

POČÍTAČOVÉ APLIKACE



MatLab

zpracoval: Šnoriková, Sedlák

ROZVRŽENÍ UČIVA

- **seznámení s SW**
- **režimy práce** - dialogový, programový, grafický
- **operace a funkce** - skalárů, vektorů a matic, submatice
- **objekty a grafika** – 2 a 3D
- **vytváření funkcí a M-file**

ROZVRŽENÍ UČIVA

- **větvení, rozhodování a cykly** - if, while, for,
- **symbolická matematika**
- **čtení a zápis do souboru**
- **řídící technika**
- **SIMULINK**

a to nejdůležitější
až nyní,
nikdy nezapomeň
na

HELPOVNÍK

to když už nebudeš vědět kudy kam

SEZNÁMENÍ SE SE SW

Seznámení se se SW

popis obrazovky

nastavení velikosti písma **HNMF/preferences/fonts**

Výběr oken **HNMF/desktop**

přesouvání oken

Režim práce - dialogový

práce na příkazové řádce

clc

základní matematické operátory

+, -, *, /, \, ^ a **.** – tečková konvence

speciální proměnné a konstanty

ans, pi(), i nebo **j, NaN -nan - 0/0, Inf - inf,**

další řídicí znaky

. ; : % ... , ! [] () {} ‘

porovnání

== ~= < > ,<= =>

Režim práce - dialogový

práce s proměnnými

who, whos, exist *proměnná*, **clear** *proměnná*,
clear global, **clear global** *proměnná*, **clear all**

práce se soubory

pwd, cd, cd *kam*, **ls** nebo **dir, what, type** *soubor*,
open *soubor*, **save** *soubor*, **delete** *soubor*, **[TAB]**

komplexní čísla

Př.č.1

abs(), angle(), conj(), real(), imag(), isreal()

goniometrické funkce

Př.č.2

sin(), sind(), ..., asin(), ...

Režim práce - dialogový

vektory

Př.č.3

řádkové a sloupcové

matice

Př.č.4

Př.č.5

Př.č.6

ones(a), ones(a,b,...), zeros(), eye(a), eye(a, b, ...),

operace s maticemi

Př.č.7

A/B, A\B, +, -, *, /, ^

maticové funkce

Př.č.8

det(A), inv(A), length(A), size(A)

funkce pro třídění a prohledávání

Př.č.9

min(), max(), mean(), median(), std(),
[hodnota index]=sort()

Režim práce - dialogový

datové struktury

Př.č.10

Kruh.polomer=5.4; ... Kruh

práce s řetězcí – převodní funkce

Př.č.11

abs(r), char(r), int2str(int), num2str(num),
str2num(r), str2double(r), ...

práce s řetězcí – vyhledávací funkce

strmatch(r), strfind(r),

práce s řetězcí –výpis řetězců

char(r1, r2, ...), strvcats(r1, r2, ...),
strcat(r1, r2, ...), sprintf(),

práce s řetězcí – spouštění řetězců

eval(r)

Grafika – 2D

vykreslování grafů

```
x=linspace(od, do, počet kroků),  
plot(x,y,"atributy"), hold on,  
subplot(x,y,z), figure(w),
```

vykreslování grafů

```
x=linspace(od, do, počet kroků),  
plot(x,y,"atributy"), hold on,
```

popisování grafů

```
title('text'), xlabel ('text'), ylabel ('text'), grid,  
axis, legend, text(x,y,'popis'),
```

plot(x,y)

x = [0:2:20];

y = sin(x) ;

plot(x,y); grid on

Grafika – 2D

*$x = \text{ linspace}(0, 2 * \pi, 360);$*

$y = \sin(x);$

$\text{plot}(x, y); \text{grid on}$

Grafika – 2D

```
x=linspace(0, 2*pi(),360);  
y = sin(x) ;  
plot(x,y); hold on;  
y=cos(x); plot(x,y);  
grid on
```

Úprava grafu

**PROVEDEME PŘÍMO VE VLASTNÍM
GRAFU,
TAK STUDENTE NYNÍ UŽ NESPI**

GONIOMETRICKÉ FUNKCE

Příkazy :

```
>> x=0:pi/20:2*pi
```

```
>> y=sin(x);
```

```
>> plot( x ,y , 'r')
```

```
>> hold on
```

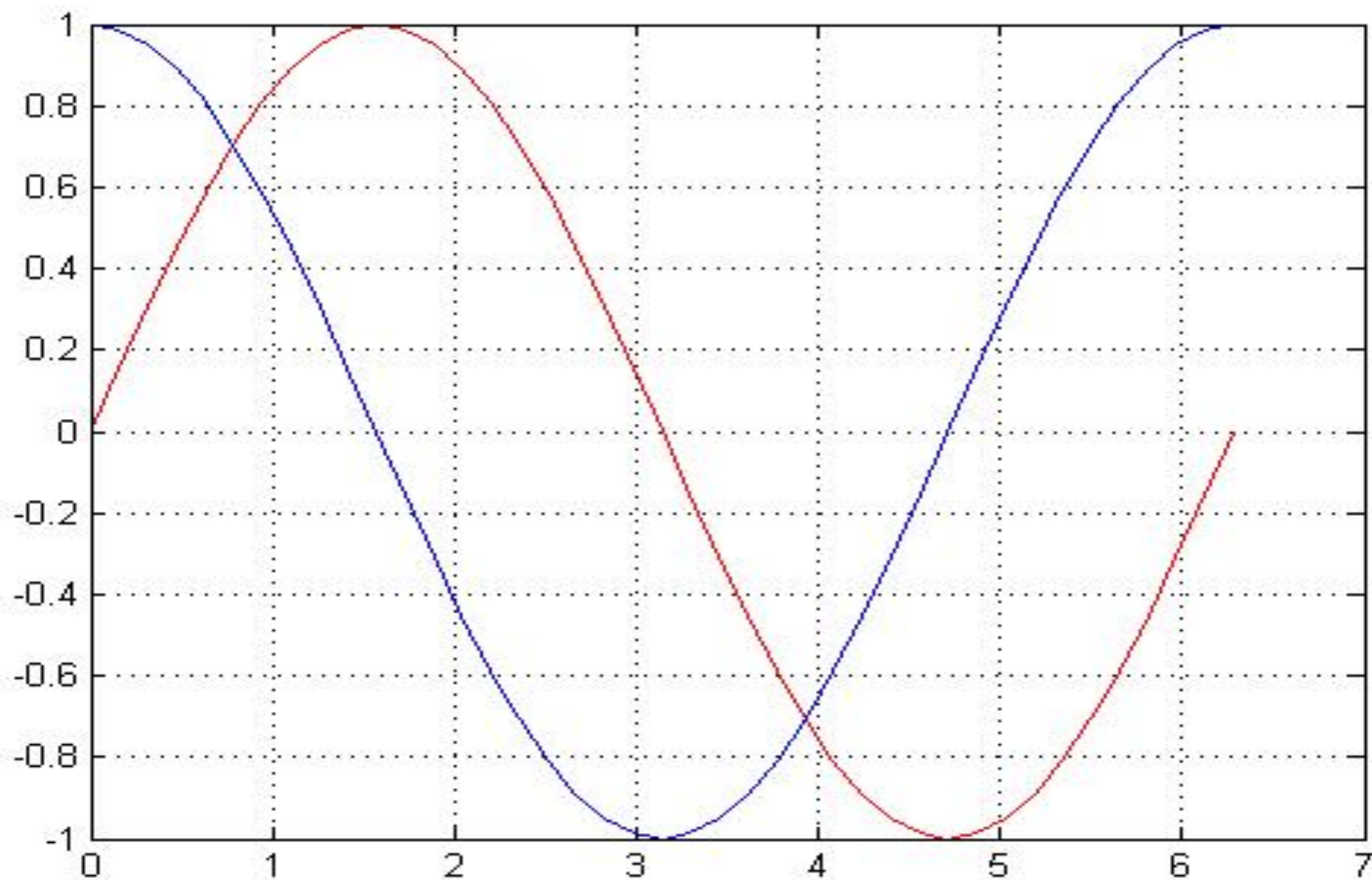
```
>> grid on
```

```
>> z=cos(x )
```

```
>> plot(x ,z , ' b')
```

Grafika – 2D

GRAF



GRAFY 3 FUNKCÍ

Změna argumentu: funkce $\sin(x)$, $\sin(2x)$, $\sin(1/2x)$

Příkazy :

```
>> x=0: pi/20:2* pi ;
```

```
>> y1=sin(x);
```

```
>> plot(x,y1,'r')
```

```
>> y2=sin(x .*2);
```

```
>> y3=sin(x .*0.5);
```

```
>> hold on
```

```
>> plot(x,y2,'b,)
```

```
>> plot(x,y3,'k')
```

```
>> plot(x,y3,'k')
```

```
>> grid on
```

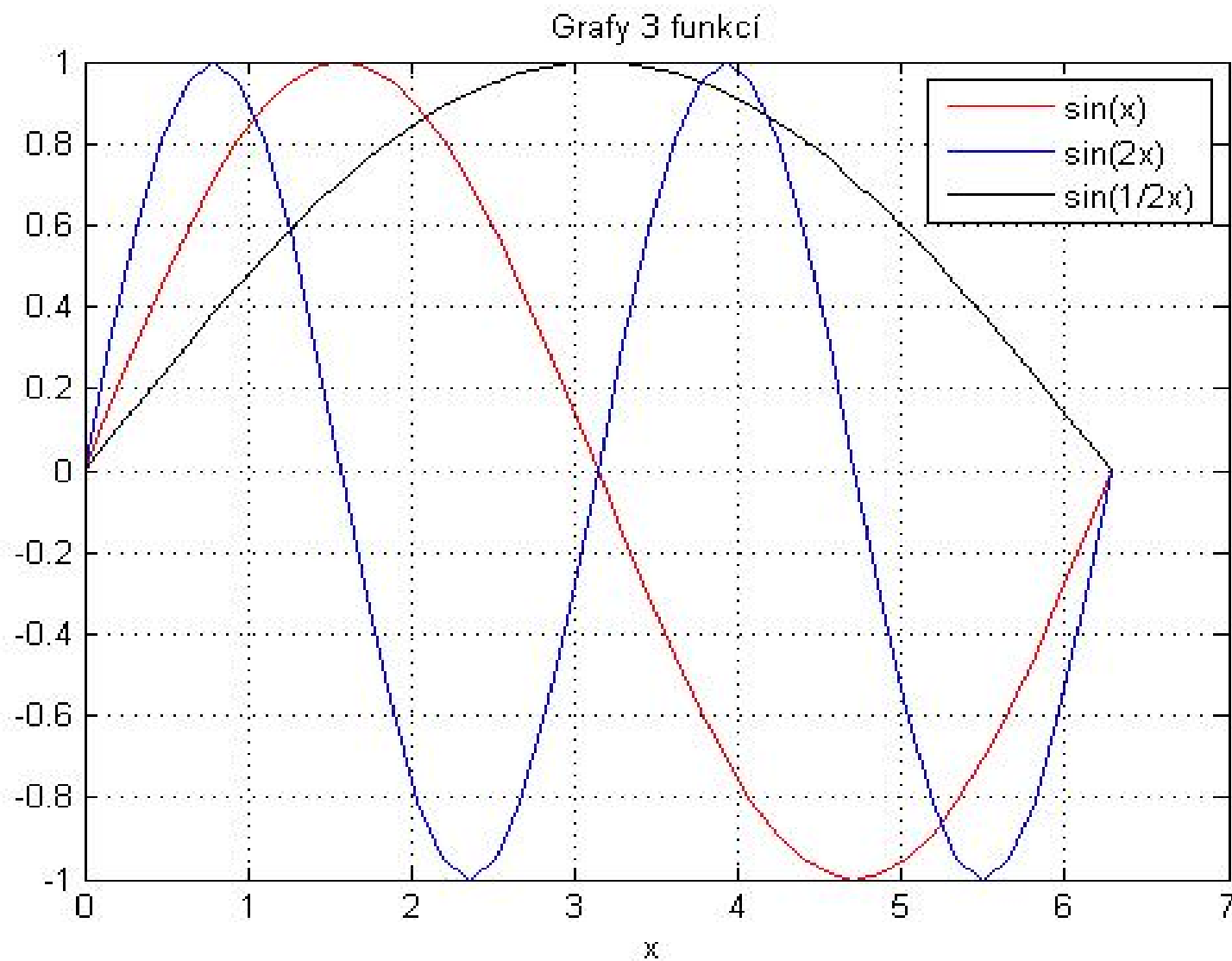
```
>> xlabel ('x')
```

```
>> title ('Grafy 3 funkcí')
```

```
>> legend('sin(x)',...
```

```
'sin(2x)','sin(1/2x)')
```

Grafika – 2D



GRAFY FUNKCÍ S RŮZNÝMI MĚŘÍTKY

Příkazy :

```
y1=[1:10];y2=[1:25]+2;  
x=[1:10];x1=[1:25];  
plot (x,y1,x1,y2);grid
```

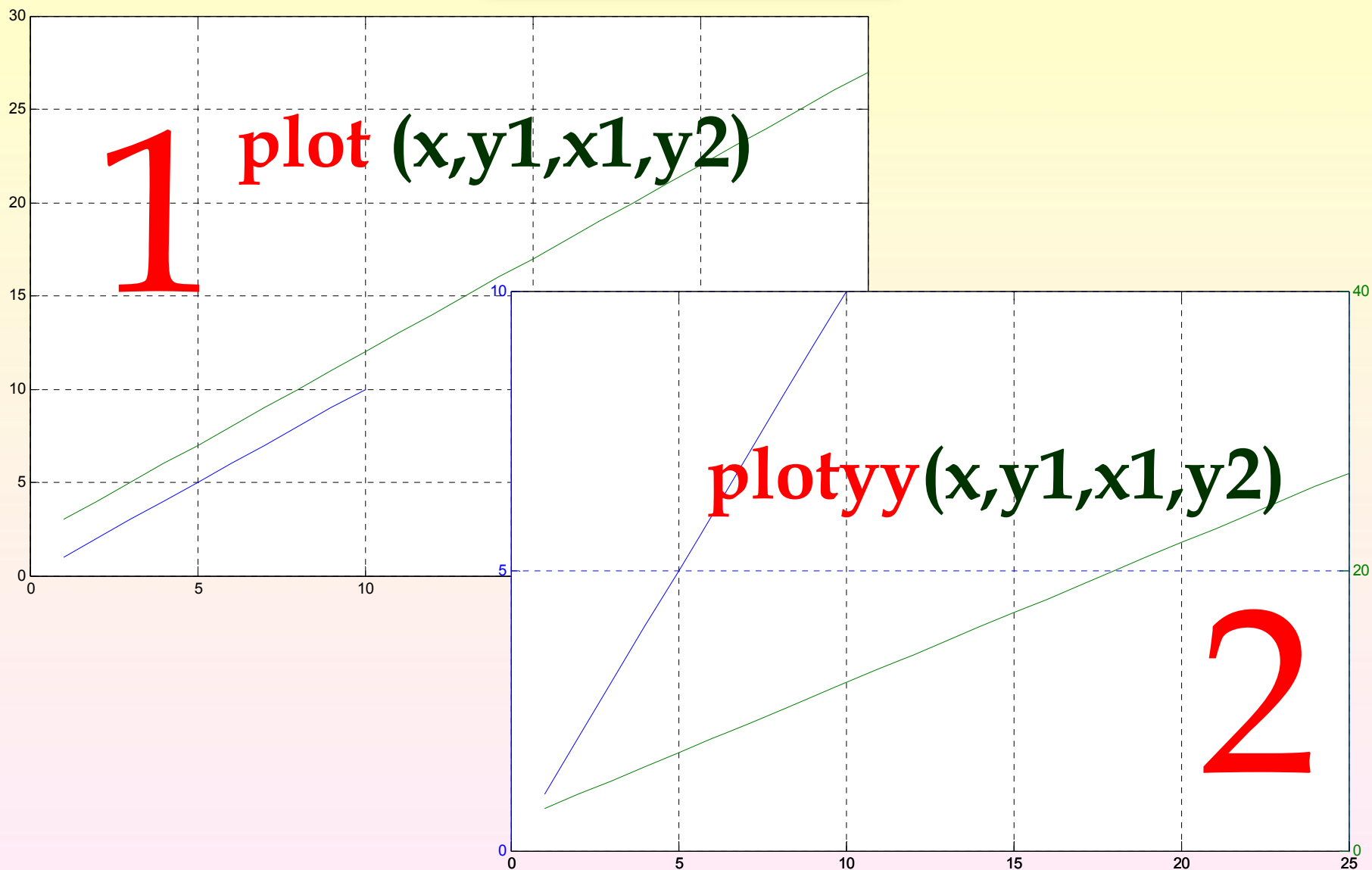
1

?

2

```
y1=[1:10];y2=[1:25]+2;  
x=[1:10];x1=[1:25];  
plotyy(x,y1,x1,y2);grid
```

Grafika – 2D



... A CO ...

PŘÍKAZ

LINE ?????

A S NÍM SOUVISEJÍCÍ

... že by DÚ ...

`plot3(x,y,z)`

`t = 0:pi/50:10*pi;`

`plot3(sin(t),cos(t),t)`

`axis square; grid on`

Grafika – 3D

```
[X,Y] = meshgrid([-2:0.1:2]);  
Z = X.*exp(- X.^2 - Y.^2 );  
plot3(X,Y,Z)  
grid on
```

```
[X,Y] = meshgrid(-8:5:8);  
R = sqrt(X.^2 + Y.^2) + eps;  
Z = sin(R)./R;  
mesh(X,Y,Z)
```

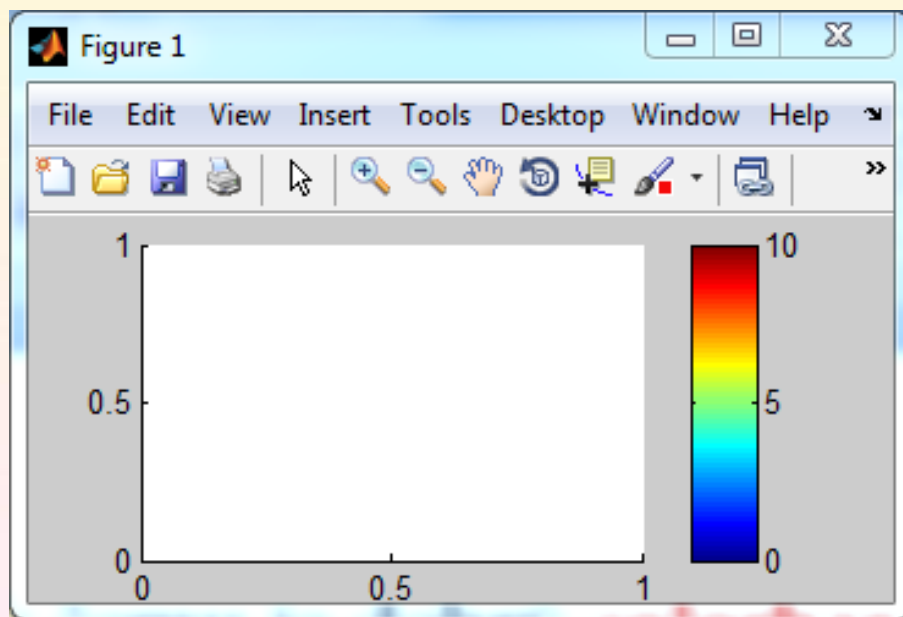
Grafika – 3D

```
[x,y] = meshgrid([-2:2:2]);  
Z = x.*exp(-x.^2-y.^2);  
surf(x,y,Z, gradient(Z) )  
colorbar
```

Grafika – 3D

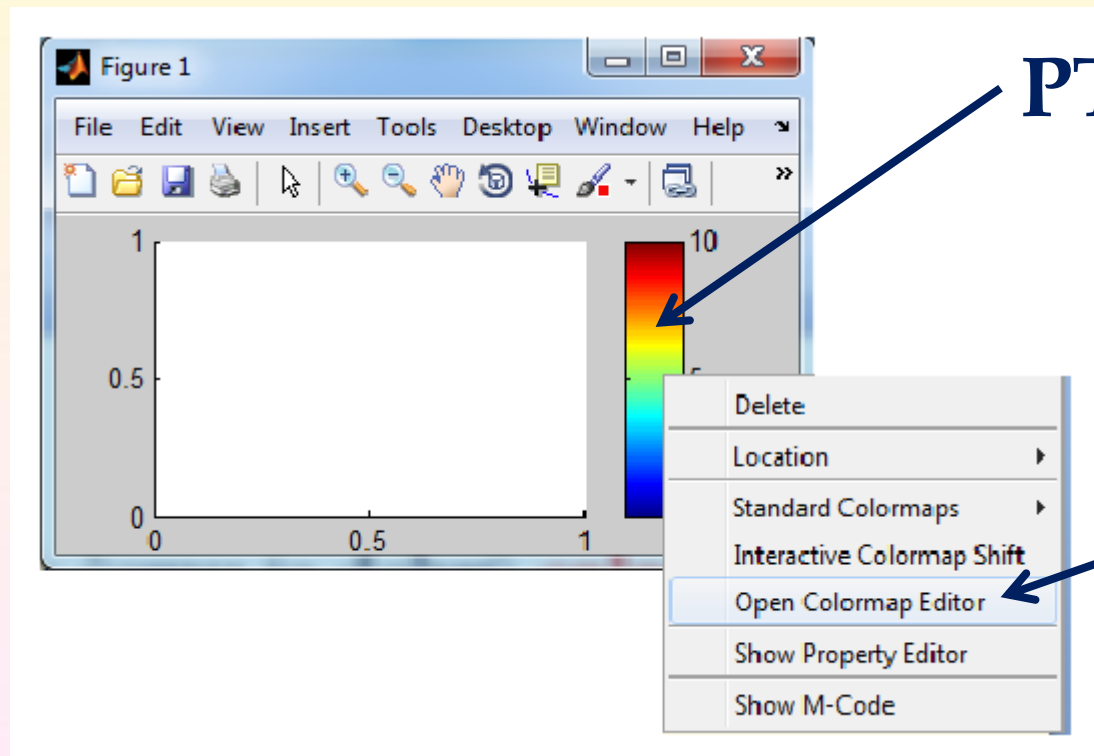
k čemu je dobrý **colorbar** ?

>> colorbar



Grafika – 3D

k čemu je dobrý **colorbar** ?

