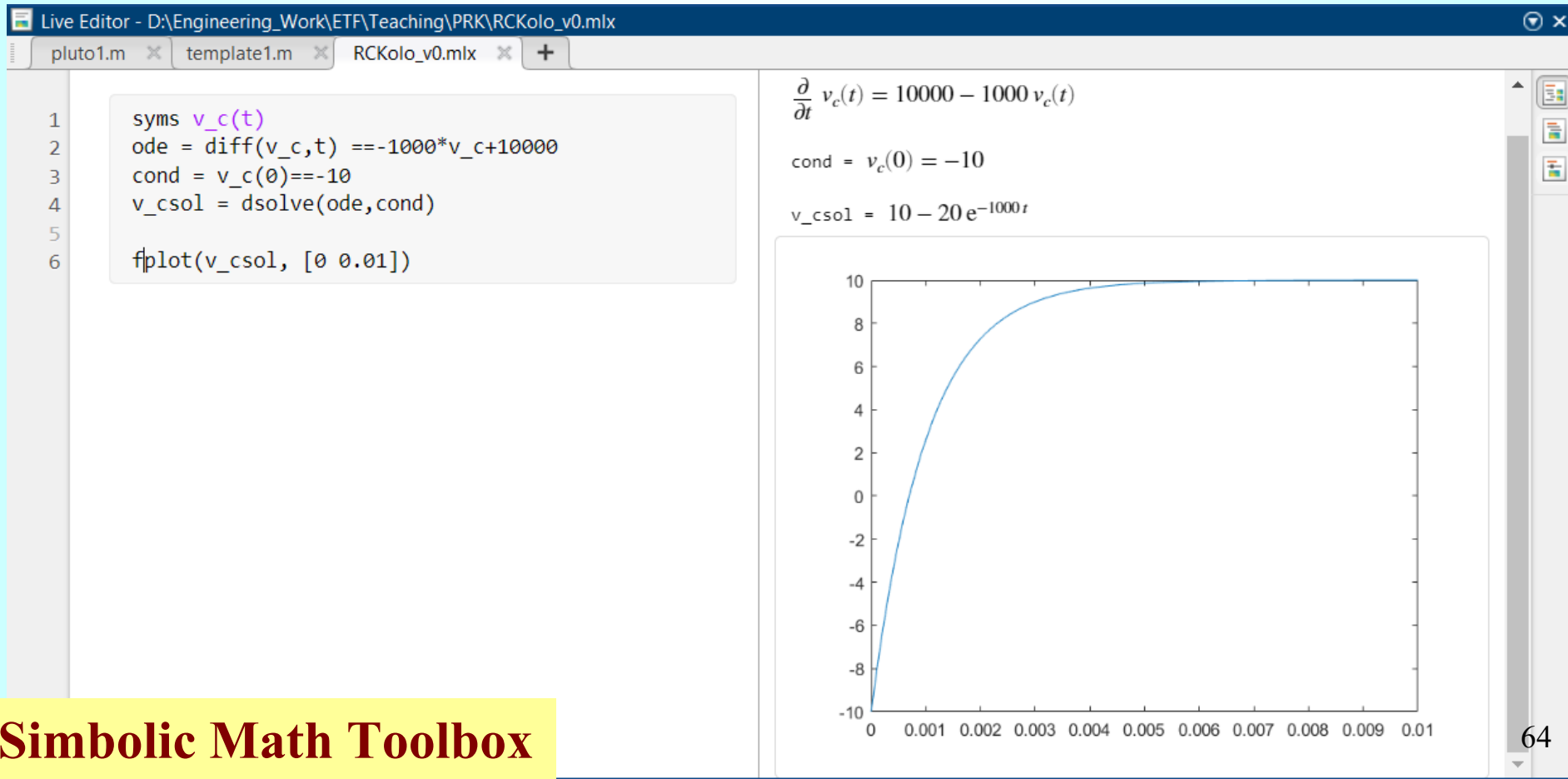
$$RC \frac{dv_C}{dt} + v_C = V_S$$

$$v_C(t) = (V_S - V_S e^{-(1/RC)t}) u_0(t)$$

MATLAB

Symbolic Math Toolbox (Live Script)

- Библиотека за симболички рачун (алгебарске и диференцијалне једначине)
- Омогућава нумеричку анализу резултата прорачуна



wxMaxima

wxMaxima 20.06.6 [unsaved*]

File Edit View Cell Maxima Equations Algebra Calculus Simplify List Plot Numeric Help

Maths

Mathematical Symbols

Plot using Draw

2D	3D
Expression	Implicit Plot
Parametric Plot	Points
Diagram title	Axis
Contour	Plot name
Line color	Fill color
Grid	Accuracy

```
(%i1) jednachine: [ug=R*iC+uC, iC=C*'diff(uC,t)];
(%o1) [ug=uC+R iC, iC=C (d/dt uC)]

(%i2) jednachineDuC: jednachine, 'diff(uC,t)=DuC;
(%o2) [ug=uC+R iC, iC=C DuC]

(%i3) JednachineIzvoda: eliminate(jednachineDuC,
[iC]);
(%o3) [-ug+uC+C DuC R]

(%i4) jednachineStanja: linsolve(JednachineIzvoda, DuC);
(%o4) [DuC=(ug-uC)/(C R)]

(%i5) jednachineDiff: jednachineStanja, DuC='diff(uC, t);
(%o5) [d/dt uC=(ug-uC)/(C R)]
```

```
(%i6) zamene: [ug=U];
(%o6) [ug=U]

(%i7) vrednosti: [R=1000, C=10^(-6), U=10, U0=-10];
(%o7) [R=1000, C= $\frac{1}{1000000}$ , U=10, U0=-10]

(%i8) JednacineDiffZamena: jednacineDiff, zamene;
(%o8) [ $\frac{d}{dt} uC = \frac{U - uC}{C R}$ ]

(%i9) ode2(JednacineDiffZamena, uC, t);

(%o9)  $uC = e^{-\frac{t}{C R}} \left( U e^{\frac{t}{C R}} + c \right)$ 

(%i10) ic1(%, t=0, uC=U0);
(%o10)  $uC = e^{-\frac{t}{C R}} \left( U e^{\frac{t}{C R}} + U0 - U \right)$ 

(%i11) uCt: ev(%, vrednosti);
(%o11)  $uC = e^{-1000 t} \left( 10 e^{1000 t} - 20 \right)$ 

(%i17) wxplot2d(rhs(uCt), [t, 0, 0.01],
[xlabel, " t [s] "],
[ylabel, " uC [V] "],
[xtics, 0, 0.002, 0.01],
[legend, "Napon kondenzatora"], grid2d)$
```