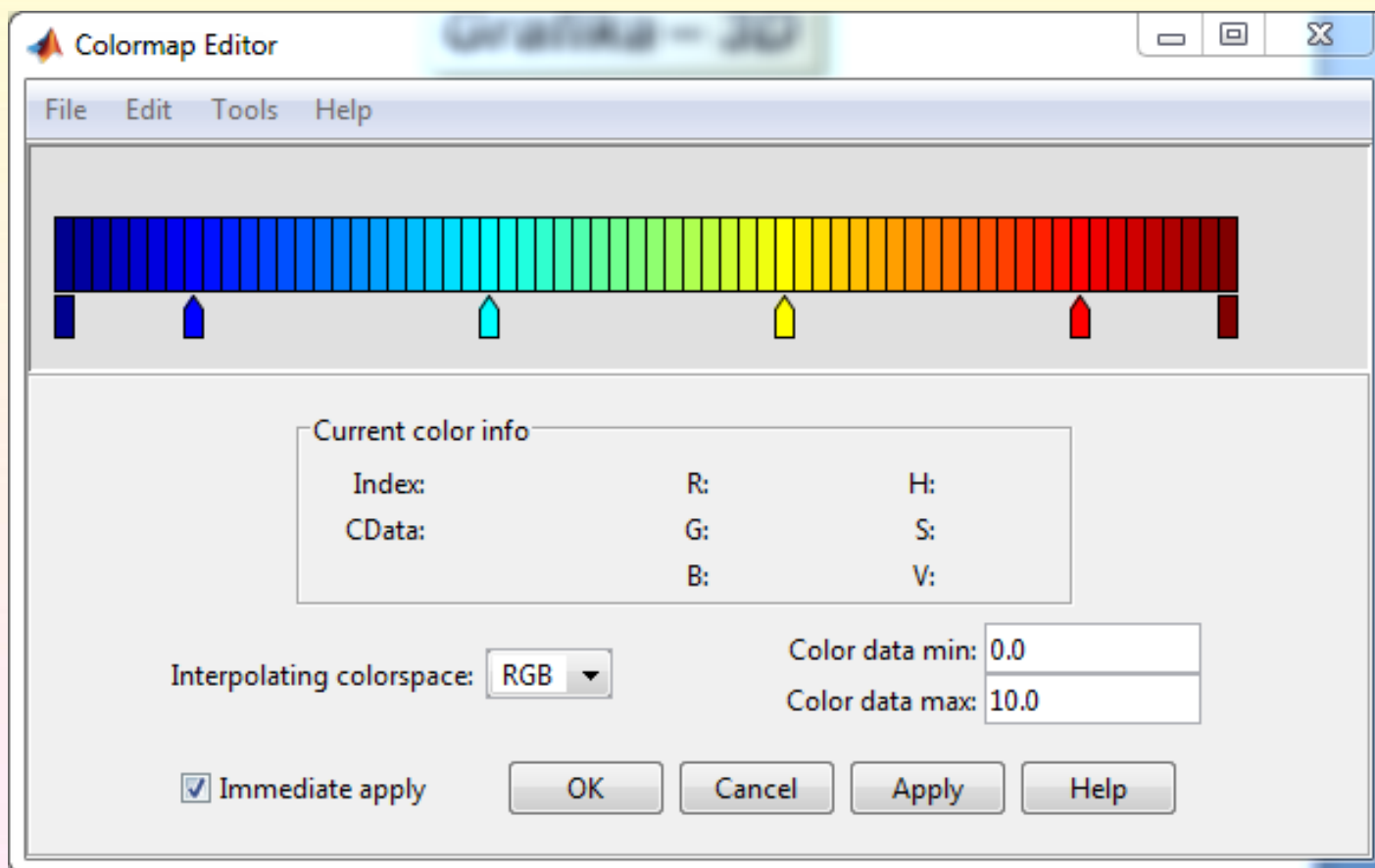


PTM

a potom
vyber

Grafika – 3D

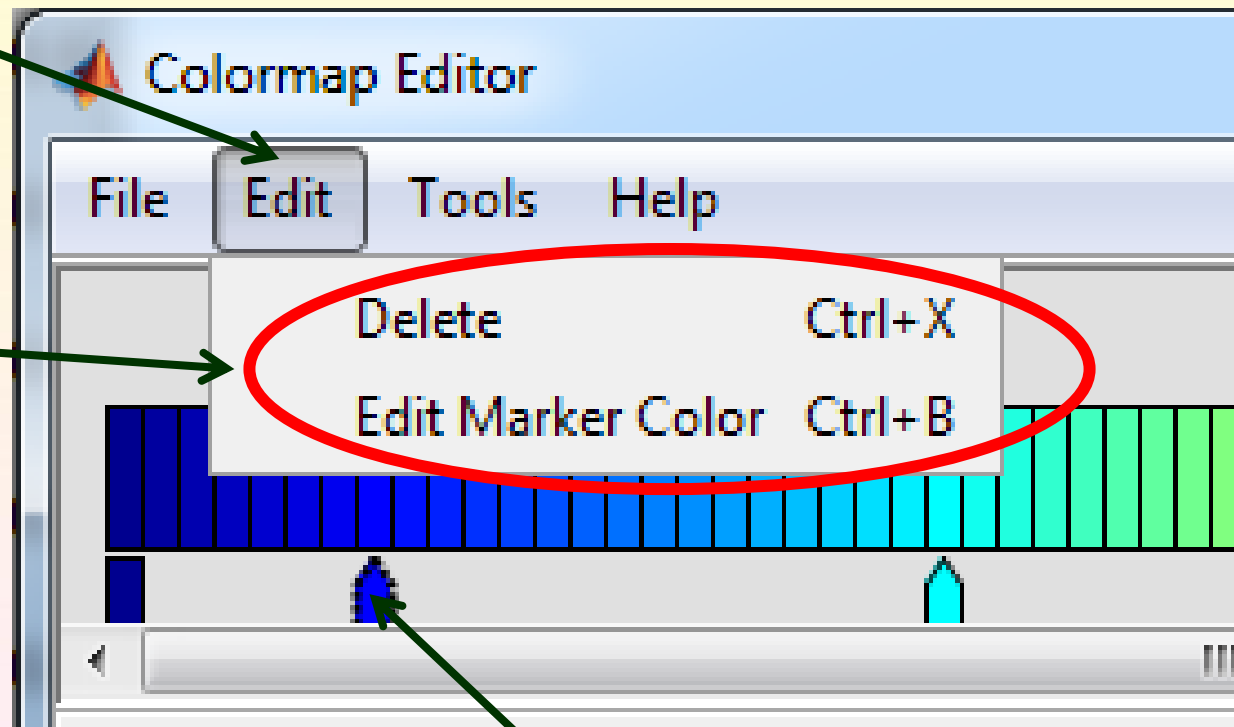
k čemu je dobrý **colorbar** ?



k čemu je dobrý **colorbar** ?

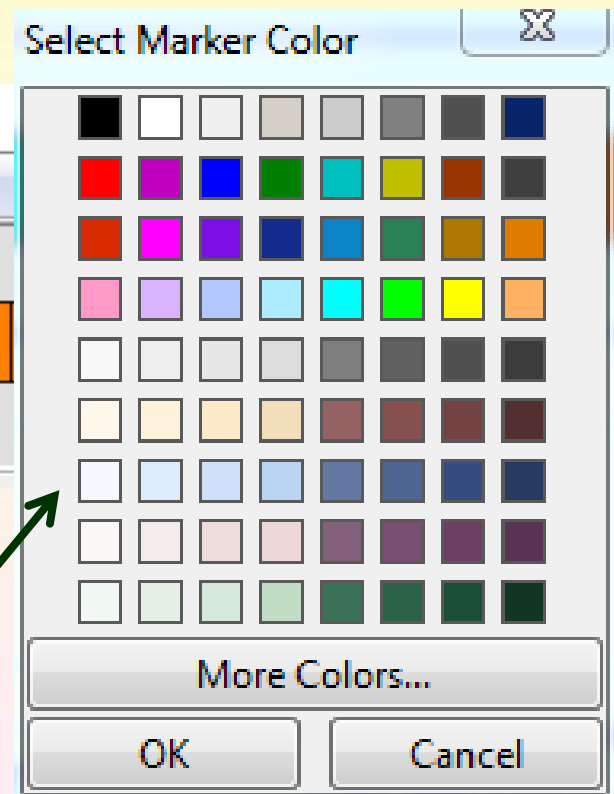
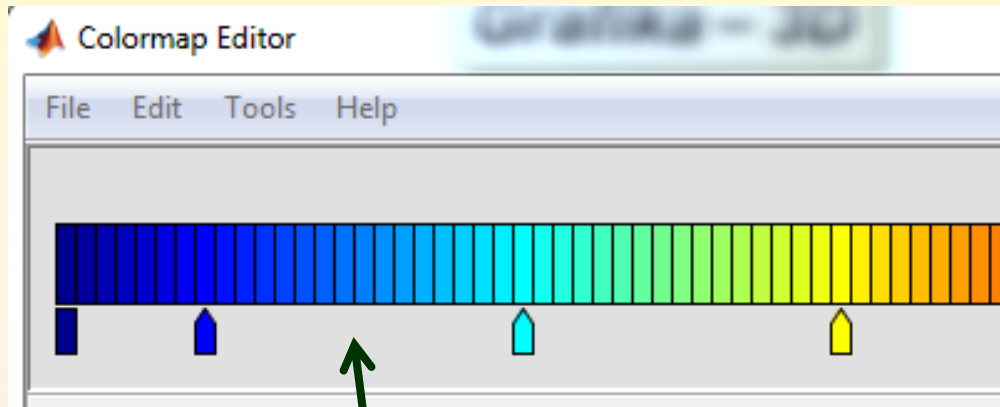
2 – otevřít **Edit**

3 – vyberu



1 – označit ukazatel barvy

k čemu je dobrý **colorbar** ?



přidání barvy – kliknu mezi
barvy a vyberu barvu

že by DÚ?

meshc surfc

meshz

surfl

surface

studente, jak skoro říkal
jeden nejmenovaný
člověk

trénuj, trénuj, trénuj

Větvení, rozhodování a cykly

if
while
for
switch



VÝSLEDEK SI NAJDI
v **HELPOVNÍKU**
(nápovědě)

Vždyť syntaxi znáš z jiných programovacích jazyků

Vytváření M-file

M - soubory

```
graph TD; A[M - soubory] --> B[příkazové]; A --> C[funkční];
```

příkazové

funkční

Vytváření M-file

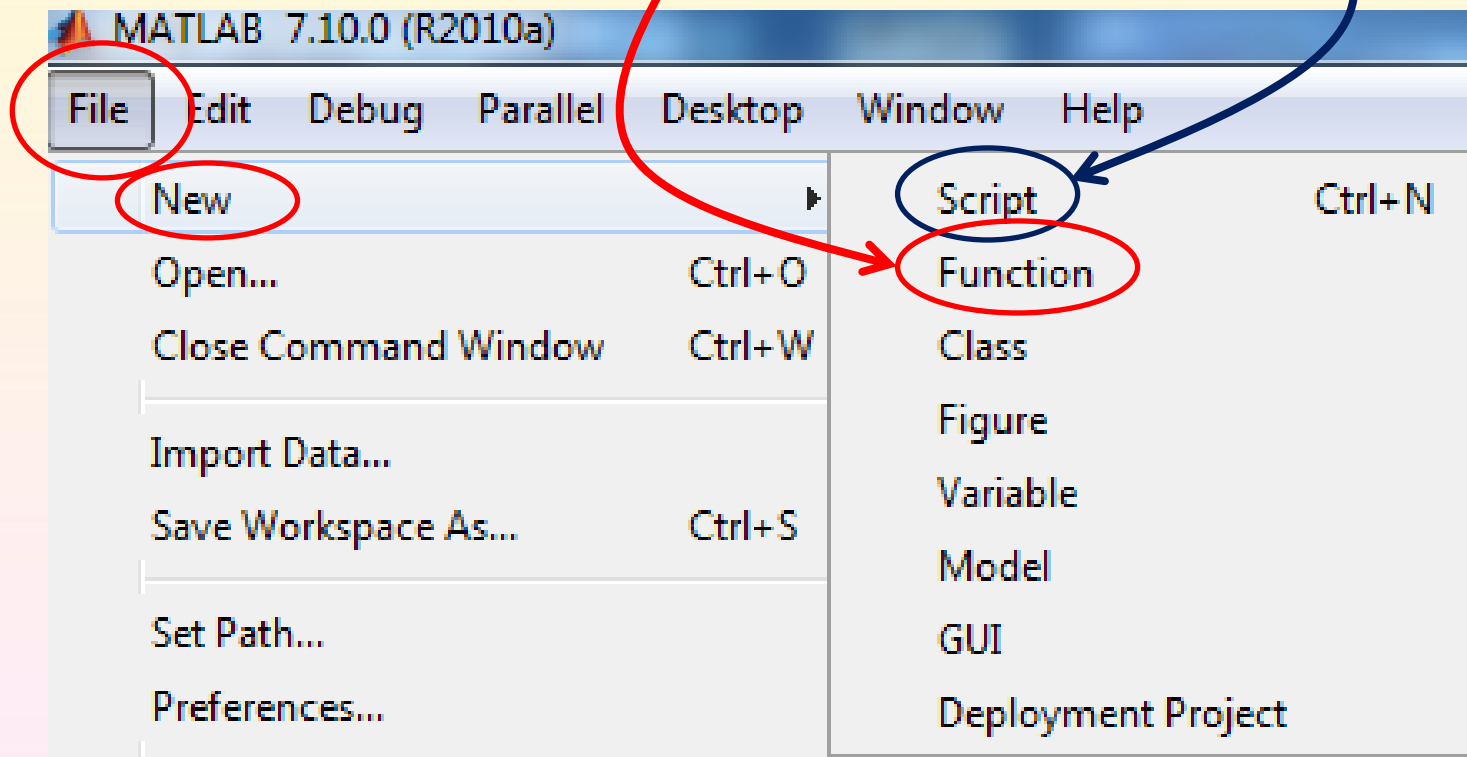
Rozdíl mezi příkazovým a funkčním souborem?

Příkazový M-file – proměnné jsou globální

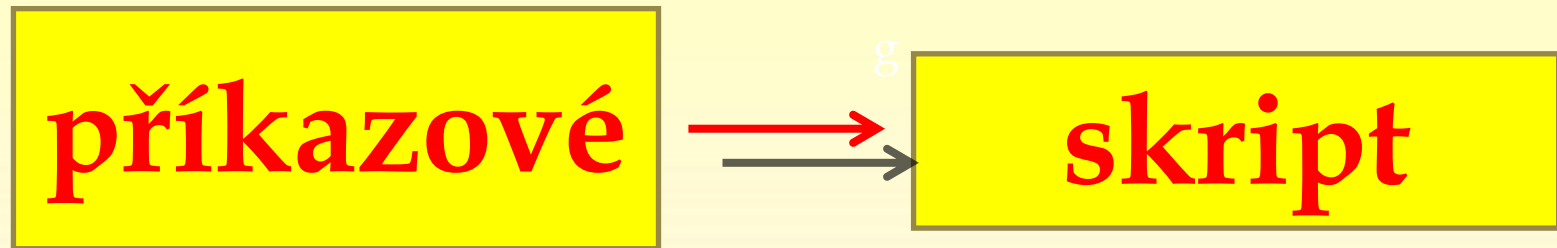
Funkční M-file – proměnné jsou lokální

Vytváření M-file

Vytvoření příkazového
nebo funkčního souboru.



Vytváření M-file

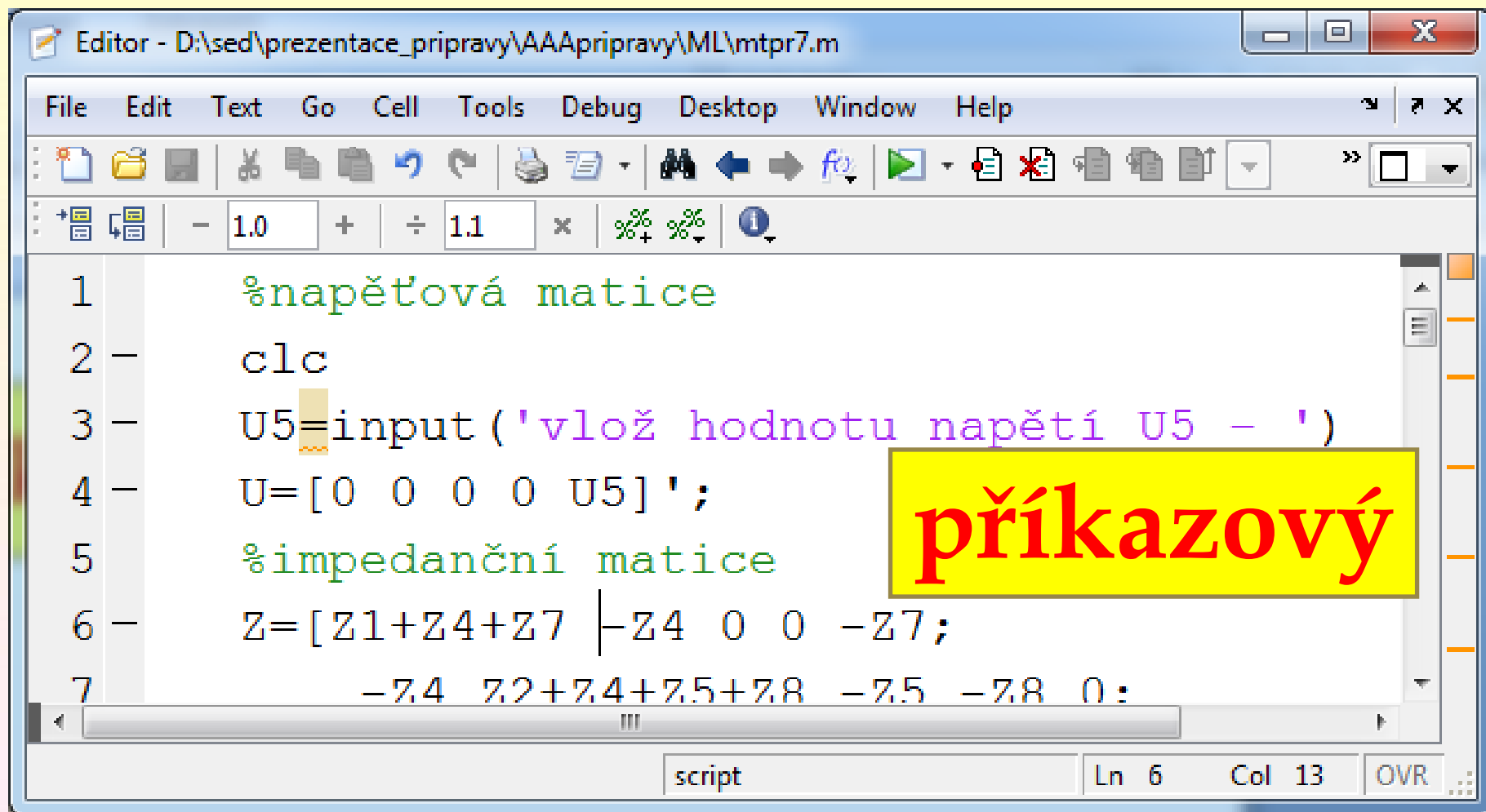


↓

Posloupnost příkazů
Musíme opravit hodnoty
Pracuje s proměnnými
pracovního prostředí



Vytváření M-file



Editor - D:\sed\prezentace_pripravy\AAApripravy\ML\mtp7.m

File Edit Text Go Cell Tools Debug Desktop Window Help

1 %napěťová matice
2 clc
3 U5=input('vlož hodnotu napětí U5 - ')
4 U=[0 0 0 0 U5]';
5 %impedanční matice
6 Z=[Z1+Z4+Z7 | -Z4 0 0 -Z7;
7 -Z4 Z2+Z4+Z5+Z8 -Z5 -Z8 0];

příkazový

script Ln 6 Col 13 OVR

Příklad skriptu

Název souboru: **Parabola.m**

```
% vykreslení paraboly  
x = -3:0.1:3;  
y = x.^2;  
plot(x,y)
```

příkazový

Vytváření M-file

funkční



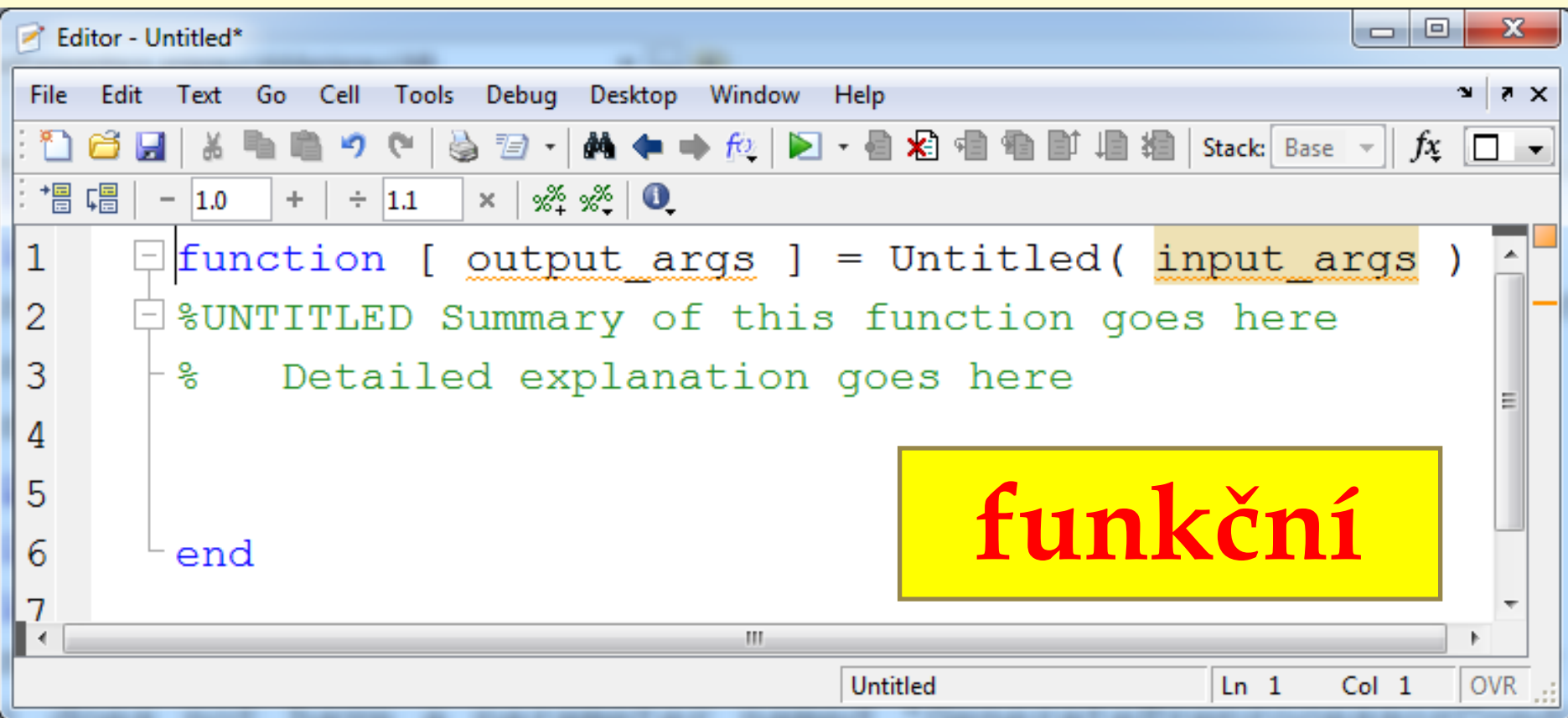
**Uživatelské
funkce**



Algoritmus pro různé situace
Vlastní pracovní prostředí
Má lokální proměnné
Vstupní a výstupní parametry

File  New  function

Vytváření M-file



The screenshot shows the MATLAB Editor window titled "Editor - Untitled*". The menu bar includes File, Edit, Text, Go, Cell, Tools, Debug, Desktop, Window, and Help. The toolbar contains various icons for file operations, editing, and execution. Below the toolbar is a numeric keypad with values 1.0, 1.1, and mathematical operators. The main editor area displays a function definition starting with "function [output_args] = Untitled(input_args)". The arguments "output_args" and "input_args" are underlined. The function body contains two comment lines: "%UNTITLED Summary of this function goes here" and "% Detailed explanation goes here". The function ends with "end". A yellow box with the red text "funkční" is overlaid on the right side of the editor. The status bar at the bottom shows "Untitled", "Ln 1", "Col 1", and "OVR".

```
1 function [ output_args ] = Untitled( input_args )
2 %UNTITLED Summary of this function goes here
3 % Detailed explanation goes here
4
5
6 end
```

funkční

Vytváření M-file

```
Editor - D:\sed\prezentace_pripravy\AAApripravy\ML\Ctverec.m

File Edit Text Go Cell Tools Debug Desktop Window Help

1 function [ O,S ] = Ctverec( a,b )
2     %popis fce dle pravidel
3     S=a*b;
4     O=2*(a+b);
5
6     end
7
```

Ctverec Ln 4 Col 1 OVR

k funkční

Vytváření M-file

```
[l,m]=Ctverec(10,5)
```

```
l =
```

```
30
```

```
m =
```

```
50
```

```
>>
```

```
a1=4;a5=7;[O,S]=Ctverec(a1,a5)
```

```
O =
```

```
22
```

```
S =
```

```
28
```

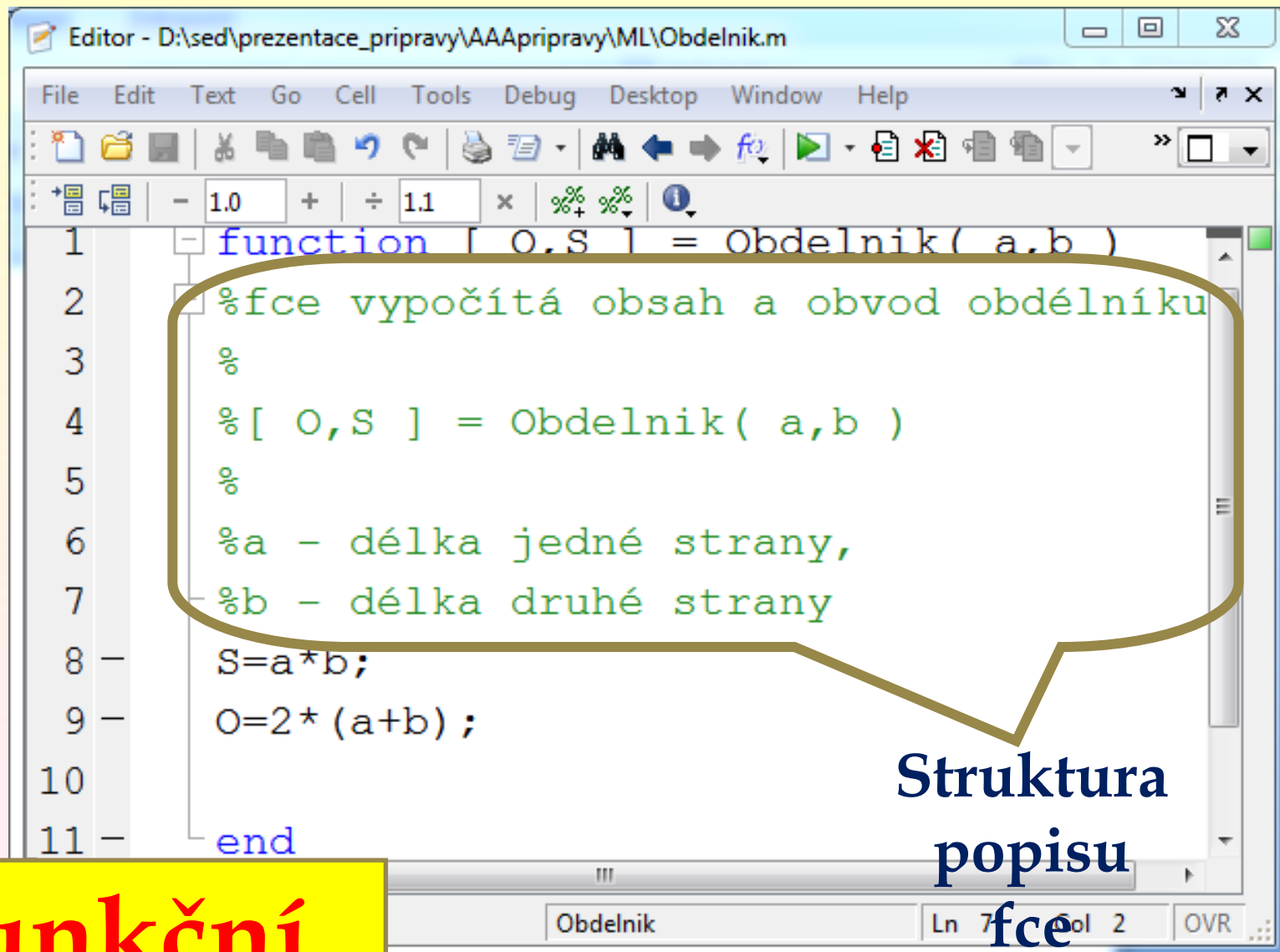
Příklad funkce

```
function f = faktorial(n)
    if(n<0)
        error('faktorial neni definovan');
    end
    f=1;
    while n>1
        f = f*n;
        n = n-1;
    end
end
```

Příklad funkce

```
function f = faktorial(n)
    if(n<0)
        error('faktorial neni
definovan');
    end
    f=1;
    while n>1
        f = f*n;
        n = n-1;
    end
end
```

Vytváření M-file



```
1 function [ O,S ] = Obdelnik( a,b )
2 %fce vypočítá obsah a obvod obdélníku
3 %
4 %[ O,S ] = Obdelnik( a,b )
5 %
6 %a - délka jedné strany,
7 %b - délka druhé strany
8 S=a*b;
9 O=2*(a+b);
10
11 end
```

Struktura popisu fce

Obdelnik Ln 7 Col 2 OVR

k funkční

Vytváření M-file

Napíšeš-li **help Obdelnik**, tak dostaneš tuto odpověď - *jméno fce*

fce vypočítá obsah a obvod obdélníku

[O,S] = Obdelnik(a,b)

a - délka jedné strany,

b - délka druhé strany

Tedy vše co bylo napsáno
v **poznámce** u **fce**.

funkční

Vytváření M-file

Dalším dobrým pomocníkem je příkaz **lookfor obsah**, tak dostaneš tuto odpověď -

část textu nacházející se v komentáři fce

klepni na jméno fce

Obdelnik

- fce vypočítá obsah a obvod obdélníku

Tedy vše co bylo napsáno
v **poznámce** u **fce**
v prvním řádku.

k funkční

Vytváření M-file

Dalším dobrým příkazem je příkaz
pause a **input**.

pause – čeká až se znovu lepne na klávesnici

$XYZ = \text{input}(\text{'text související s danou proměnnou - XYZ'})$

_k **funkční**

Řešení rovnic

`syms x`

`y=solve('x^2-5*x+5')`

`y =`

$$\begin{aligned} &5/2 - 5^{(1/2)}/2 \\ &5^{(1/2)}/2 + 5/2 \end{aligned}$$

Symbolická matematika

pretty(y)

$$\begin{array}{cc}
 + - & - + \\
 | & 1/2 | \\
 | & 5 \ 5 | \\
 | & - - - - | \\
 | & 2 \ 2 | \\
 | & | \\
 | & 1/2 | \\
 | & 5 \ 5 | \\
 | & - - - + - | \\
 | & 2 \ 2 | \\
 + - & - +
 \end{array}$$

Pozor na znaménka ...

a nebo

$y = \text{solve}('x^2 - 5x = -5')$

$y =$

$$\begin{aligned} &5/2 - 5^{(1/2)}/2 \\ &5^{(1/2)}/2 + 5/2 \end{aligned}$$

Takto můžeme řešit i soustavu n -rovníc o n -neznámých,
včetně diferenciálních.

Řešení limit

$\text{limit}(\sin(x)/x)$

ans =

1

Řešení limit

$\text{limit}(1/x, x, 0, \text{'right'})$

ans =

Inf

Řešení derivací

první derivace

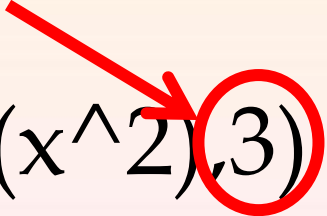
$\text{diff}(\sin(x^2))$

ans =

$2*x*\cos(x^2)$

třetí derivace

$\text{diff}(\sin(x^2), 3)$



ans =

$-12*x*\sin(x^2) - 8*x^3*\cos(x^2)$

`int(-2*x)`

Řešení integrálů

`ans =`

$-x^2$

`int(-2*x/(1 + x^2)^2)`

`ans =`

obecný

$1/(x^2 + 1)$

Řešení integrálů

$\text{int}(x \cdot \log(1 + x), 0, 1)$

ans =

určitý

$1/4$

Co s tím se všechno dá
řešit?

Je to závislé od toho, co
vše potřebuješ vyřešit
a jestli si to najdeš
v *HELPOvníku*

Proč to potřebujeme?

- 1) Uložení dat
- 2) Použití získaných dat pro další zpracování
- 3) Použití dat v jiných programech - aplikacích

Vytvoříme matici(e)

```
>> A=[2 4 5;6 5 2;1 2 1;1 4 0]
```

4 – řádky a 3 – sloupce ...

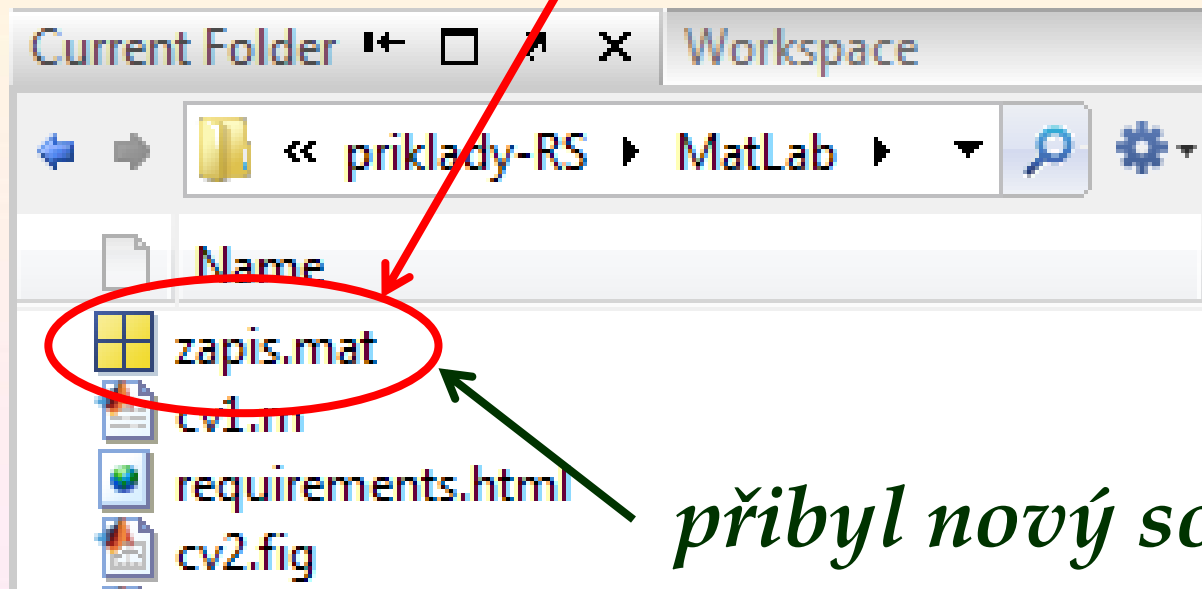
```
>> B=[1 2 1;1 4 0]
```

```
>> save zapis A B
```

a je zapsáno

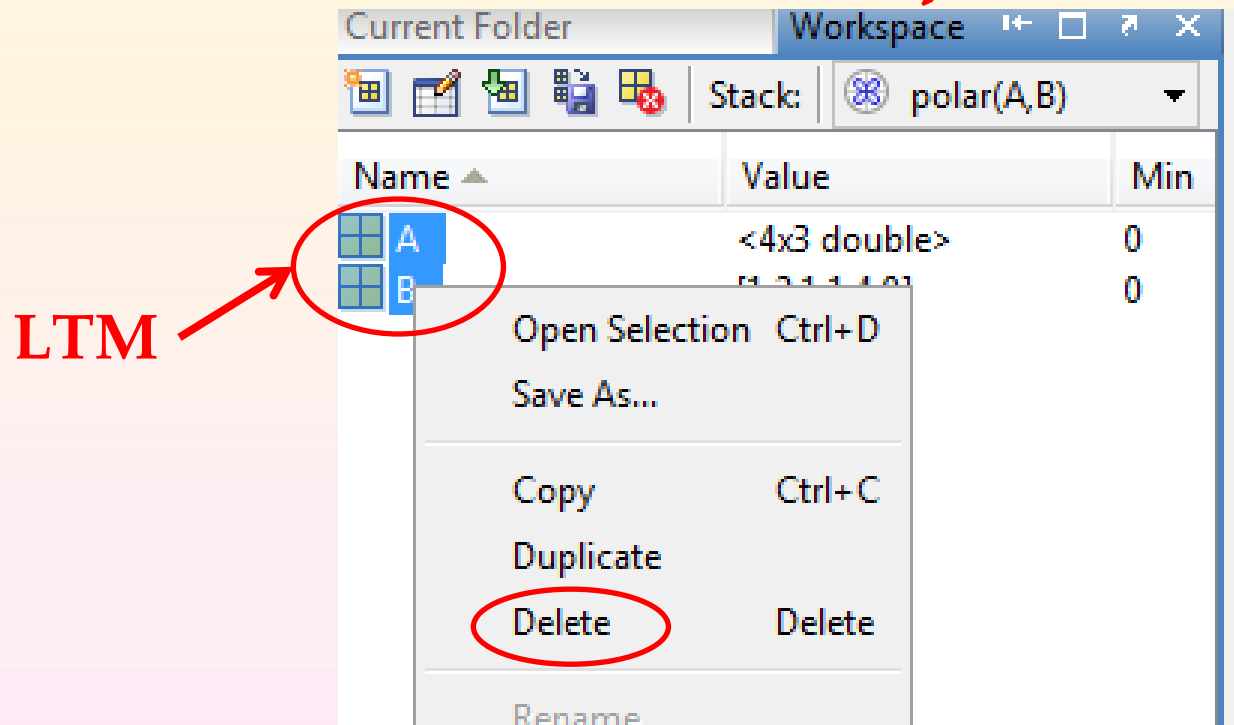
Čtení a zápis do souboru

a takto vypadá okno
aktivního adresáře



Čtení a zápis do souboru

a nyní budeme číst,
ale nejprve vymažeme
matice **A**, **B**



Čtení a zápis do souboru

```
>> A
```

```
??? Undefined function or variable 'A'.
```

```
>> load zapis
```

```
>> A,B
```

```
A =
```

```
2 4 5
```

```
6 5 2
```

```
1 2 1
```

```
1 4 0
```

```
B =
```

```
1 2 1
```

```
1 4 0
```

a po výmazu matic A a B ?
A po naloudování?



Čtení a zápis do souboru

a ještě to vyzkoušíme
s EXCELeM

```
>> xlswrite('cv1.xls',A,'List1')
```

soubor se vytvoří v aktivní složce,
ale ve verzi 1997 - 2003

Čtení a zápis do souboru

a ještě to vyzkoušíme
s EXCELeM

```
>> xlsread('cv1.xls','List1','a1:b3')
```

ans =

2 4

6 5

1 2

A =

2	4	5
6	5	2
1	2	1
1	4	0

Čtení a zápis do souboru

a ještě existují další
způsoby zápis,
ale to si najdi
v **HELP**ovníku

Základ je v **diferenciálních** rovnicích.
Z nich se vytvoří rovnice v **LT** a potom
se přepíše na přenos.
To je základ pro použití v **ML**.

$$F(s) = \frac{1}{2 + 3s + 5s^2 + s^3}$$

$$F = tf([čitatel], [jmenovatel])$$

$$F=tf(c,j)$$

$F = \text{tf}([1], [1 \ 5 \ 3 \ 2])$

`bode(F, {w_min, w_max})`

`[Amplituda, Faze] = bode(...)`

`nichols(F, {w_min, w_max})`

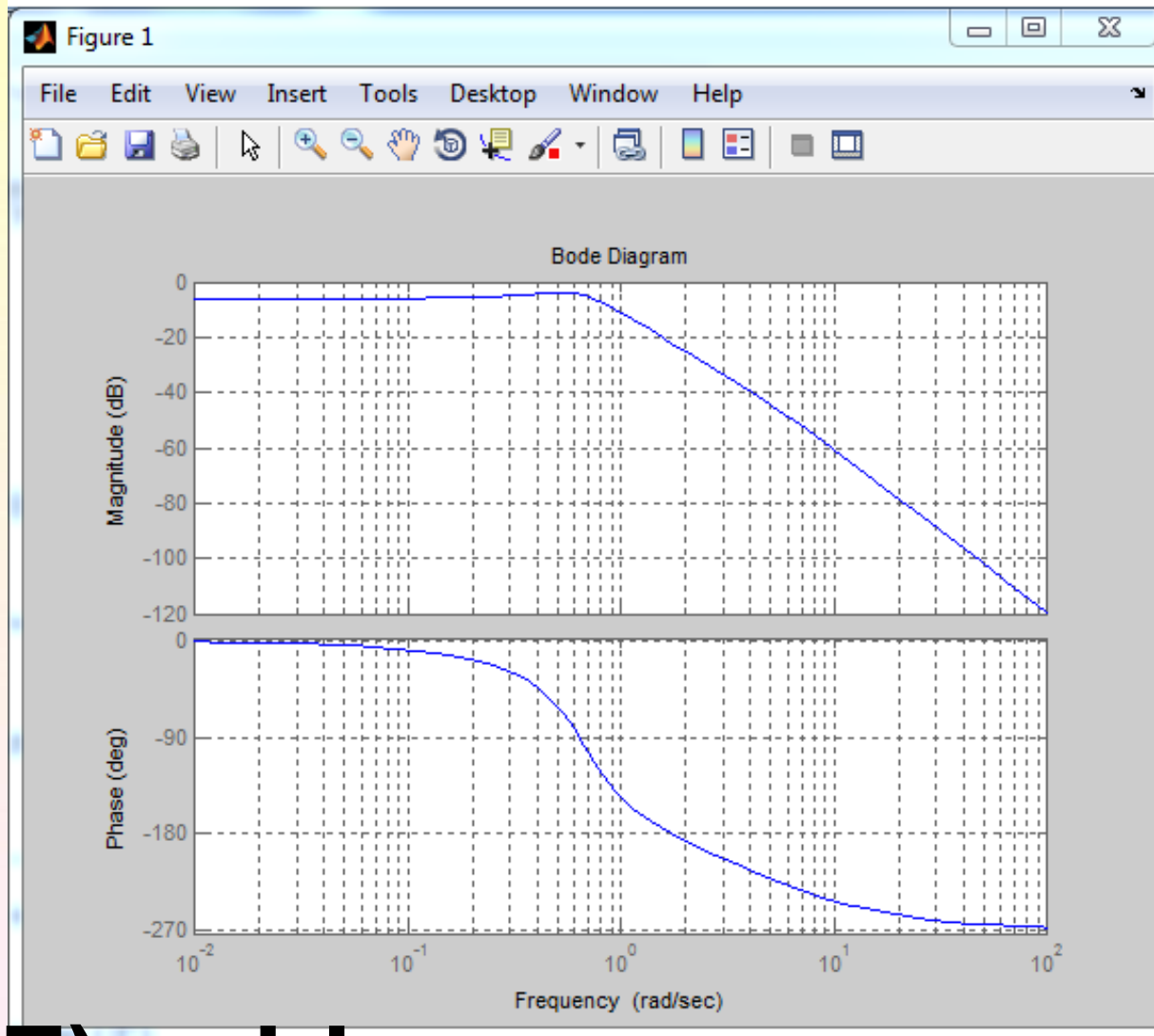
`nyquist(F, {w_min, w_max})`

!!!POZOR !!!

- hodnoty se
píší
od nejvyšší
mocniny

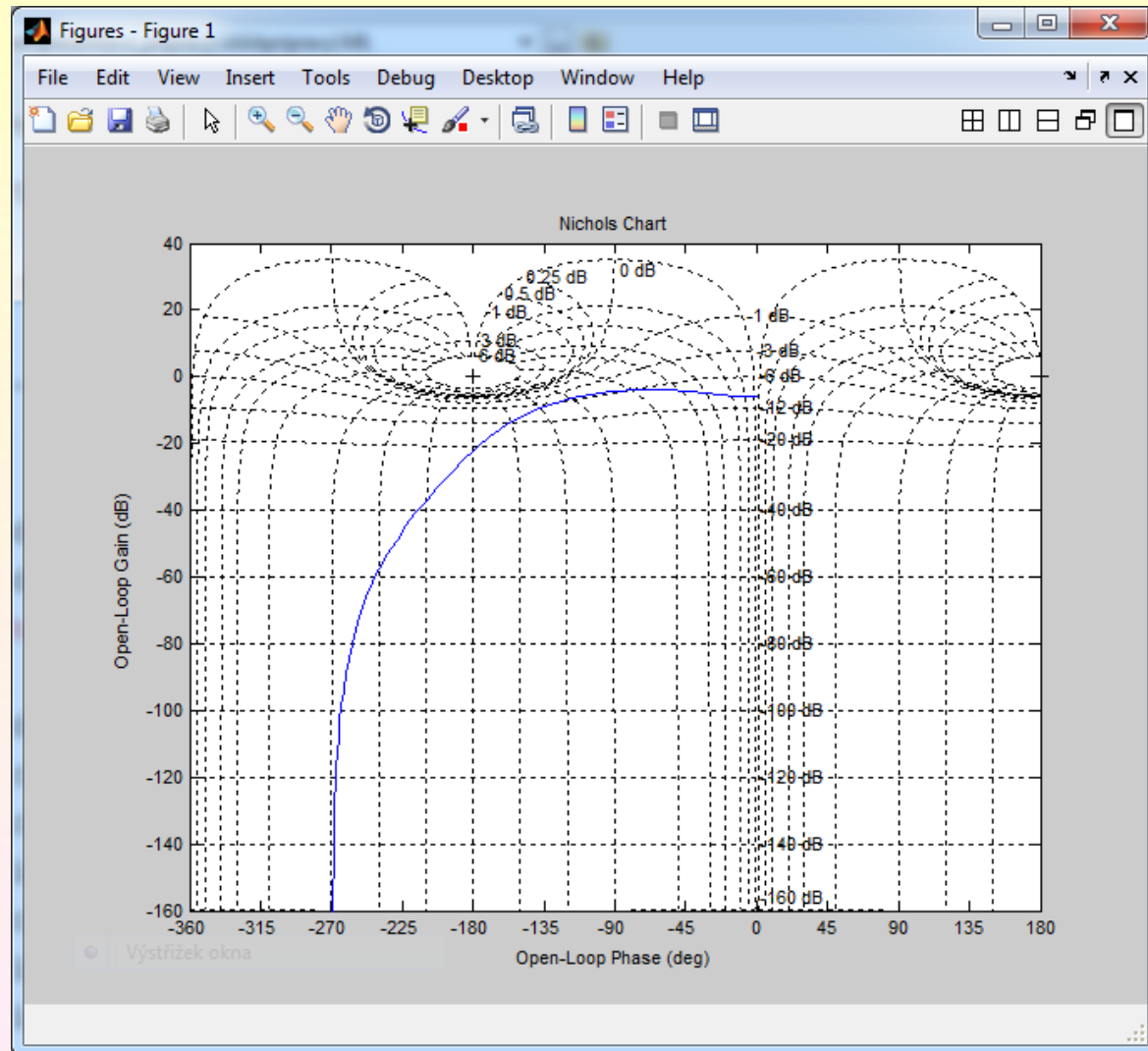


Řídicí technika



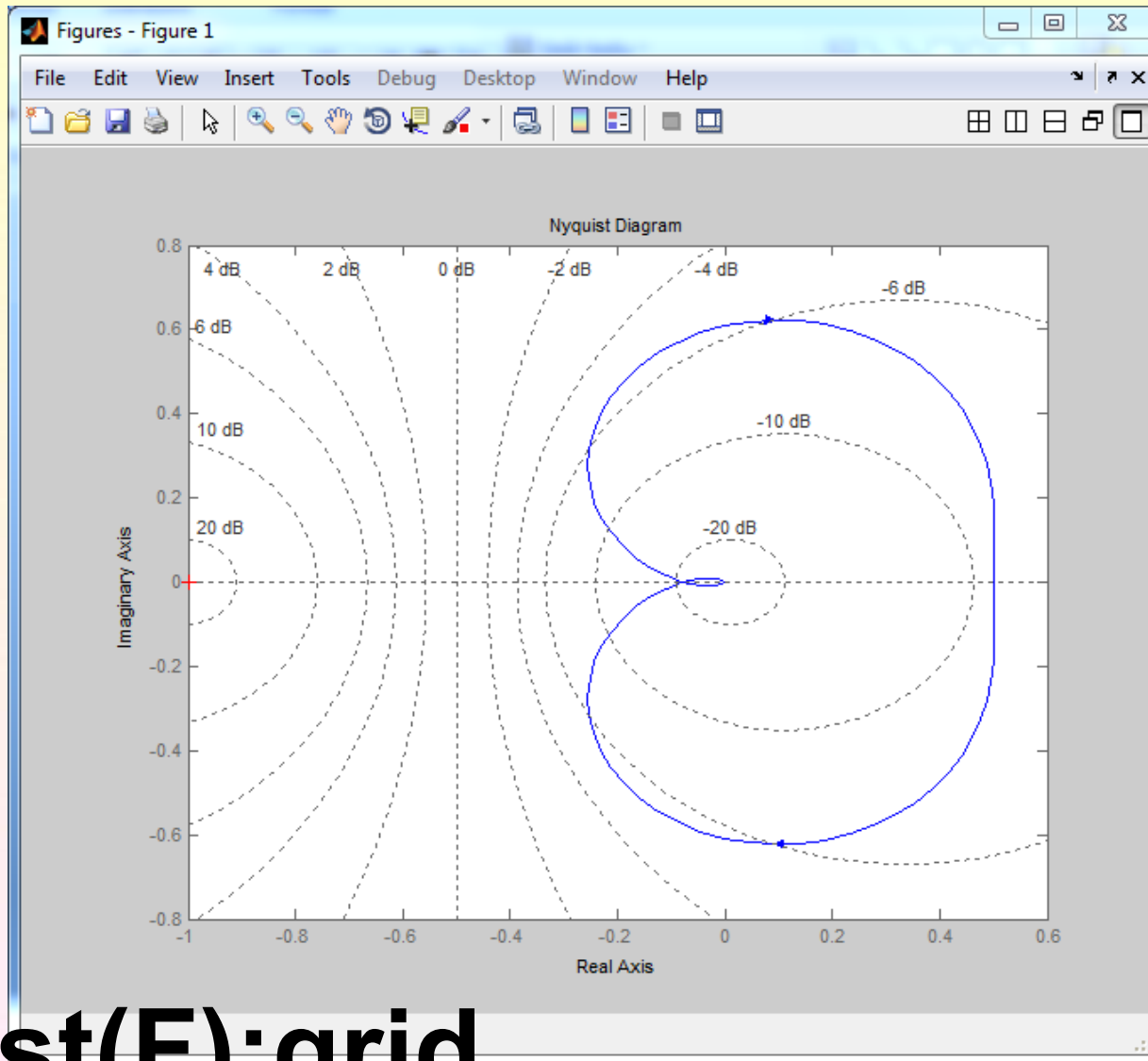
`bode(F);grid`

Řídicí technika



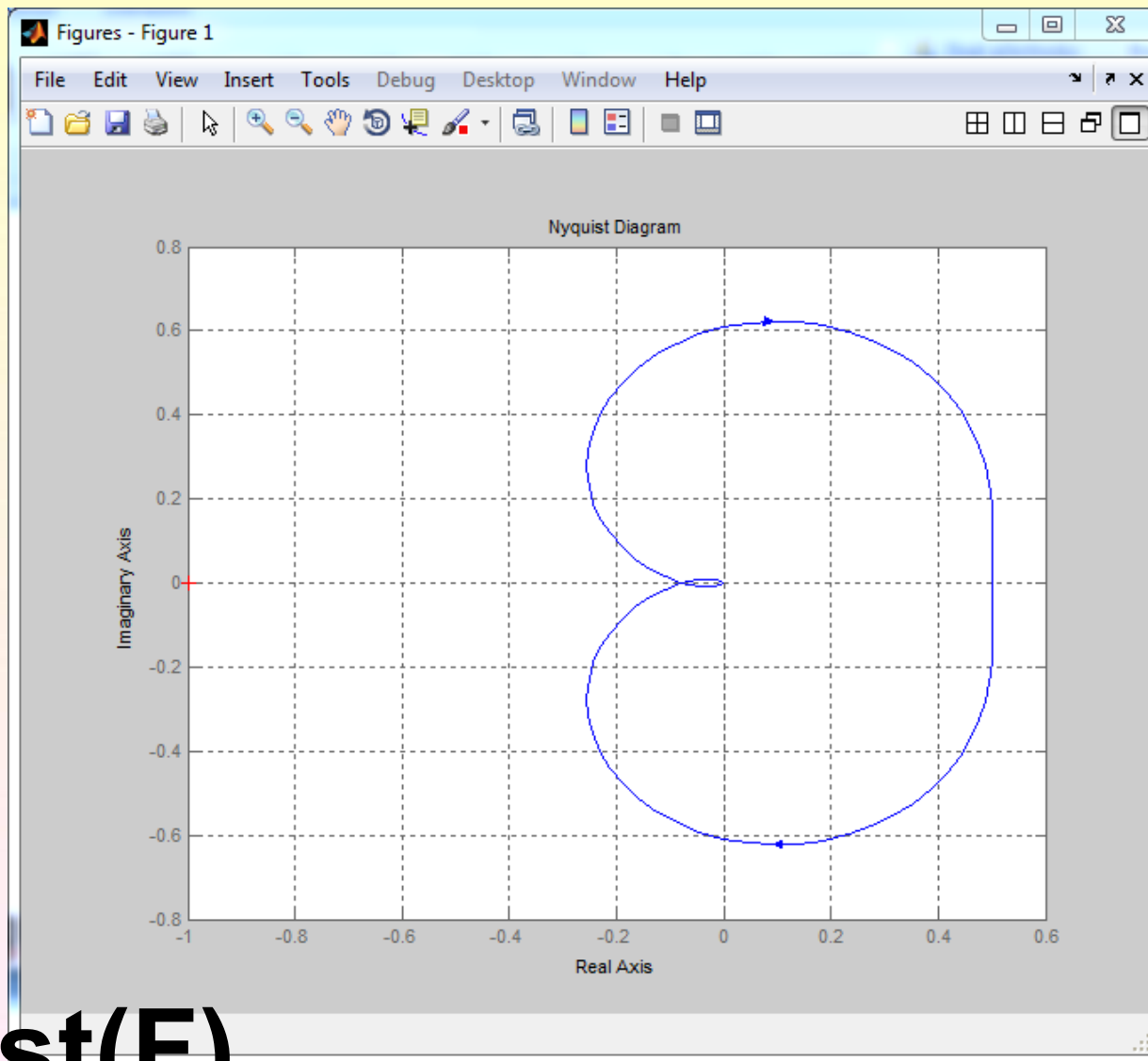
`nichols(F);grid`

Řídicí technika



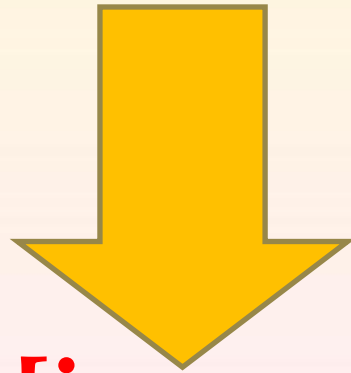
`nyquist(F);grid`

až po
úpravě
grafu
(v
grafu)



nyquist(F)

Známe-li kořeny čitatele, jmenovatele
a zesílení,
tak můžeme použít tento příkaz



$F = \text{zpk}([\text{čitatele}], [\text{jmenovatele}], \text{zesílení})$

$F = \text{zpk}(c, j, k)$

$$F = \text{zpk}([-0.5], [0 \ -5 \ -2], 3)$$

přenos má tvar →

$$\frac{3 (s+0.5)}{s (s+5) (s+2)}$$

rlocus(F)

převod mezi zpk a tf

$$[c,j]=zp2tf(z,p,k)$$

$$[z,p,k]=tf2zp(c,j)$$

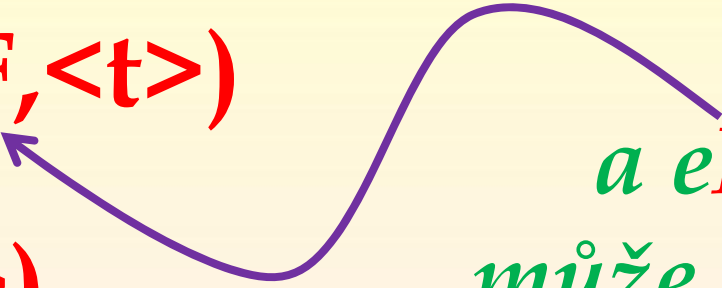
analýza v časové oblasti

impulse(F,<t>)

step(F,<t>)

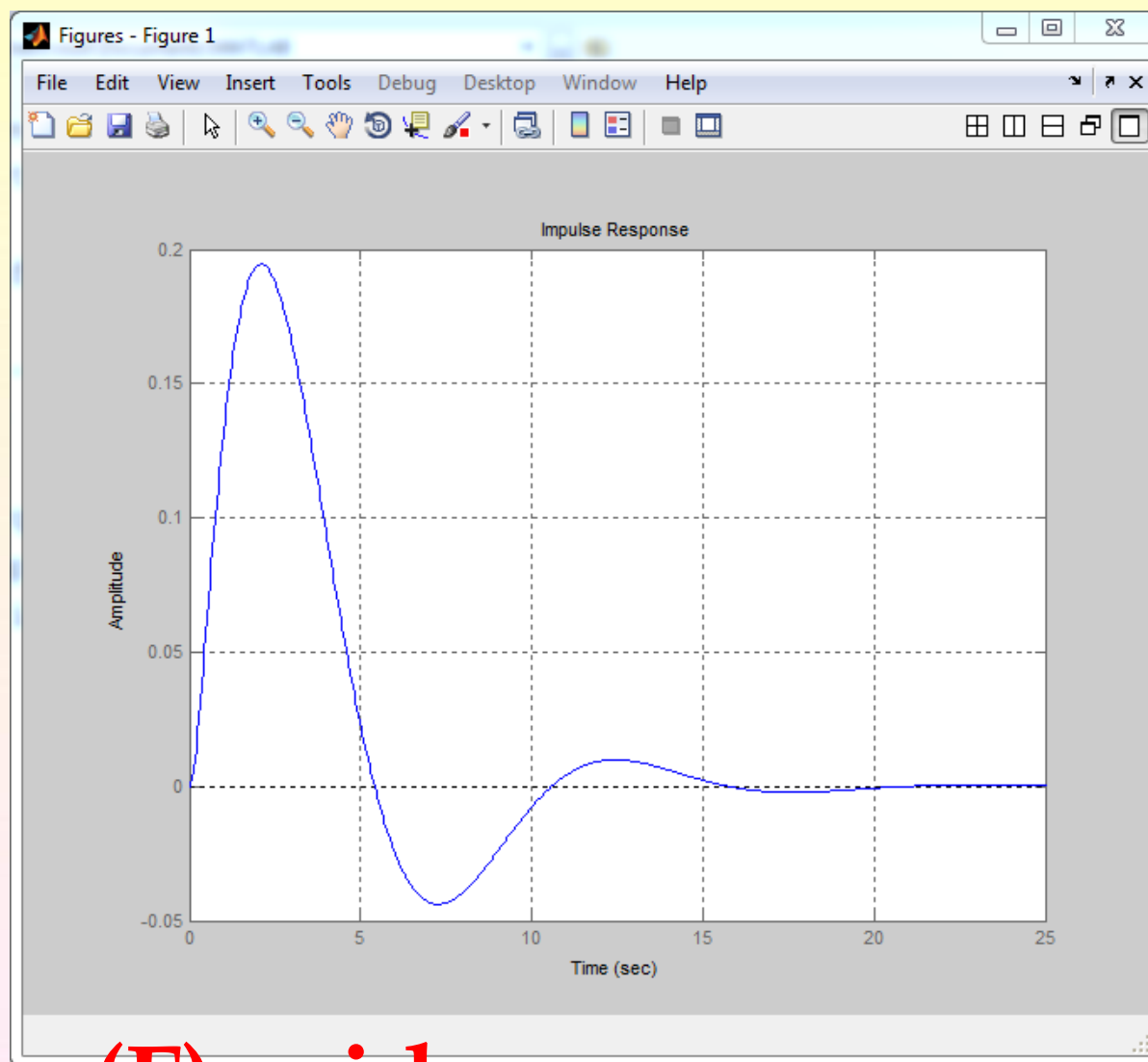
nichols (F,<t>)

*a eFek
může být
i více*



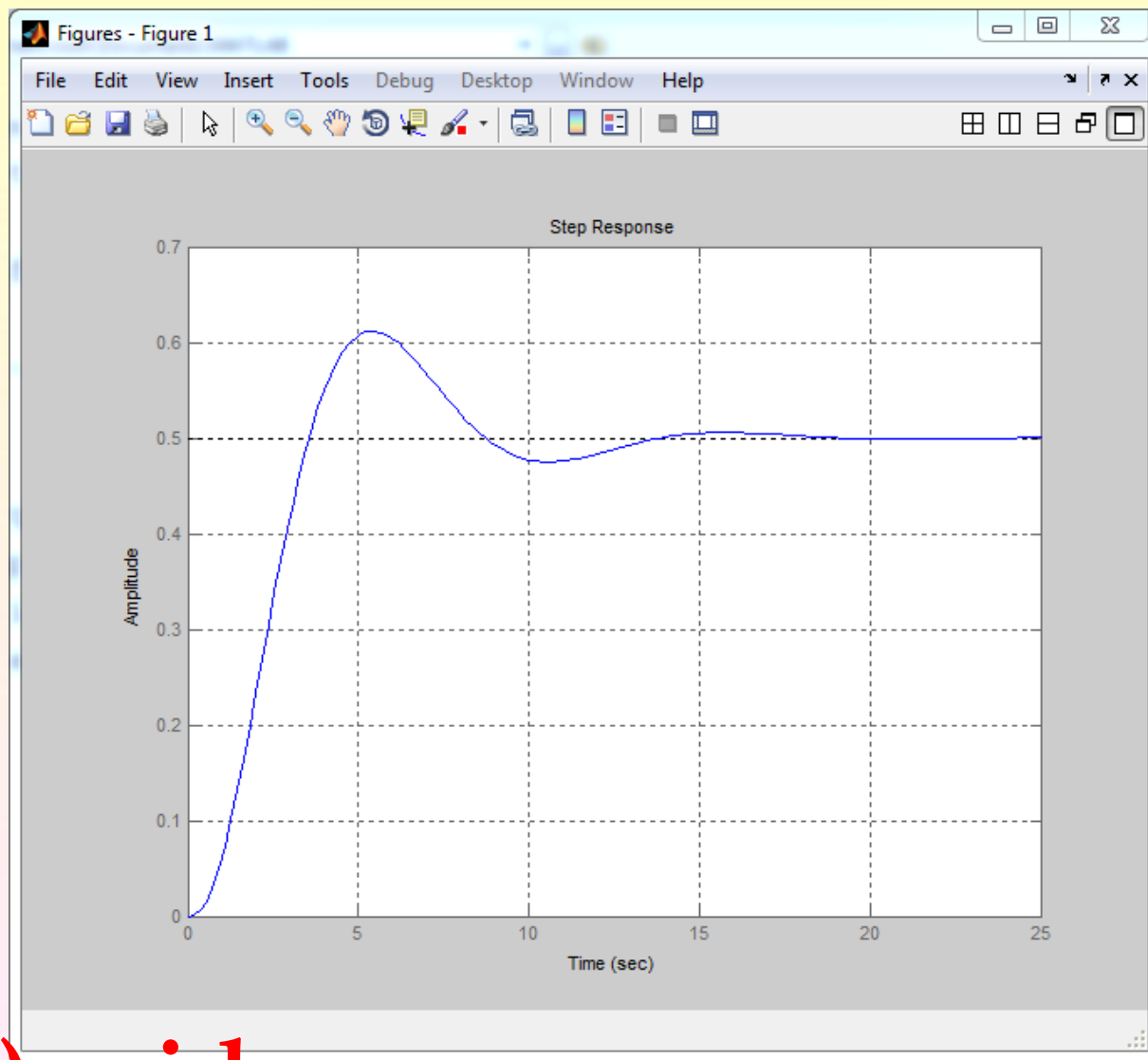
t – doba simulace nebo se zadá vektor a ten zajistí výpočty ve zvolených hodnotách

Řídicí technika



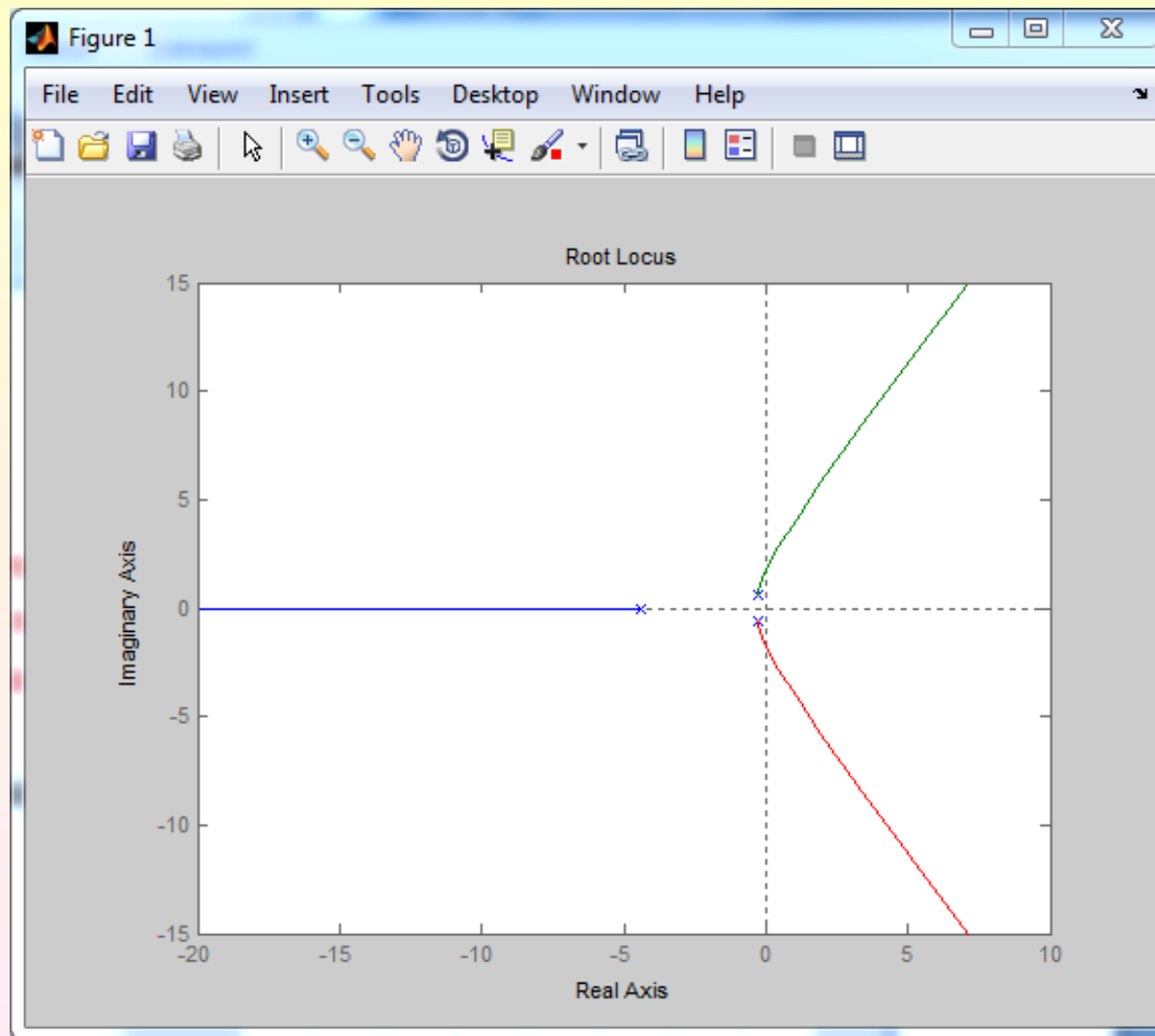
$k_{\text{impulse}}(F)$, grid

Řídicí technika



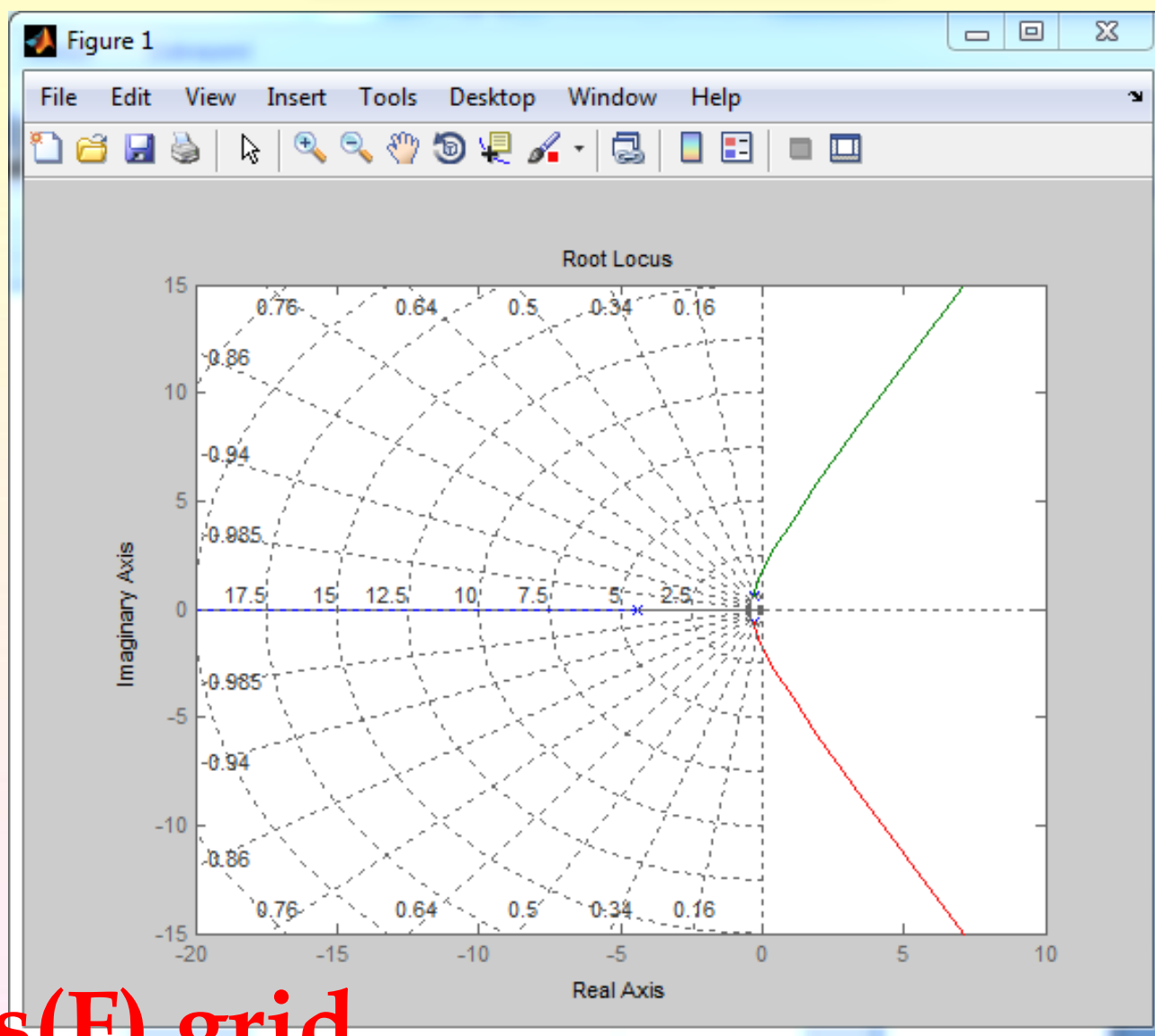
$k_{\text{step}}(F), \text{grid}$

Řídicí technika



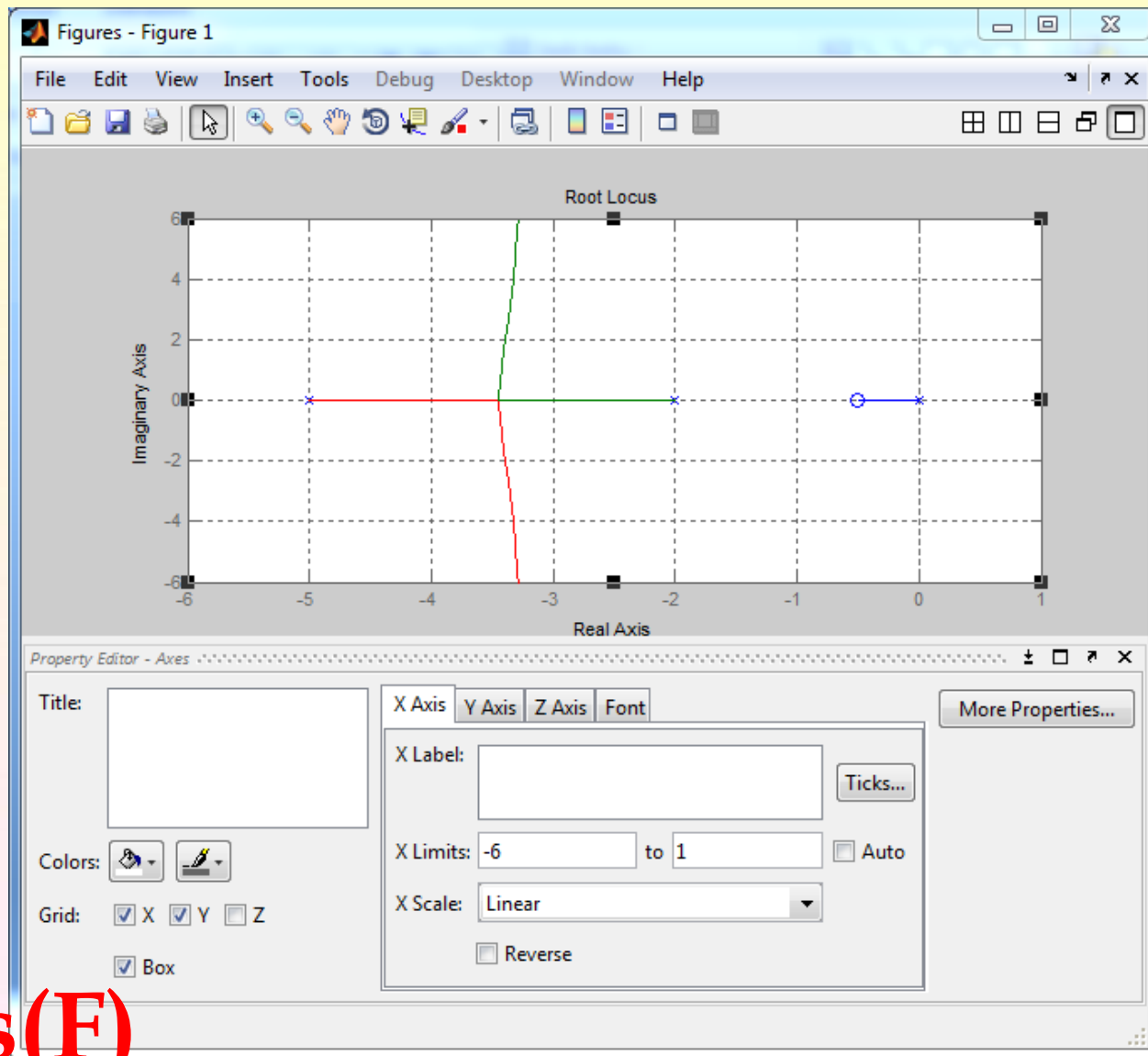
$k_r \text{locus}(F)$

Řídicí technika



`zrlocus(F),grid`

Řídicí technika



RS dvoukapacitní

... a když jsme již „**konečně**“
pochopili funkce
pro vykreslení přechodových
charakteristik,
tak než si začneme „**malovat**“
různé charakteristiky,
tak si uděláme nejdříve rozbor
tohoto typu soustavy ...

RS dvoukapacitní

1. Aperiodická - $t_1 \neq t_2$

2. Na mezi aperiodicity - $t_1 = t_2$

3. Kmitavá - $t = Re \pm jIm$

odvození

RS dvoukapacitní

ad 1) $t_1 \neq t_2$

$$(s + t_1)(s + t_2) =$$

$$= s^2 + st_2 + st_1 + t_1t_2 =$$

$$= s^2 + s(t_1 + t_2) + st_1t_2$$

$$a_2 = 1$$

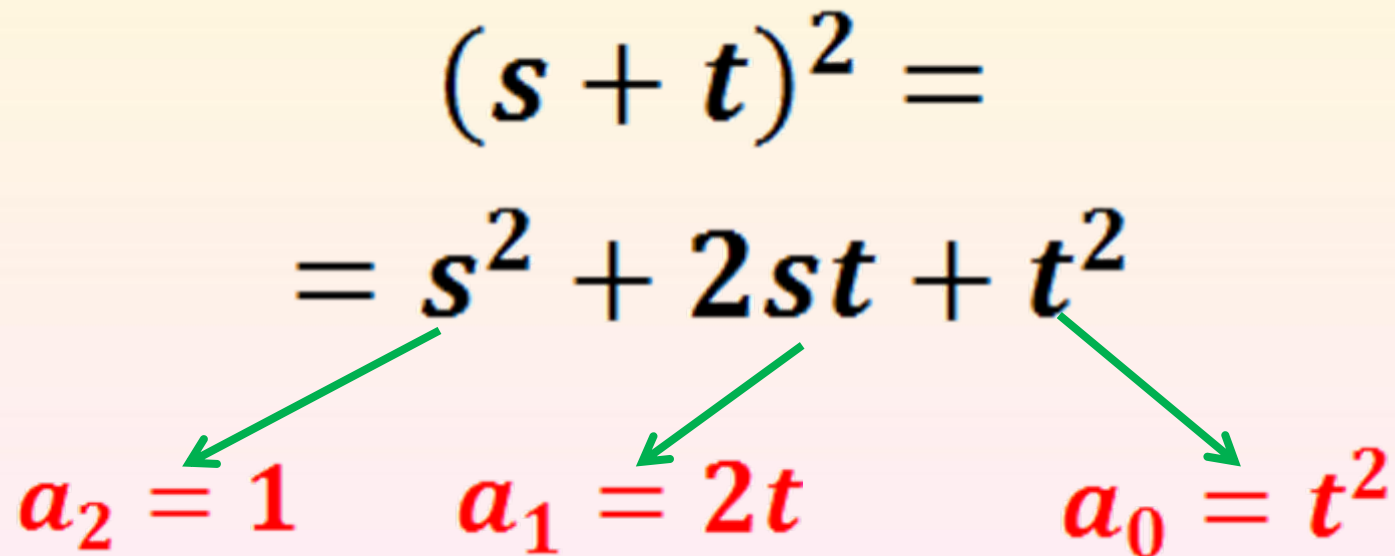
$$a_1 = t_1 + t_2$$

$$a_0 = t_1t_2$$

odvození

RS dvoukapacitní

ad 2) $t_1 = t_2$

$$\begin{aligned}(s + t)^2 &= \\ &= s^2 + 2st + t^2\end{aligned}$$


The diagram illustrates the expansion of the polynomial $(s + t)^2$. The expansion is shown as $s^2 + 2st + t^2$. Three green arrows point from the terms in the expansion to the coefficients a_2 , a_1 , and a_0 respectively. The first arrow points from s^2 to $a_2 = 1$. The second arrow points from $2st$ to $a_1 = 2t$. The third arrow points from t^2 to $a_0 = t^2$. All coefficient expressions are in red.

$$a_2 = 1 \quad a_1 = 2t \quad a_0 = t^2$$

odvození

RS dvoukapacitní

ad 3) $t_{1,2} = Re \pm jIm$

$$\begin{aligned}(s + Re + jIm)(s + Re - jIm) &= \\ s^2 + sRe - \cancel{sIm} + sRe + Re^2 - \\ - \cancel{ReIm} + \cancel{sIm} + \cancel{ReIm} + Im^2 &= \\ = s^2 + 2sRe + Re^2 + Im^2\end{aligned}$$

$$a_2 = 1 \quad a_1 = 2Re \quad a_0 = Re^2 + Im^2$$

příklad

pro $a_0 \cong 100$

RS dvoukapacitní

ad 1) $100 = t_1 t_2 \rightarrow 5 \quad 20$

ad 2) $100 = t^2 \rightarrow t = 10$

ad 3) $100 = Re^2 + Im^2 \rightarrow$ Re Im

a) $7 \quad 7 = 98$

b) $10 \quad 1 = 101$

c) $1 \quad 10 = 101$

odvození

pro $a_0 \cong 100$

RS dvoukapacitní

	a_2	a_1	a_0	
ad 1)	1	25	100	S1
ad 2)	1	20	100	S2
ad 3)	1	14	98	S3
	1	20	101	S4
	1	2	101	S5

RS dvoukapacitní

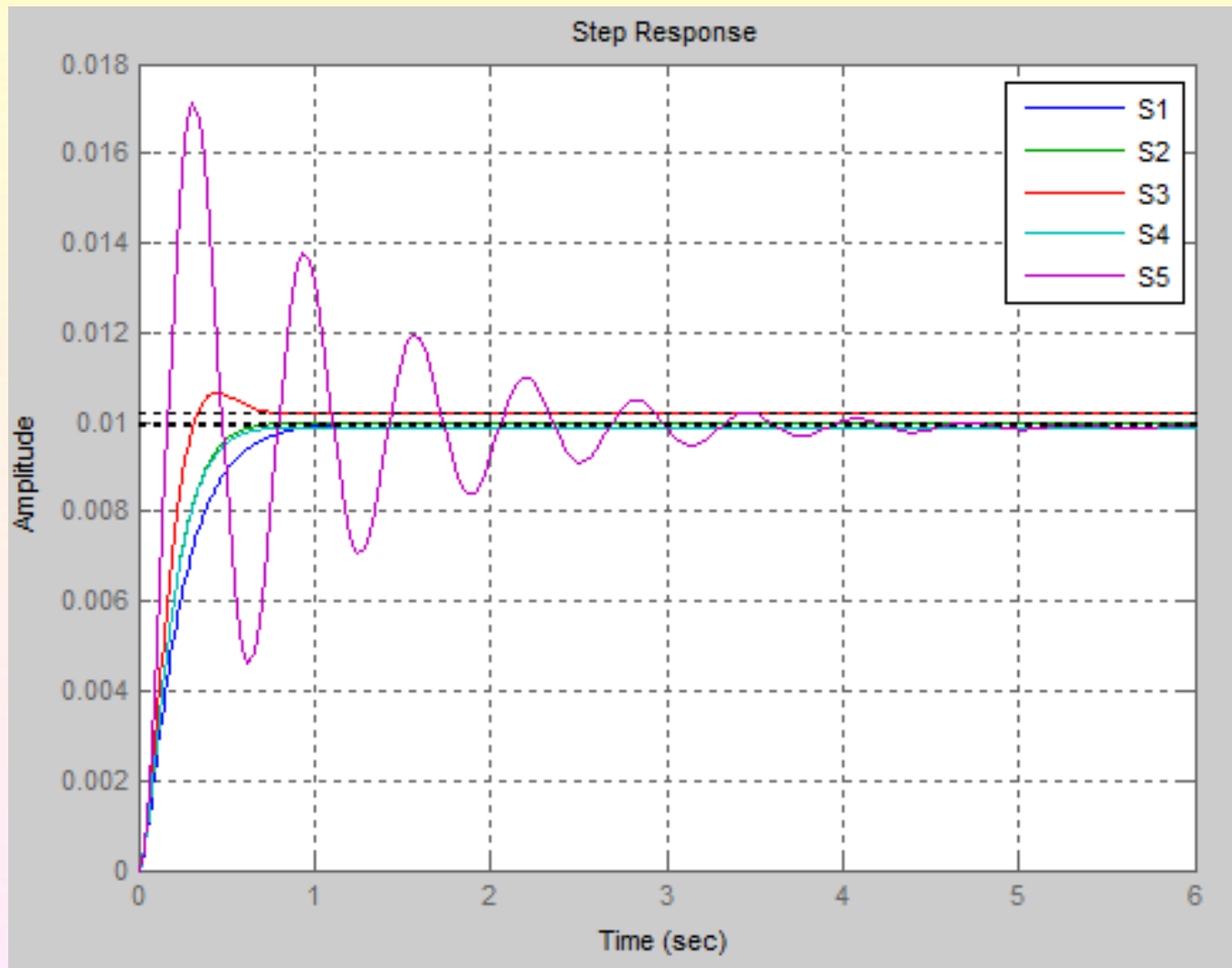
a jdeme to realizovat
v MatLábku a později
i v Simulinku

RS dvoukapacitní

vyřeš to pomocí
FOR_u

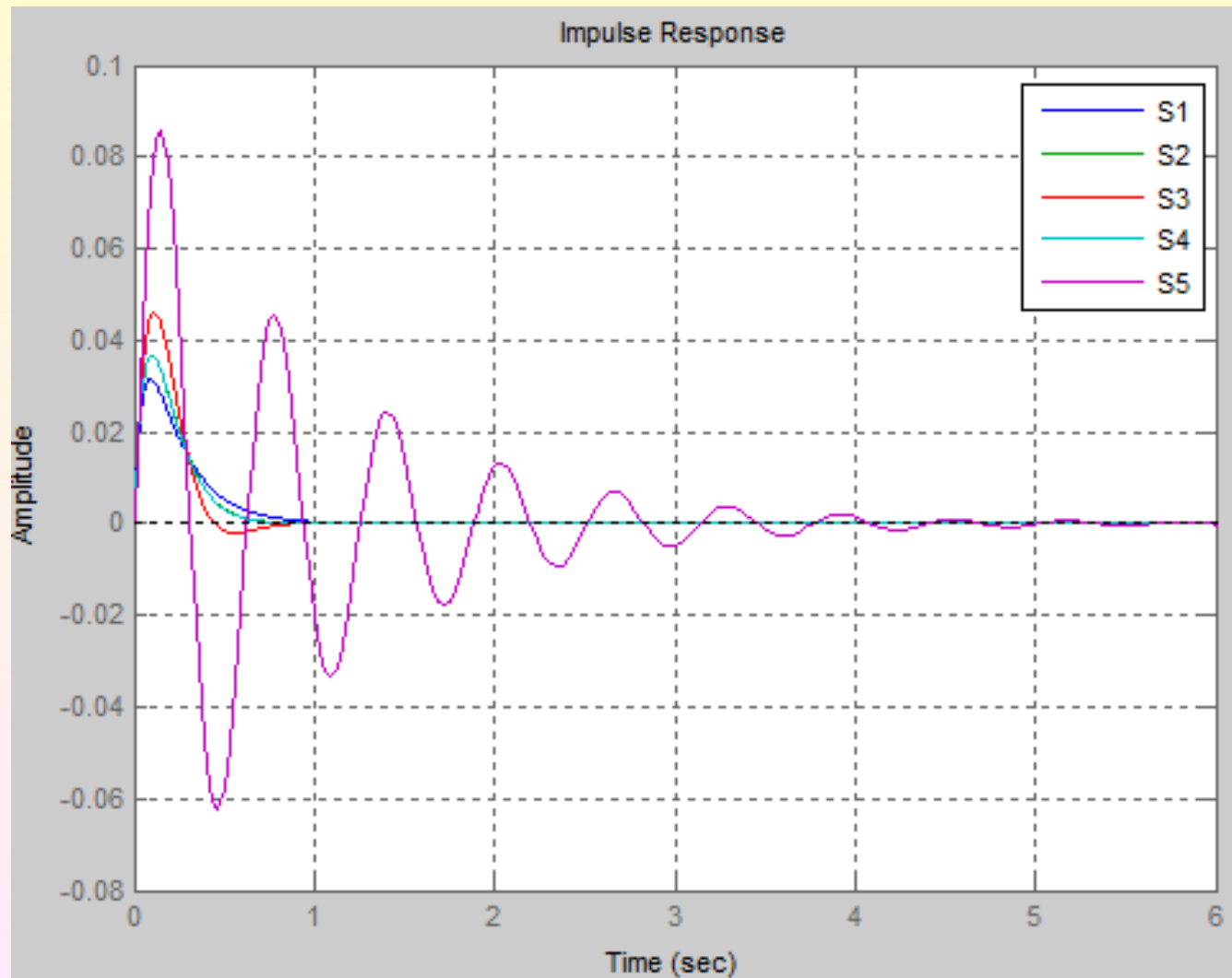
a to jsou výsledky

RS dvoukapacitní



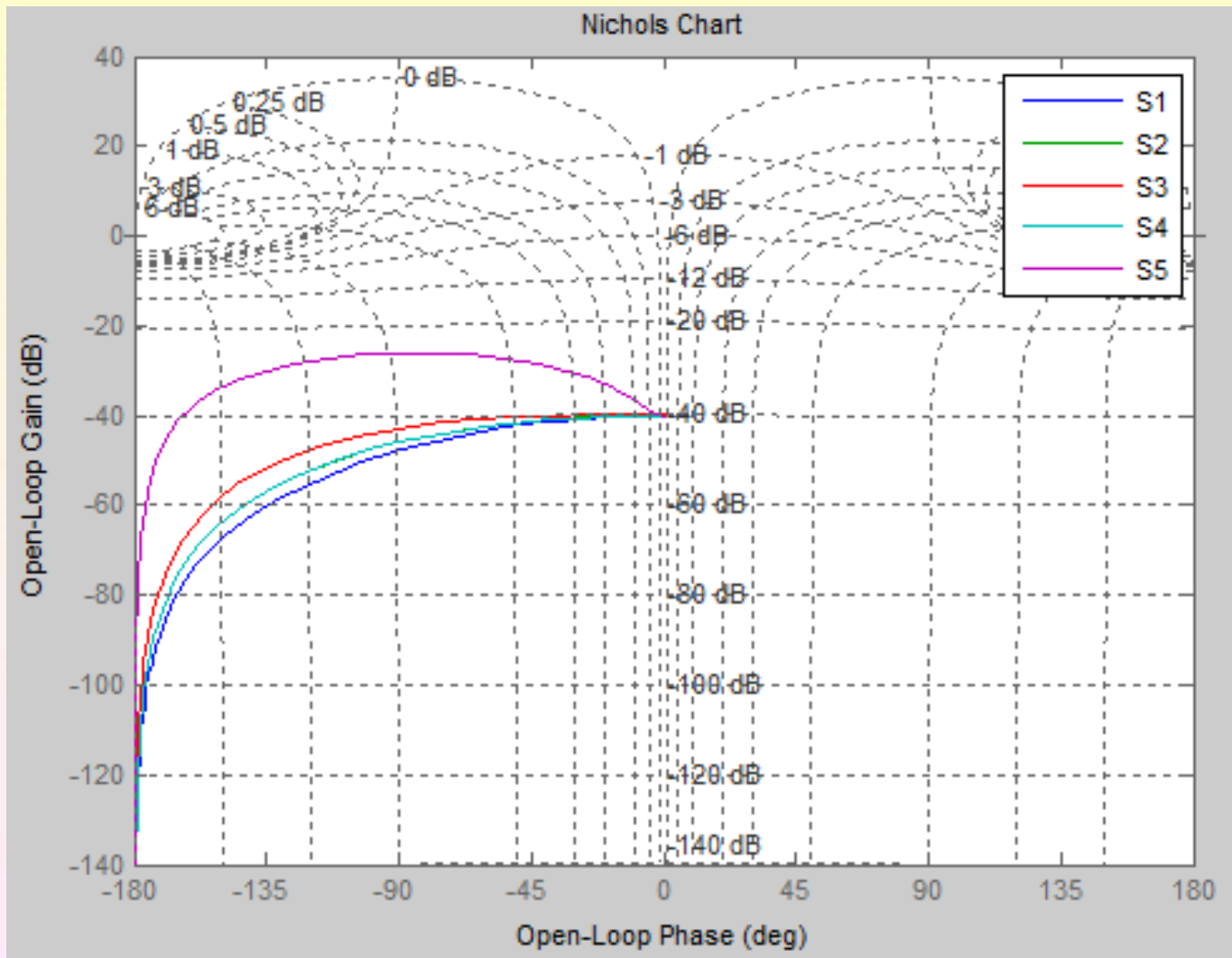
a to jsou výsledky

RS dvoukapacitní



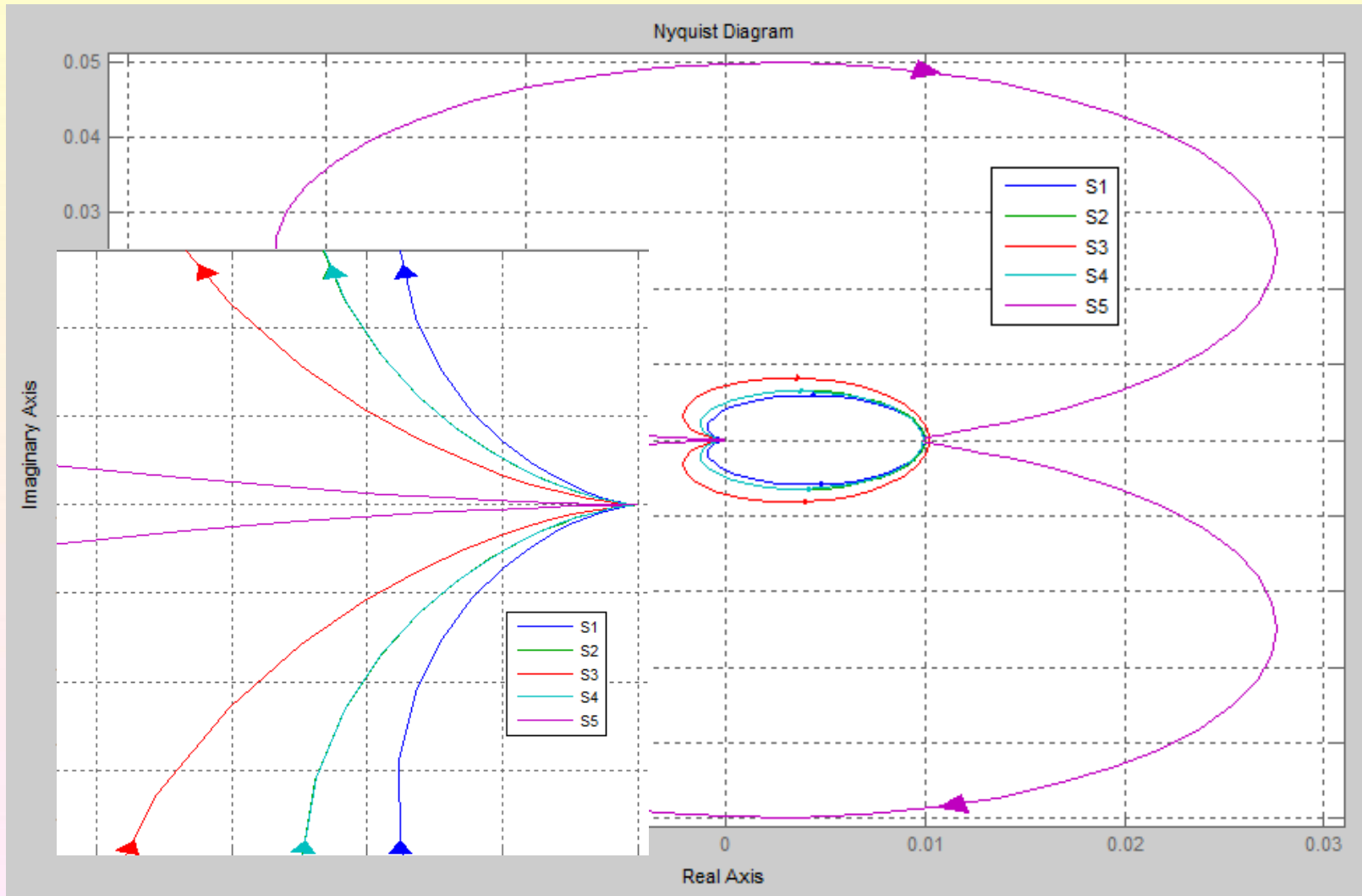
a to jsou výsledky

RS dvoukapacitní



a to jsou výsledky

RS dvoukapacitní



RS dvoukapacitní

!!!zdůvodni,
proč jsou
jenom čtyři
průběhy!!!

uzavřený RO

```
>> Sr=tf([2 .1 2],[1 0]) %regulátor
```

Transfer function:

$$2s^2 + 0.1s + 2$$

s

Napiš DR regulátoru

uzavřený RO

```
>> Ss=tf([1],[1 2 5 1 1]) %RS
```

Transfer function:

1

$s^4 + 2 s^3 + 5 s^2 + s + 1$

Napiš DR RS a jaký je to typ

uzavřený RO

>> $F_o = S_r * S_s$ %otevřený RO

Transfer function:

$$2 s^2 + 0.1 s + 2$$

$$s^5 + 2 s^4 + 5 s^3 + s^2 + s$$

uzavřený RO

>> $F_j = 1 + F_o$ %jmenovatel

Transfer function:

$$s^5 + 2 s^4 + 5 s^3 + 3 s^2 + 1.1 s + 2$$

$$s^5 + 2 s^4 + 5 s^3 + s^2 + s$$

uzavřený RO

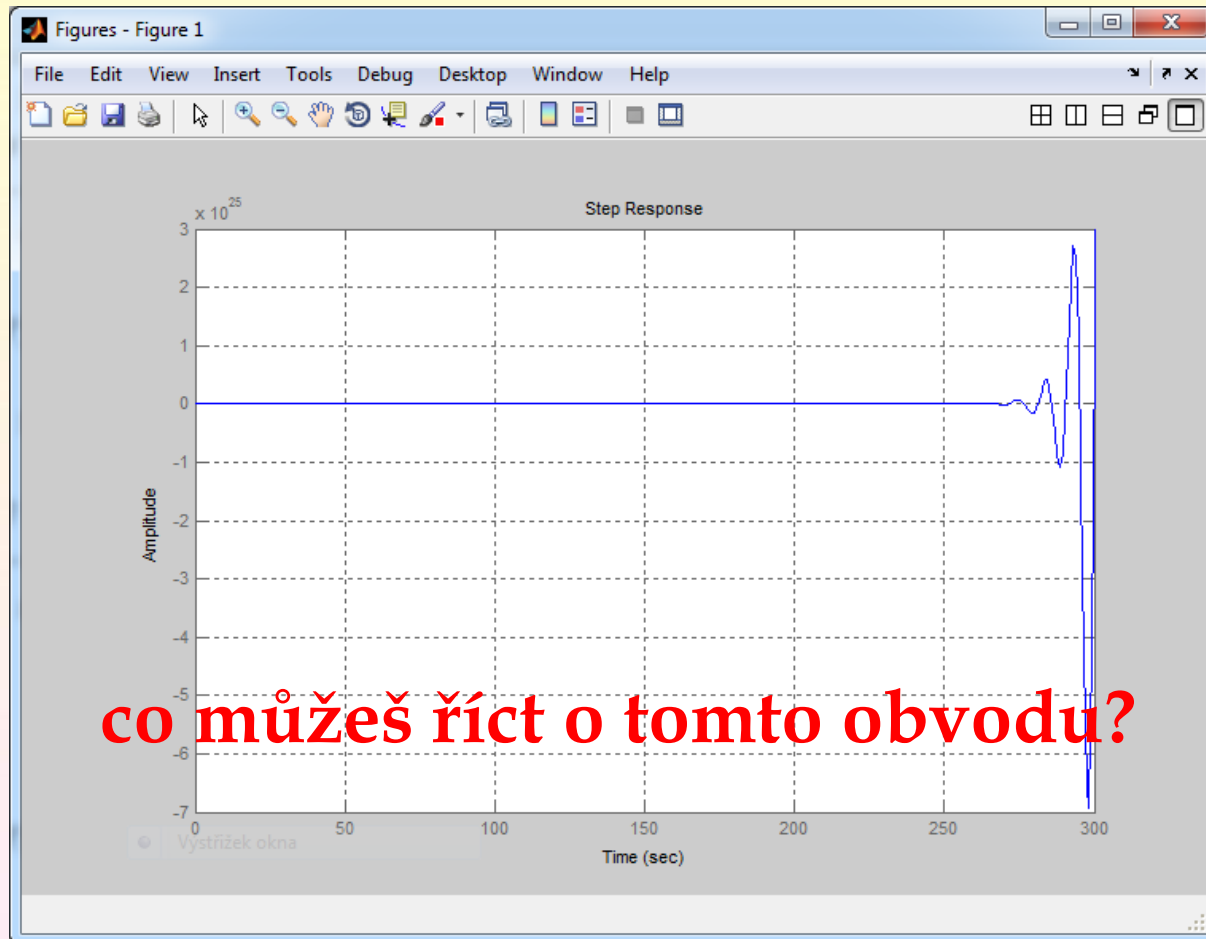
>> $F = F_o / F_j$ %celkový přenos

Transfer function:

$$2 s^7 + 4.1 s^6 + 12.2 s^5 + 6.5 s^4 + 12.1 s^3 + 2.1 s^2 + 2 s$$

$$s^{10} + 4 s^9 + 14 s^8 + 24 s^7 + 35.1 s^6 + 26.2 s^5 + 17.5 s^4 + 14.1 s^3 + 3.1 s^2 + 2 s$$

uzavřený RO



uzavřený RO

```
Ss=tf([1],[1 2 5 1 1]);
```

```
Sr=tf([2 2 .1],[1 0]);
```

```
Fo=Sr*Ss;
```

```
Fj=1+Fo;
```

```
F=Fo/Fj;
```

```
step(F),grid
```

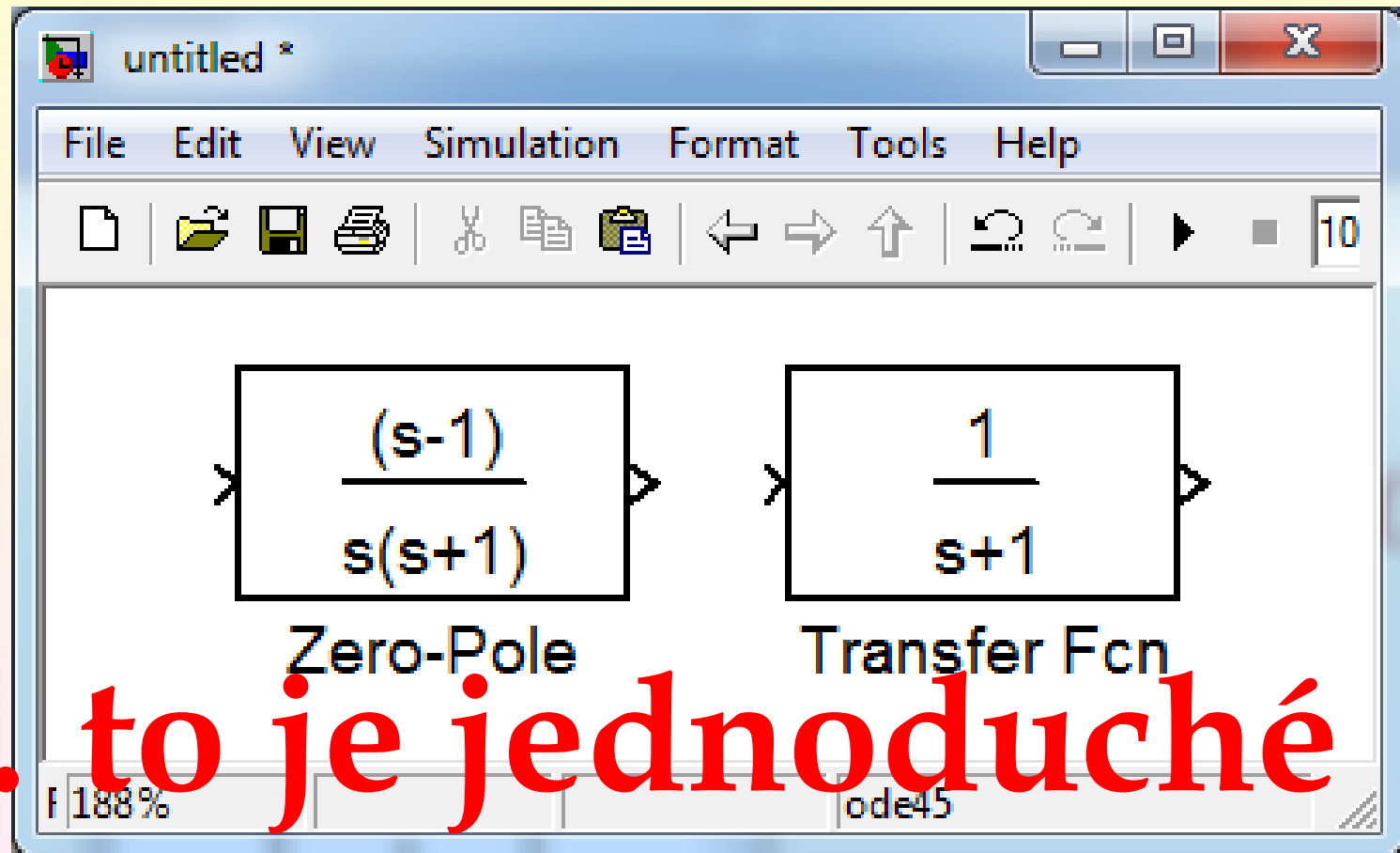
uzavřený RO



co můžeš říct o tomto obvodu?

**... a jak to budeš
řešit
v SIMILINKu?**

uzavřený RO



... to je jednoduché ...

Výsledný
přenos si
můžeme
rozdělit na
parciální
(dílčí) zlomky.
K tomu nám
slouží příkaz -

```
>> [R,P,K]=residue([1],[1 5 3 2])
```

R =

0.0572

-0.0286 - 0.1948i

-0.0286 + 0.1948i

P =

-4.4241

-0.2880 + 0.6076i

-0.2880 - 0.6076i

K =

[]

*parciální
zlomky*

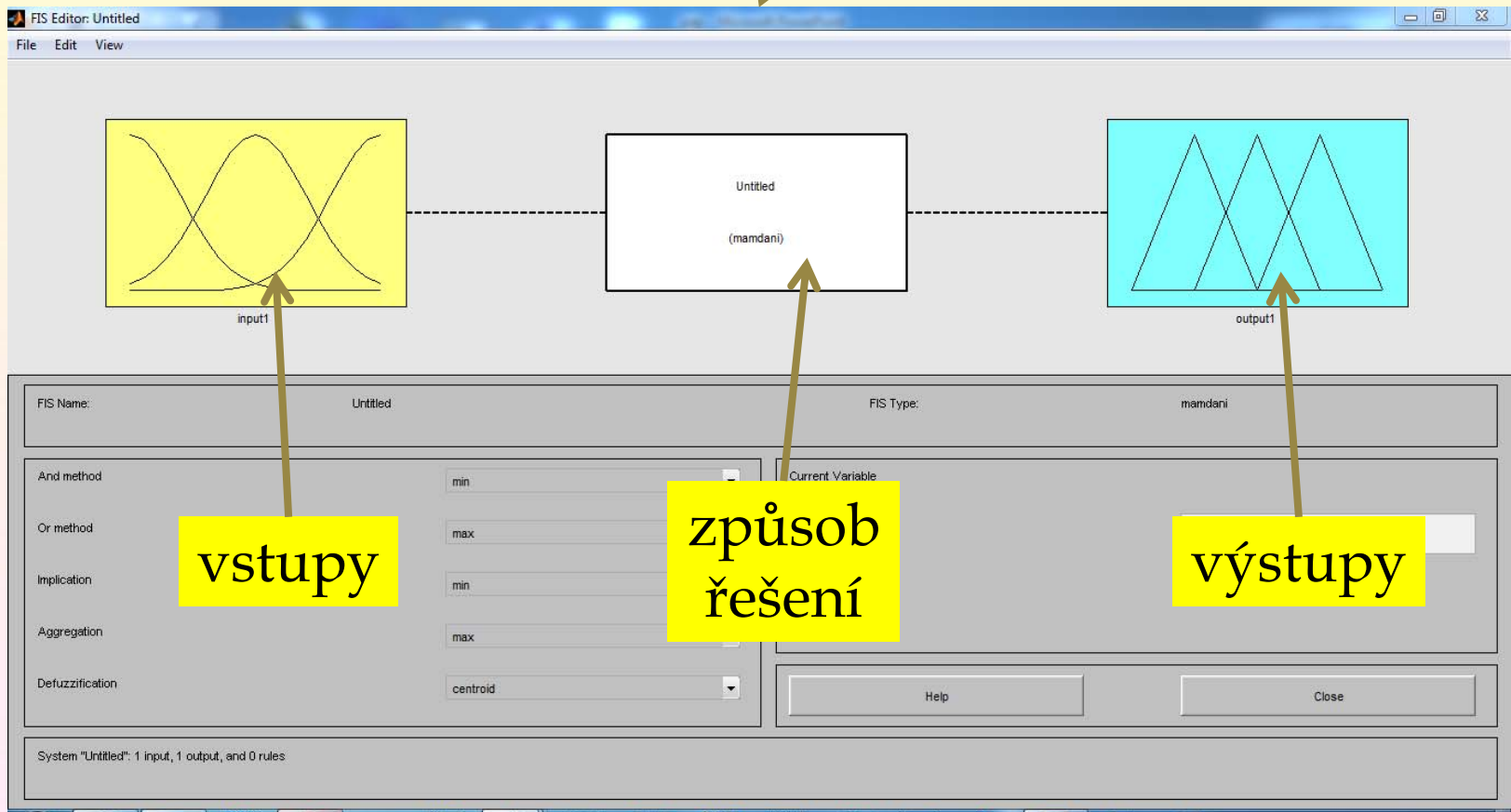
[R,P,K]=residue(č,j)

helpovník

Řídicí technika

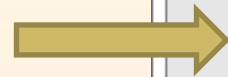
Fuzzy řízení

fuzzy ← po napsání tohoto příkazu
se objeví toto okno

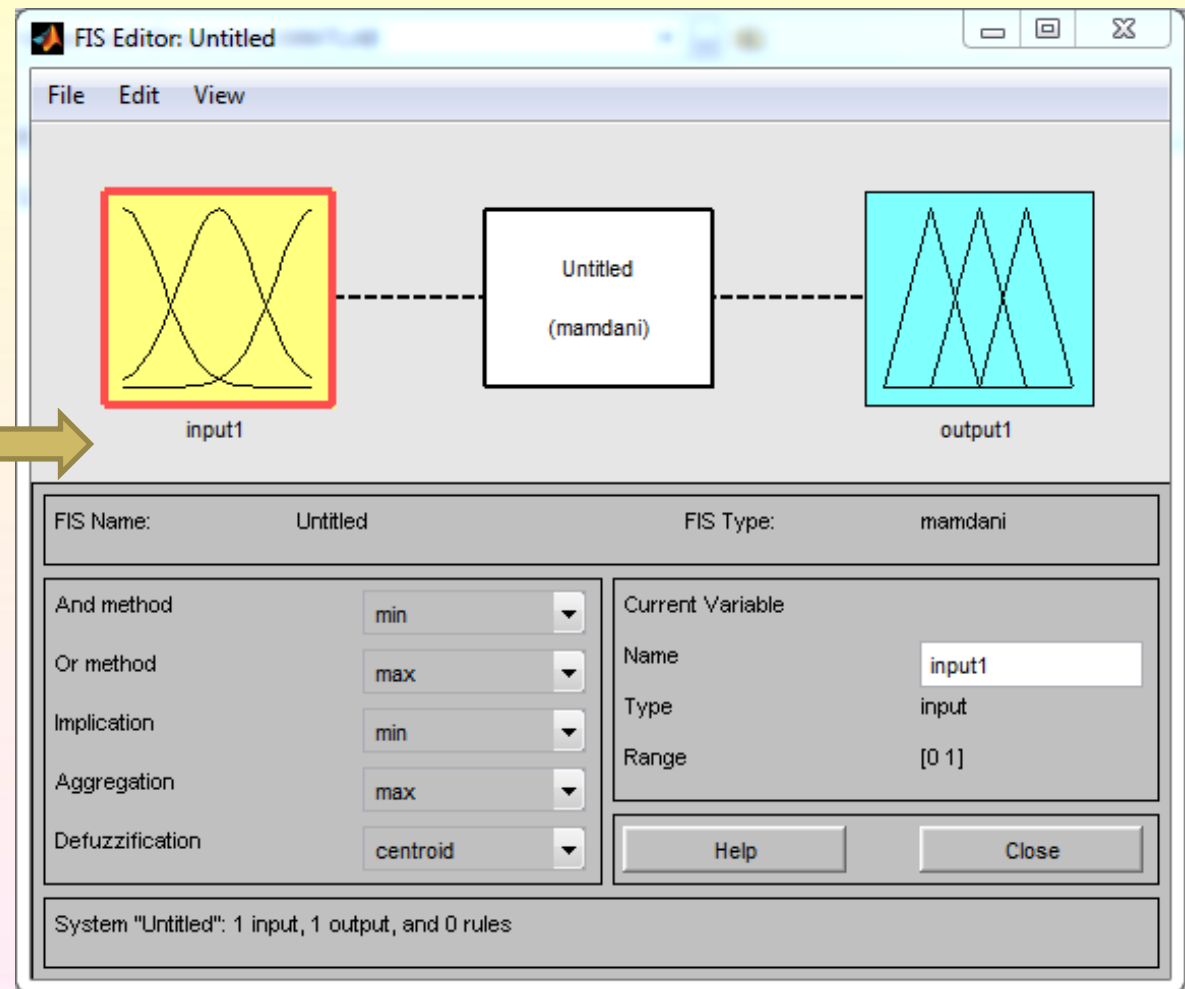


a nyní již
budeme
pokračovat
pouze s ML

Napíšeme
fuzzy a objeví se
toto okno

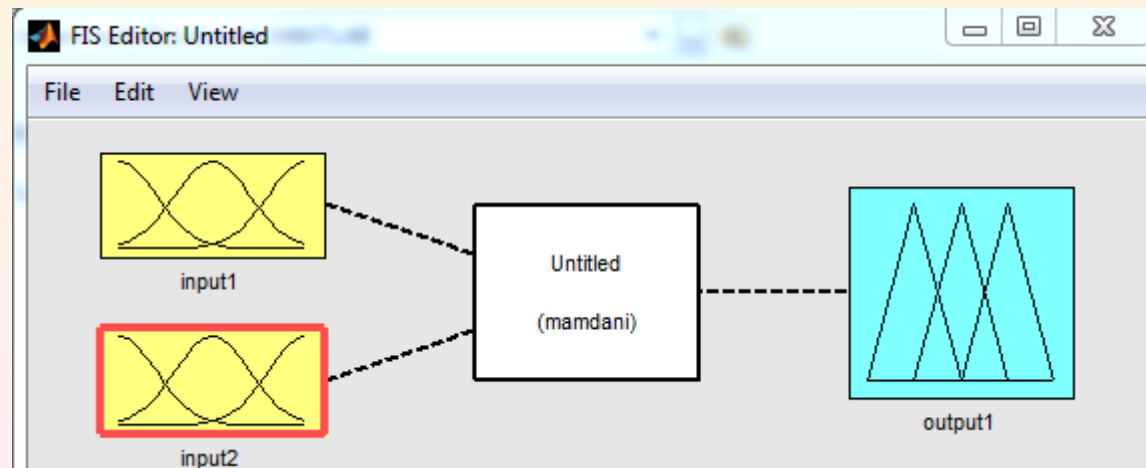


Popsat jednotlivé
části okna



Fuzzy řízení

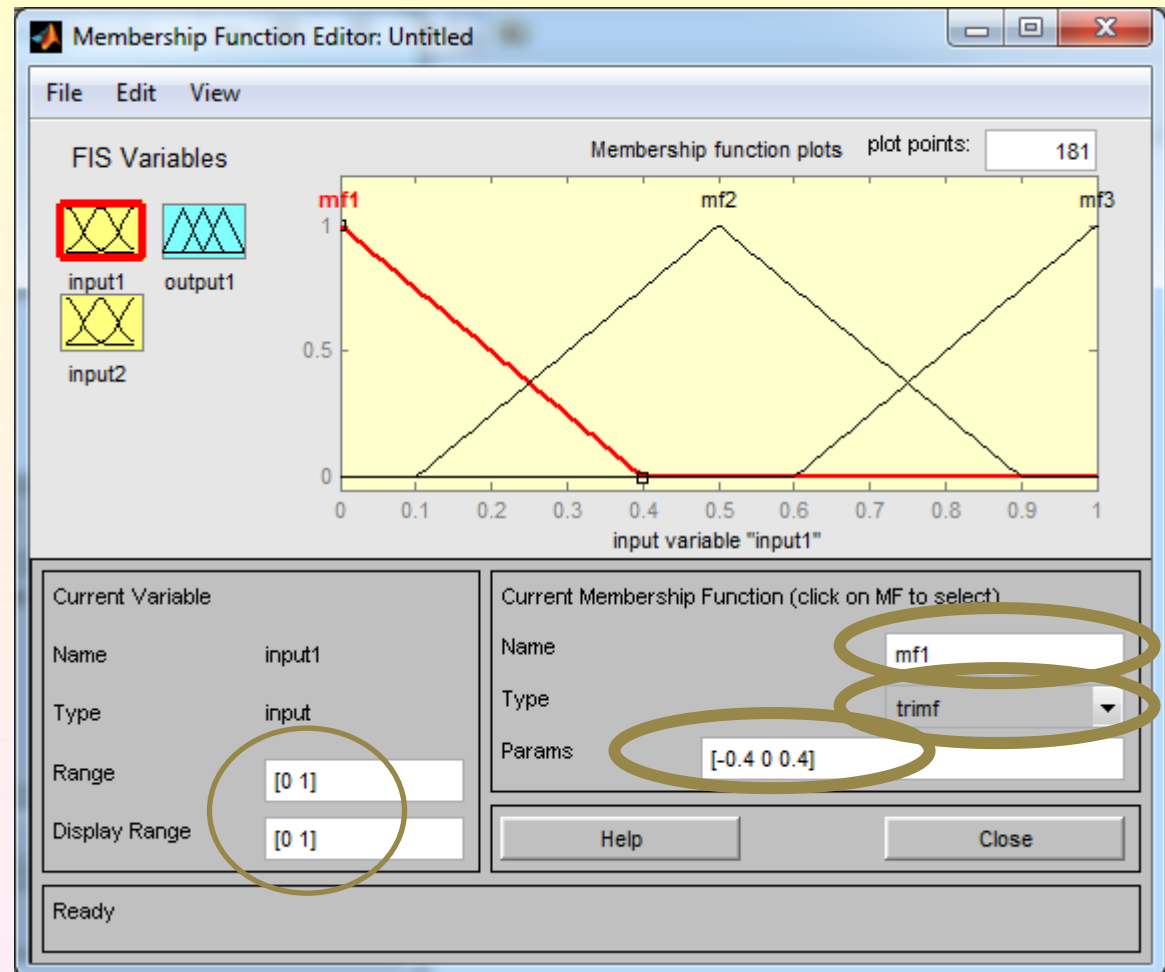
Z HNM vybere file/new FIS/Mamdani nebo Sugeno
Dále zvolíme počet vstupů a výstupů - z HNM vybere
edit/add ../I nebo O



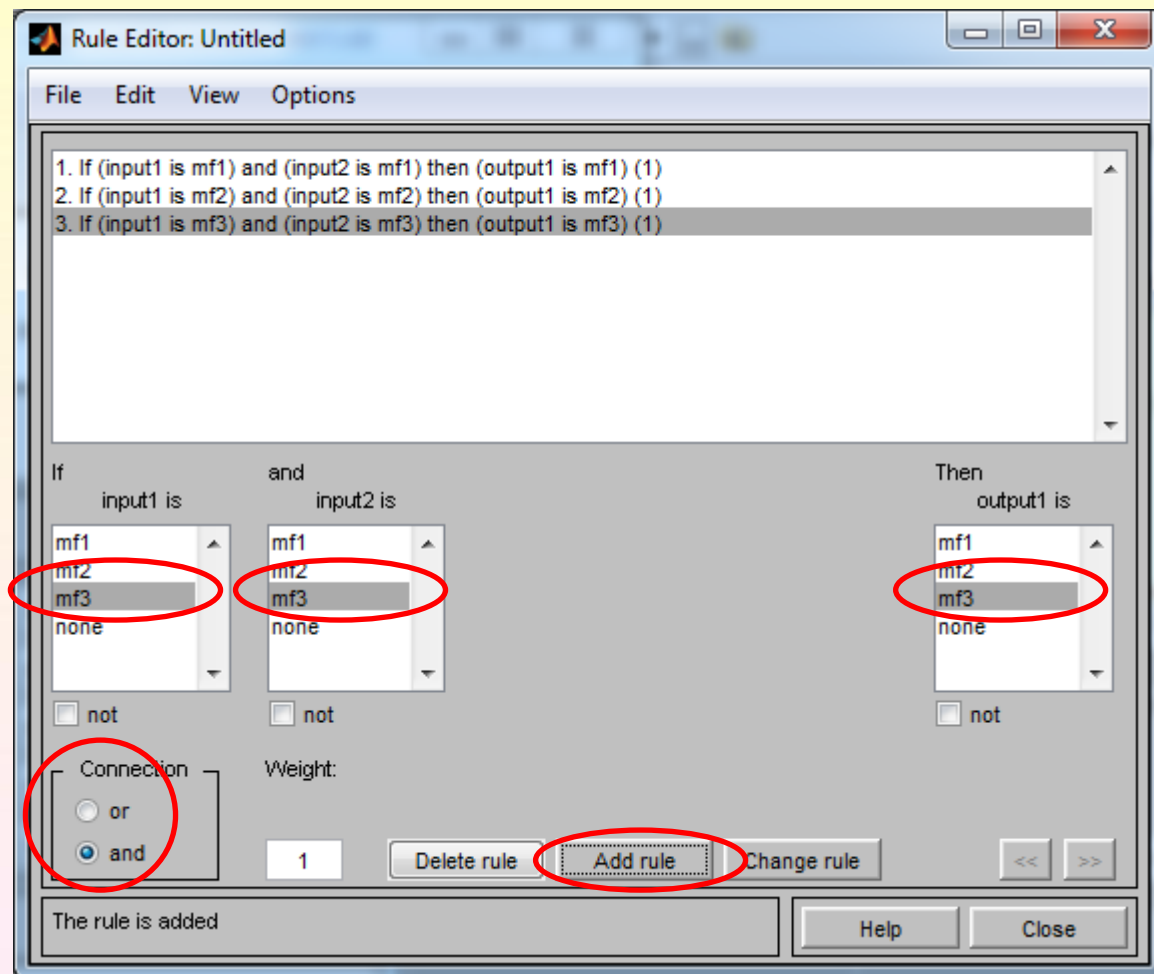
Pojmenování I/O

FIS Name: Untitled		FIS Type: mamdani	
And method	min	Current Variable	
Or method	max	Name	input2
Implication	min	Type	input
Aggregation	max	Range	[0 1]
Defuzzification	centroid		
		Help	Close
Ready			

Editace I/O - DC
na I/O

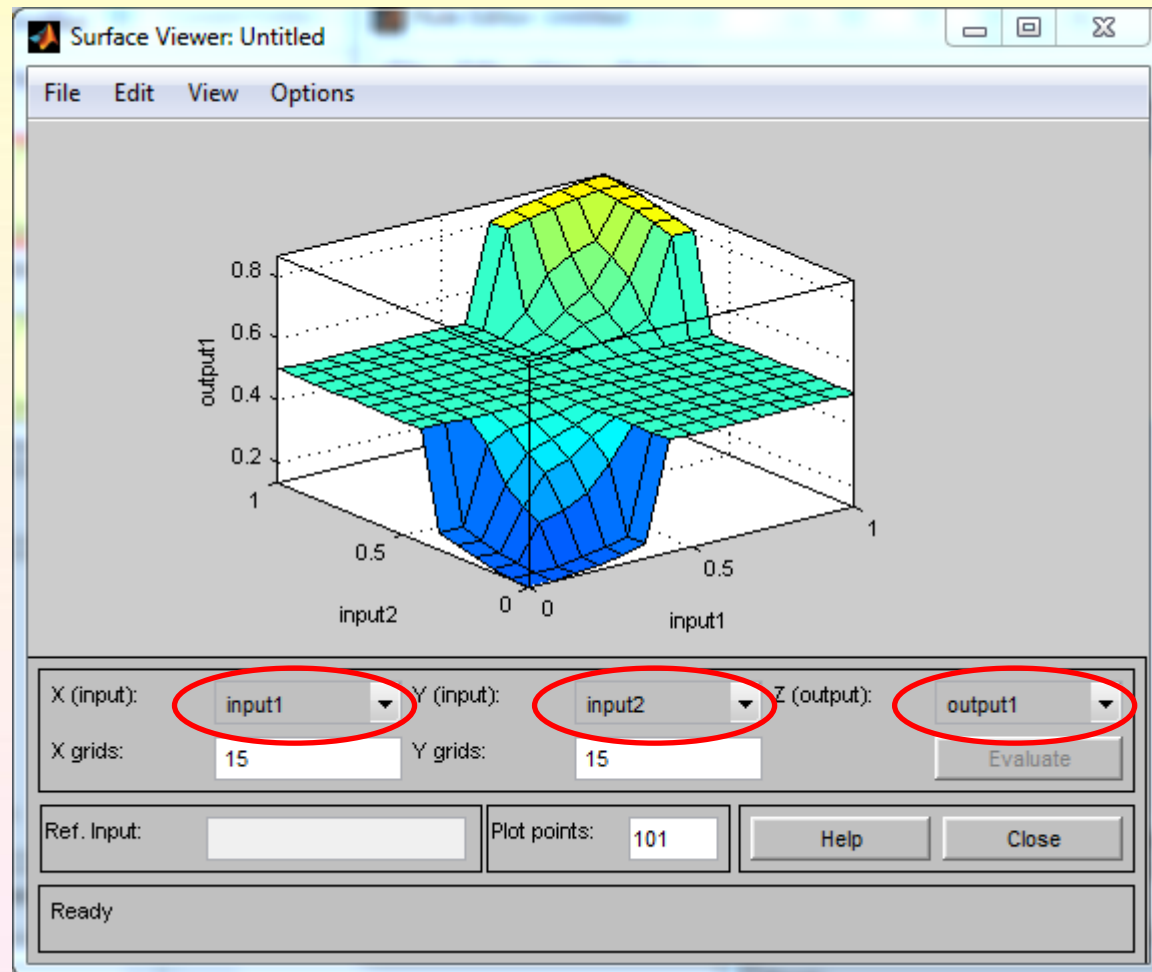


Dále nadefinujeme
pravidla - z HNM
vybere
edit/rules ...



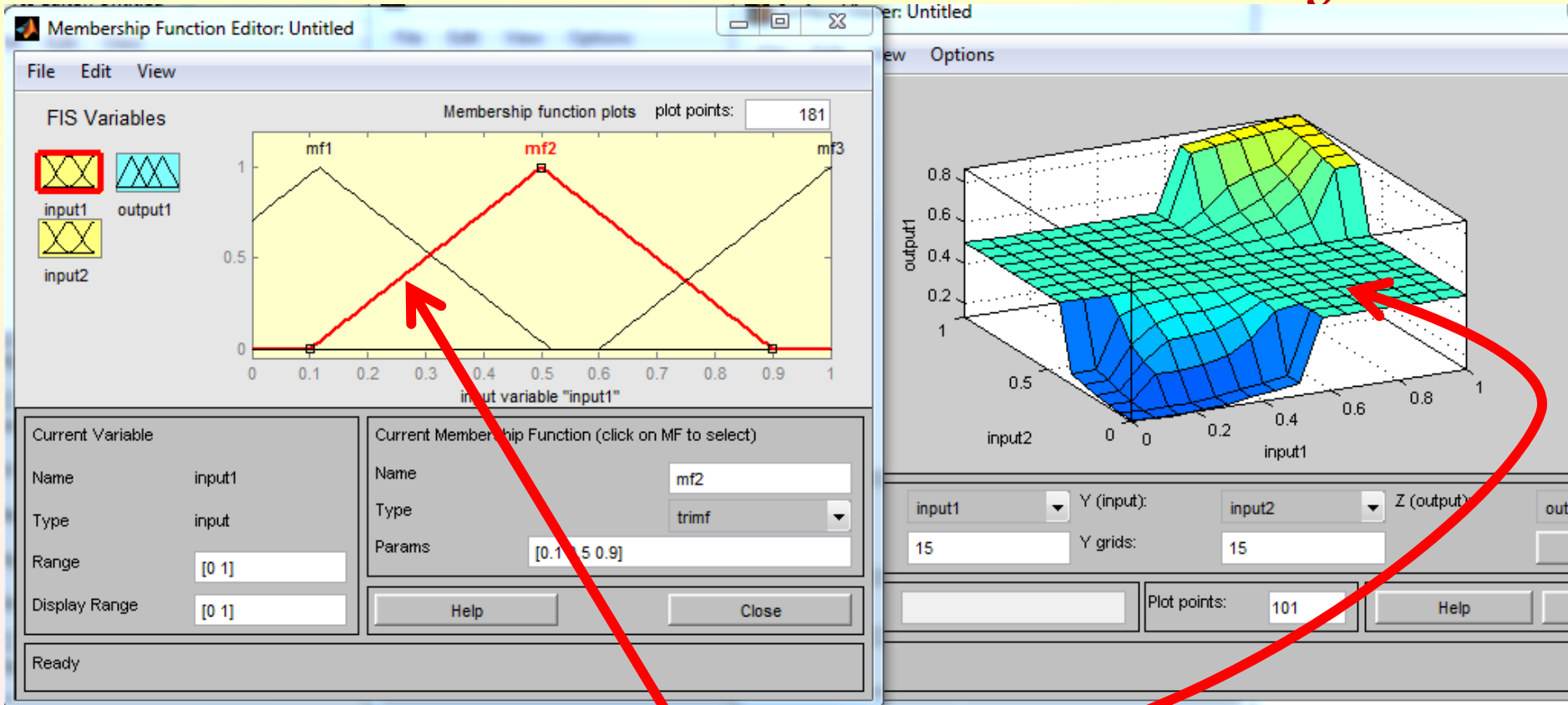
Fuzzy řízení

A už máme **graf**
- z HNM
vybere
view/surface



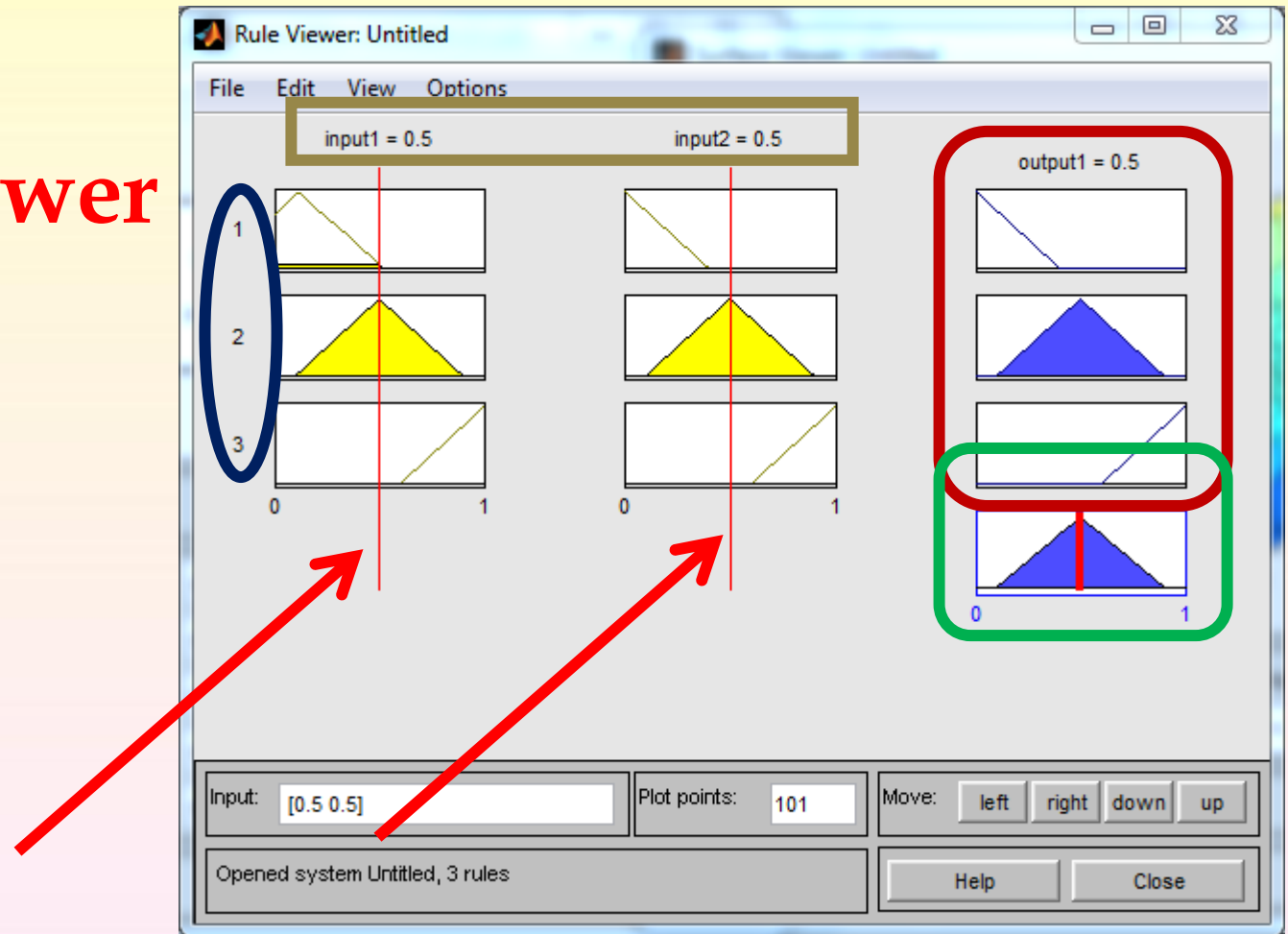
točení s grafem

Fuzzy řízení



změnou polohy průběhu se **okamžitě** mění
k tvar výstupu

v okně
surface viewer
z HNM
vybereme
view/rules



Fuzzy řízení

Neuronové sítě

Postup je následovný

1. nadefinování **vstupů** a **výstupů**
2. určení **typu** NN
3. **trénování** sítě
4. **simulace** sítě

Neuronové sítě

```
1. >> p = [-1 -1 2 2;0 5 0 5];  
2. t = [-1 -1 1 1];  
3. net = newff(p,t,3);  
4. net = train(net,p,t);  
5. y = sim(net,p)
```

y = *výsledek*

1.3870	0.5882	1.5002	1.7571
--------	--------	--------	--------

... a v *helpovniku* si to zjistí ...

Neuronové sítě

*Po změně vektoru p
bude jiný i výsledek*

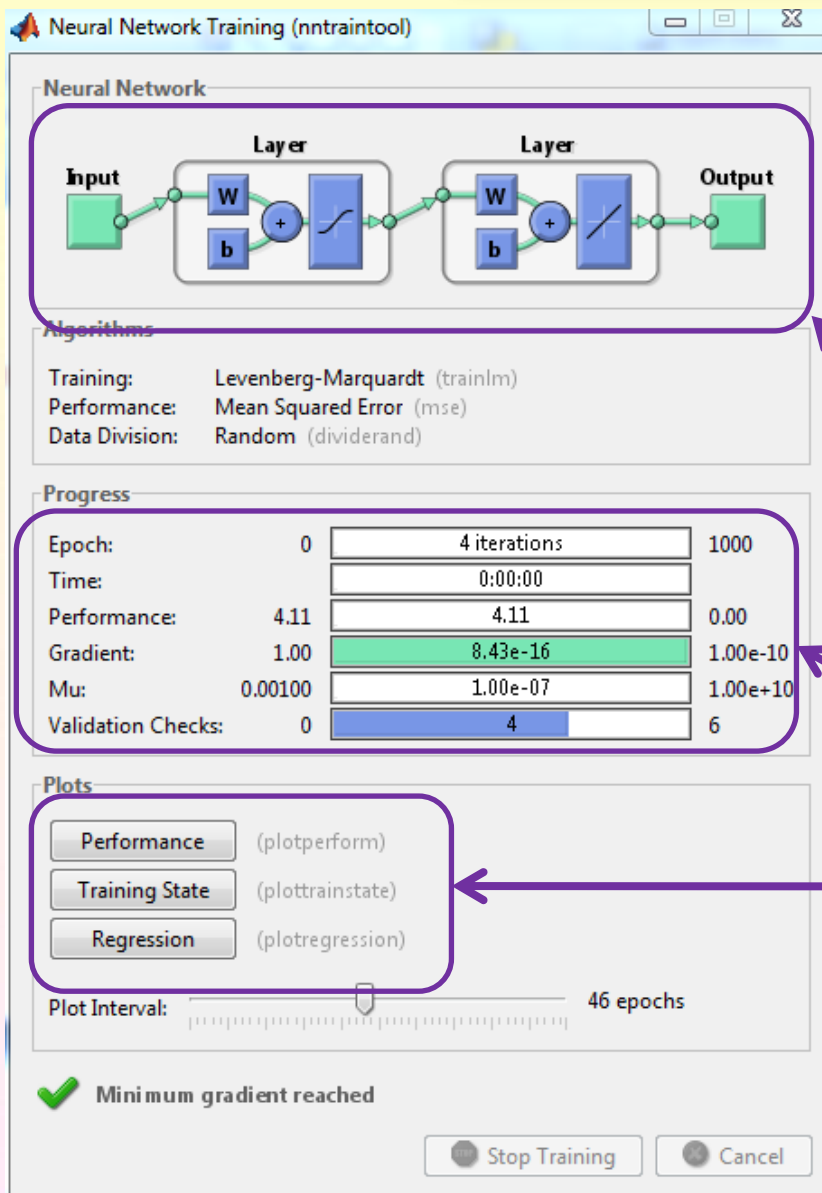
```
>> p = [-1 -1 2 2; 0 5 1 5]; y = sim(net,p)
```

y =

1.3870	0.5882	2.0166	1.7571
1.3870	0.5882	1.5002	1.7571

Řídicí technika

Neuronové sítě

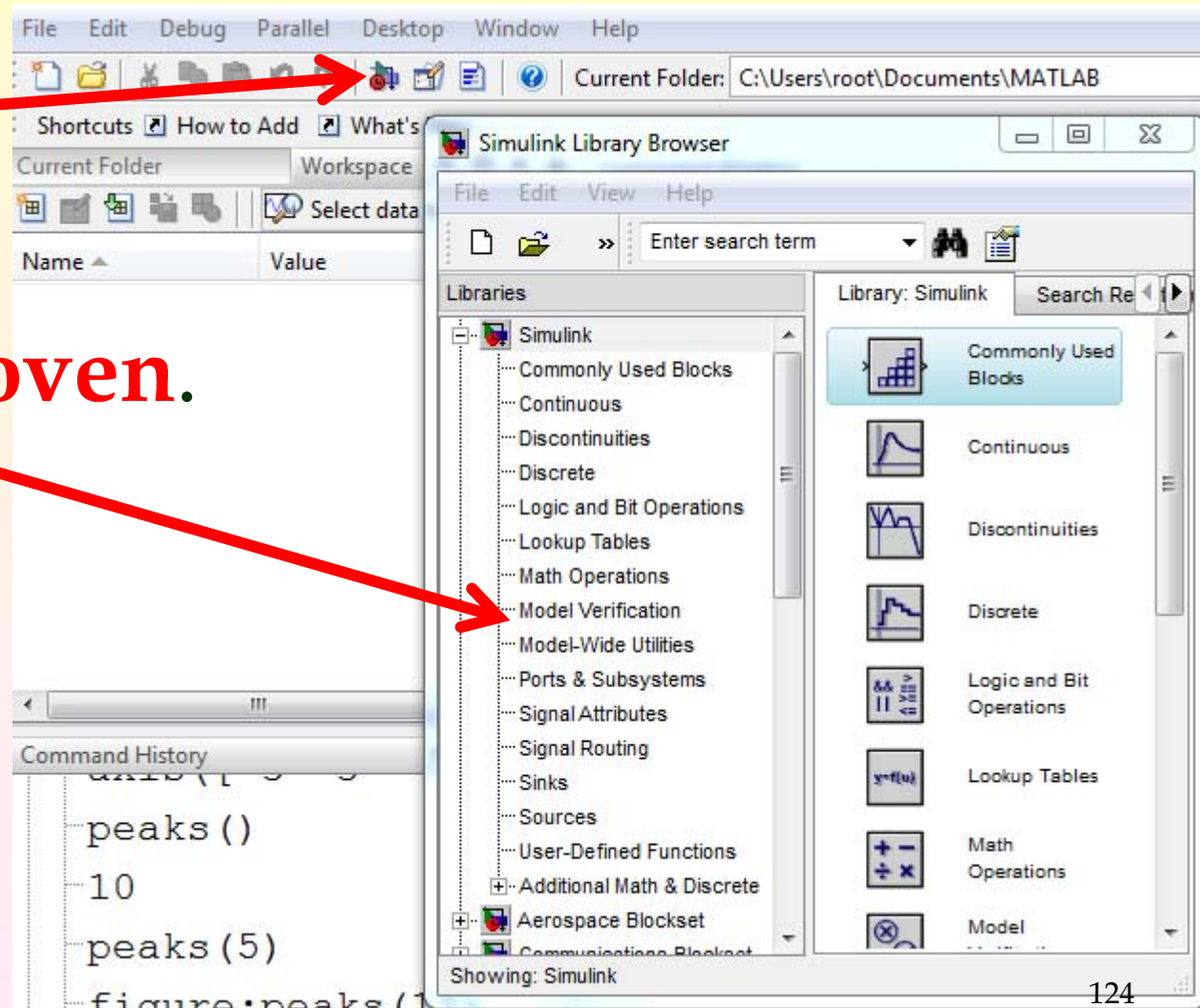


a při použití
příkazu **train** se
nám otevře toto
okno a v něm
můžeme zjistit toto
... **helpovníček** ...

Neuronové sítě

SIMULINK

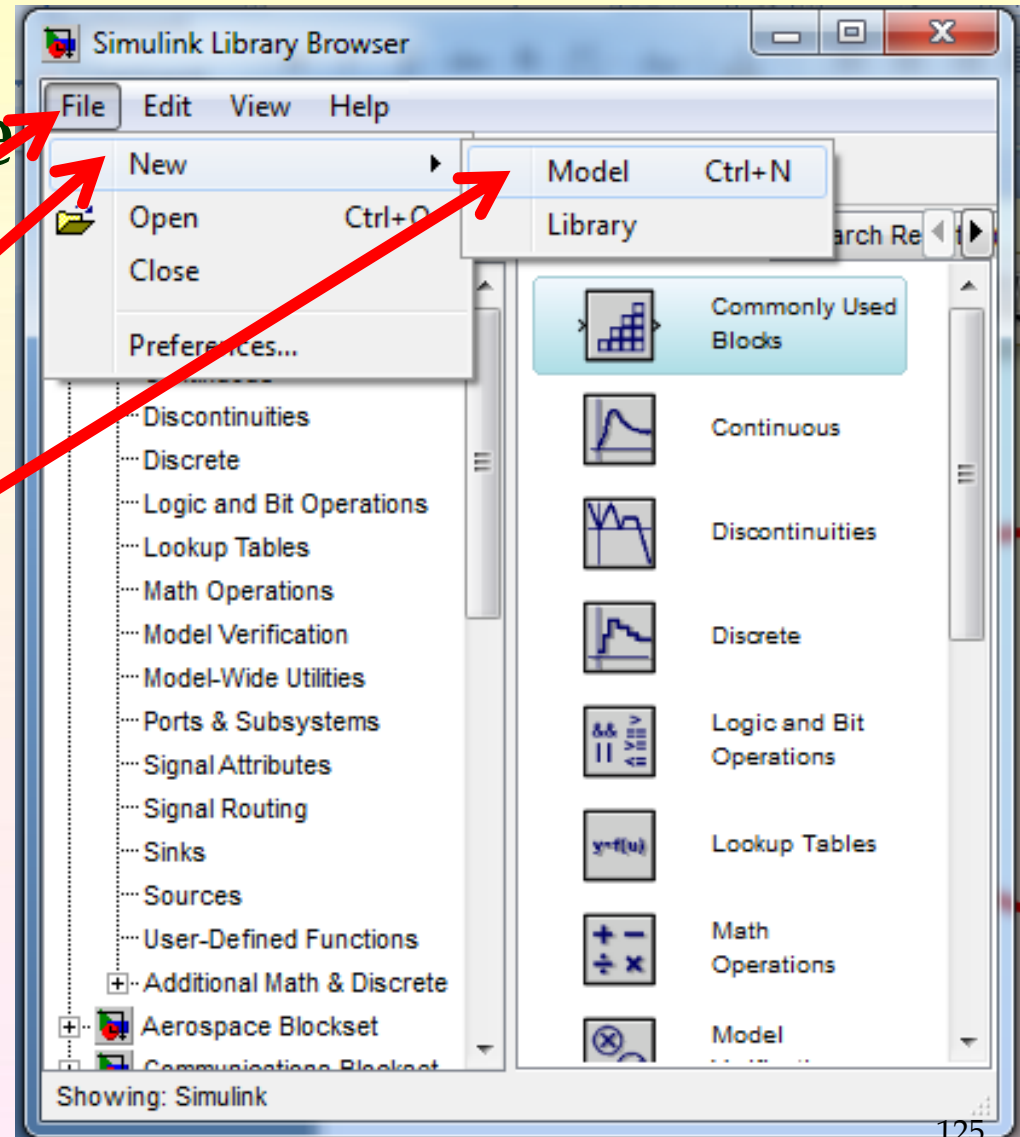
Po spuštění
Simulinku se
nám zobrazí
seznam knihoven.



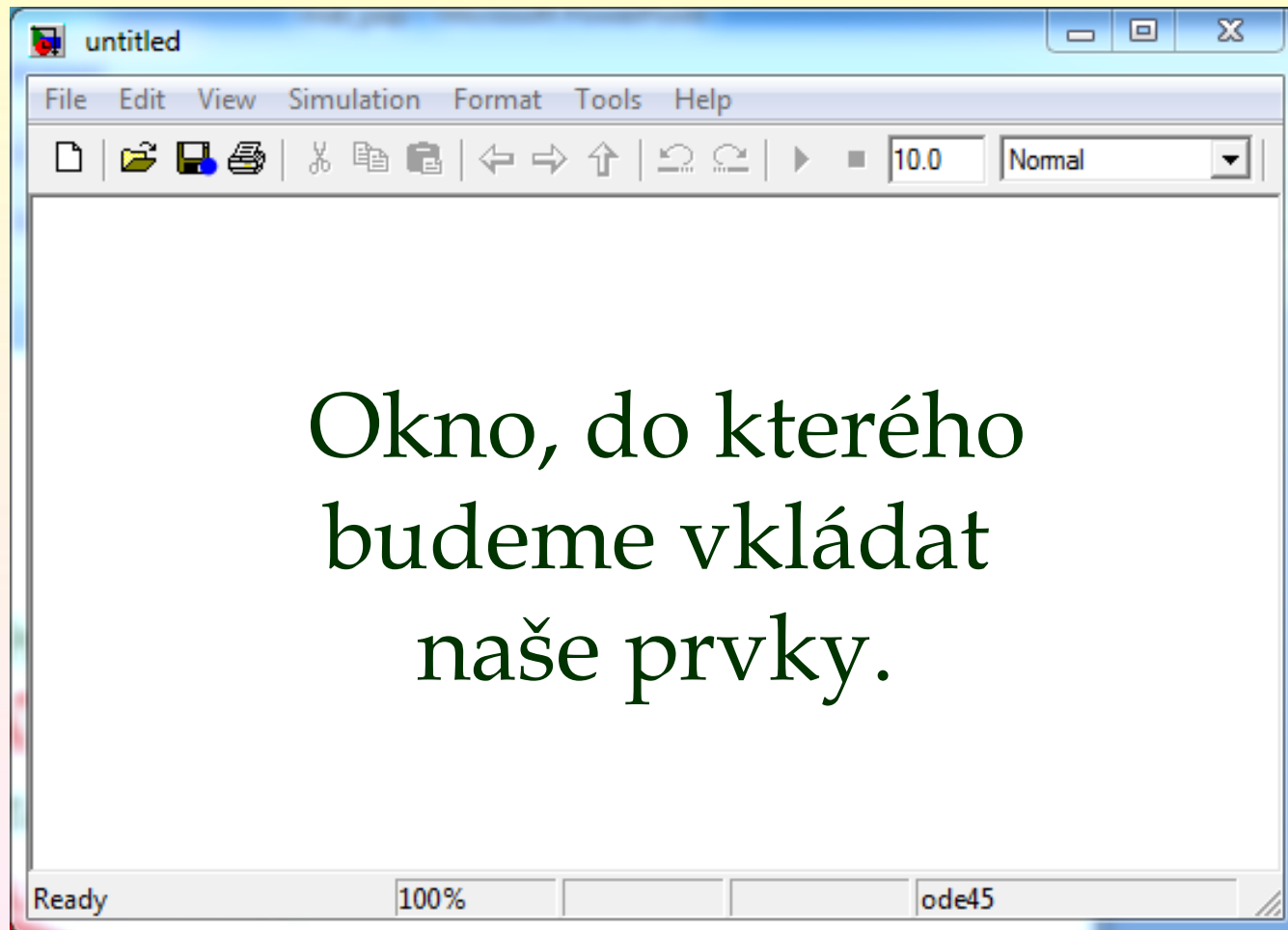
SIMULINK

Nejprve vytvoříme
nový **model**

File
New
Model.

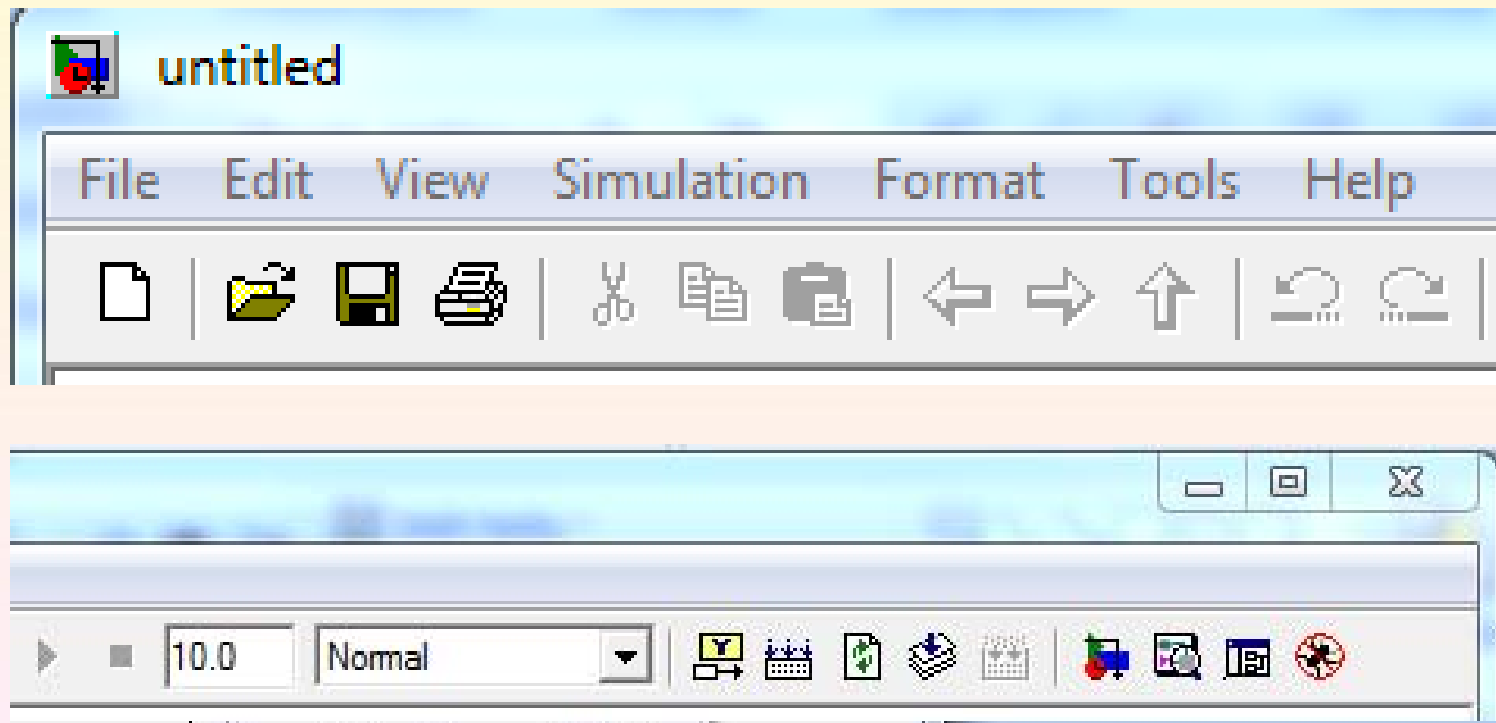


SIMULINK

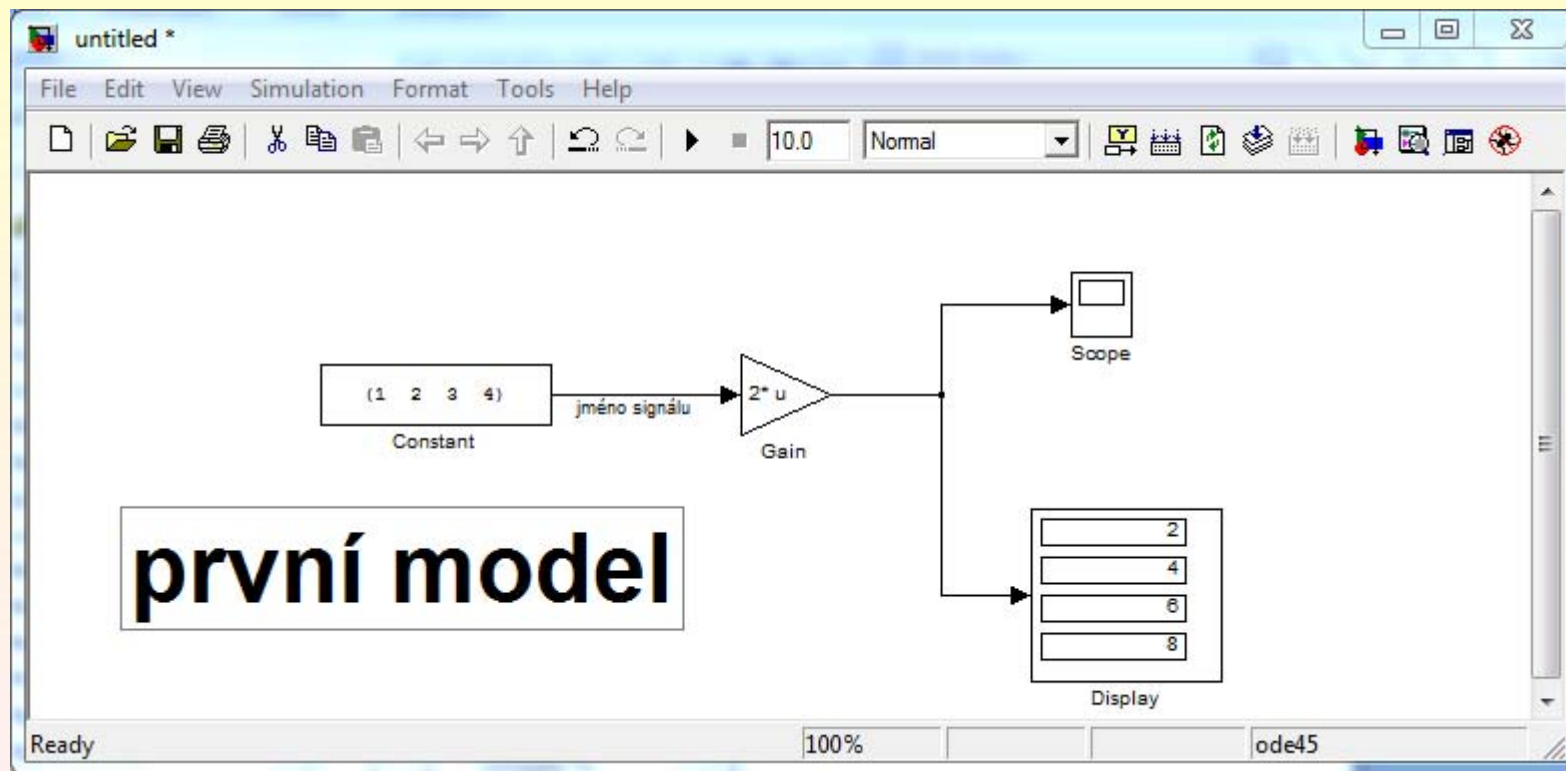


SIMULINK

HNM popíšeme až dle potřeby

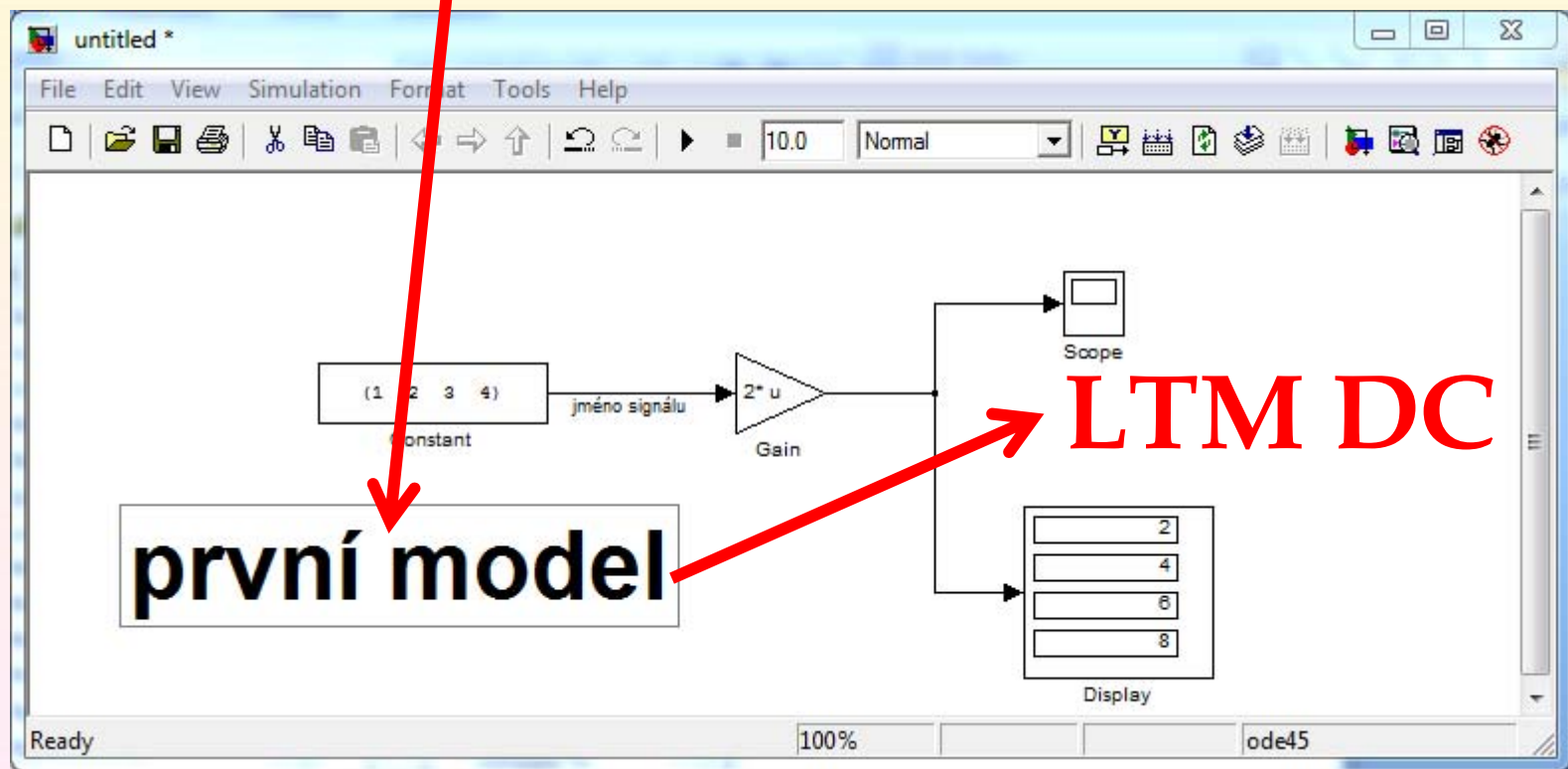


SIMULINK

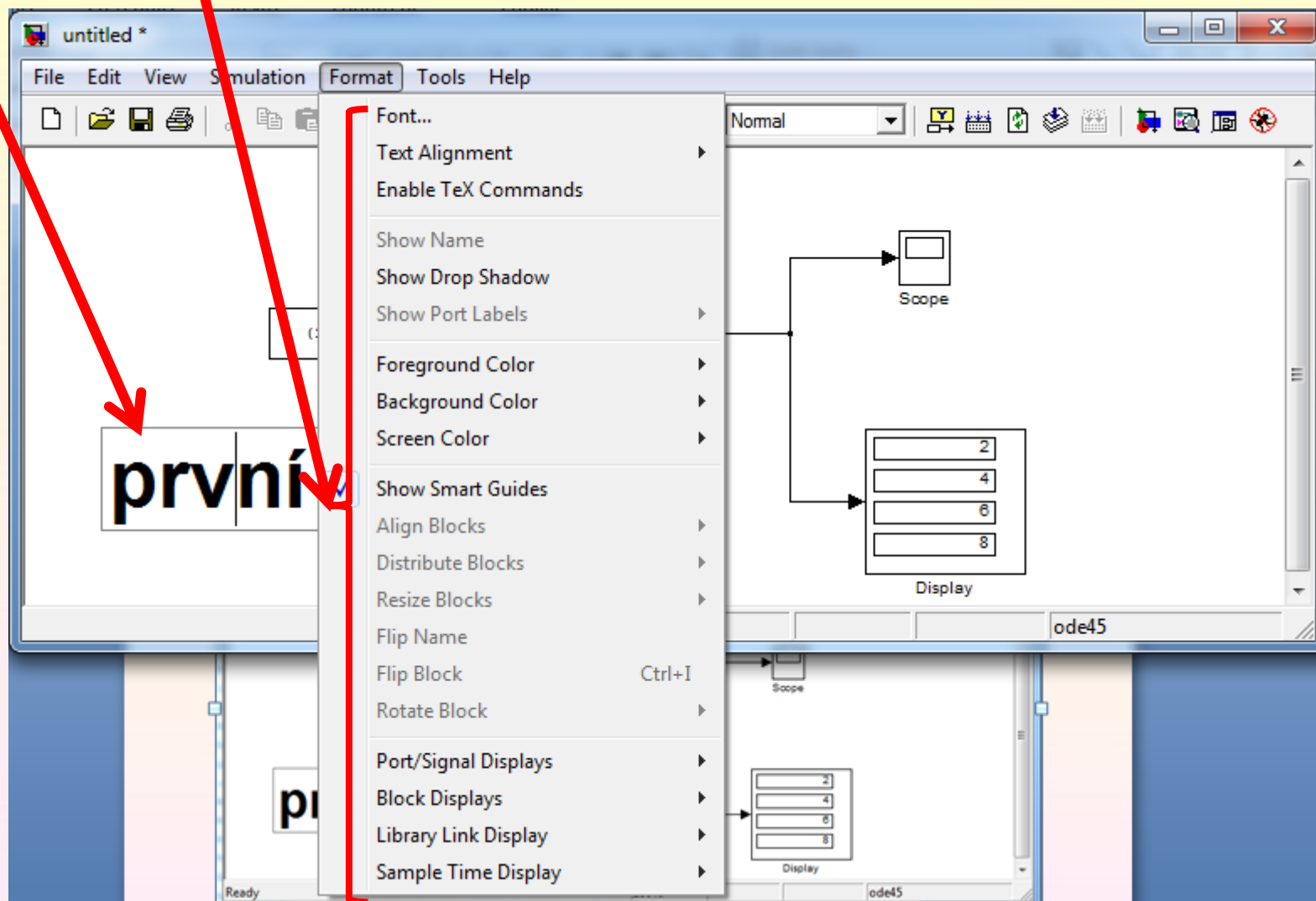


Výsledek našeho prvního modelu

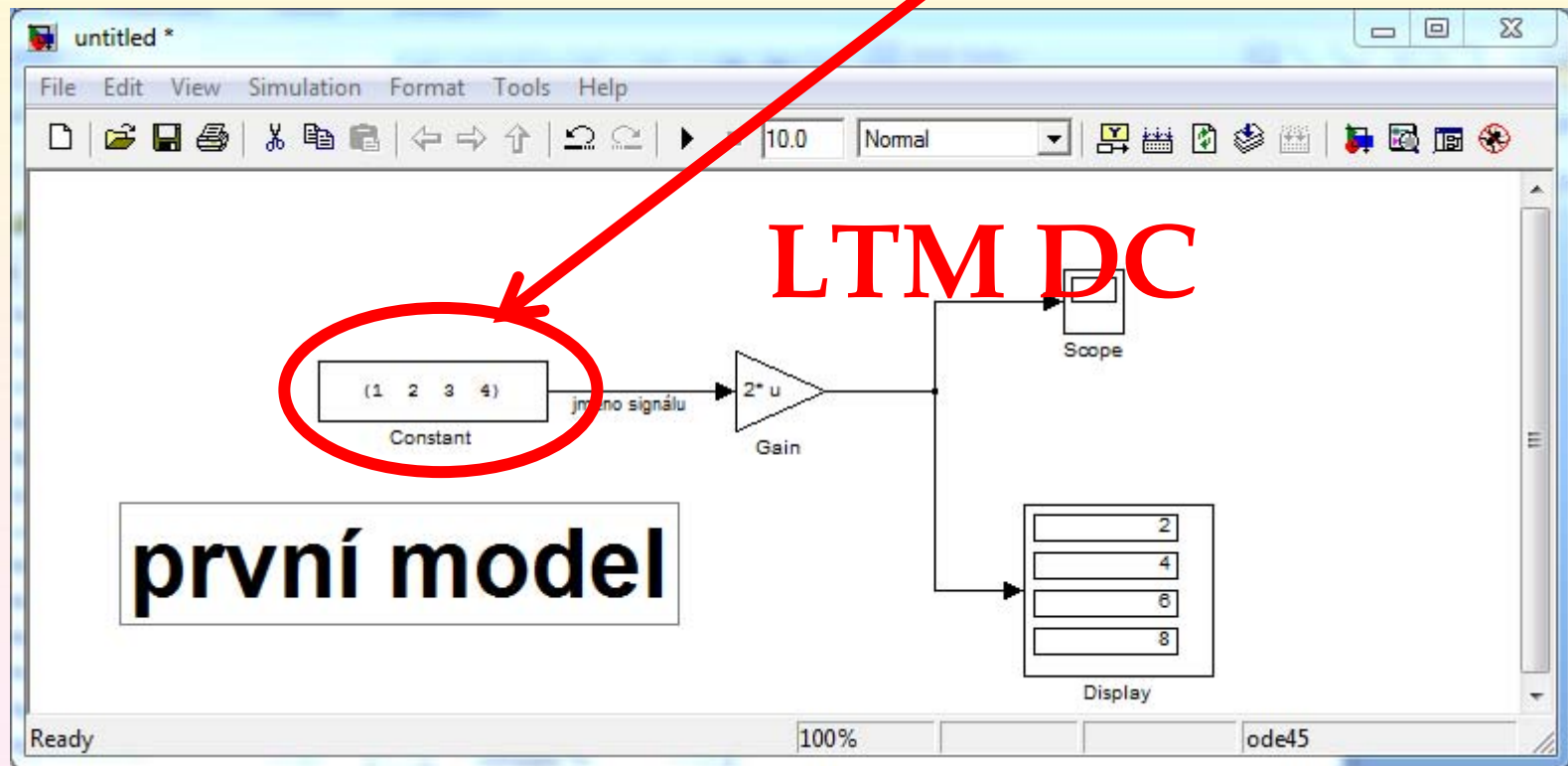
Jak napsat **text** do modelu?



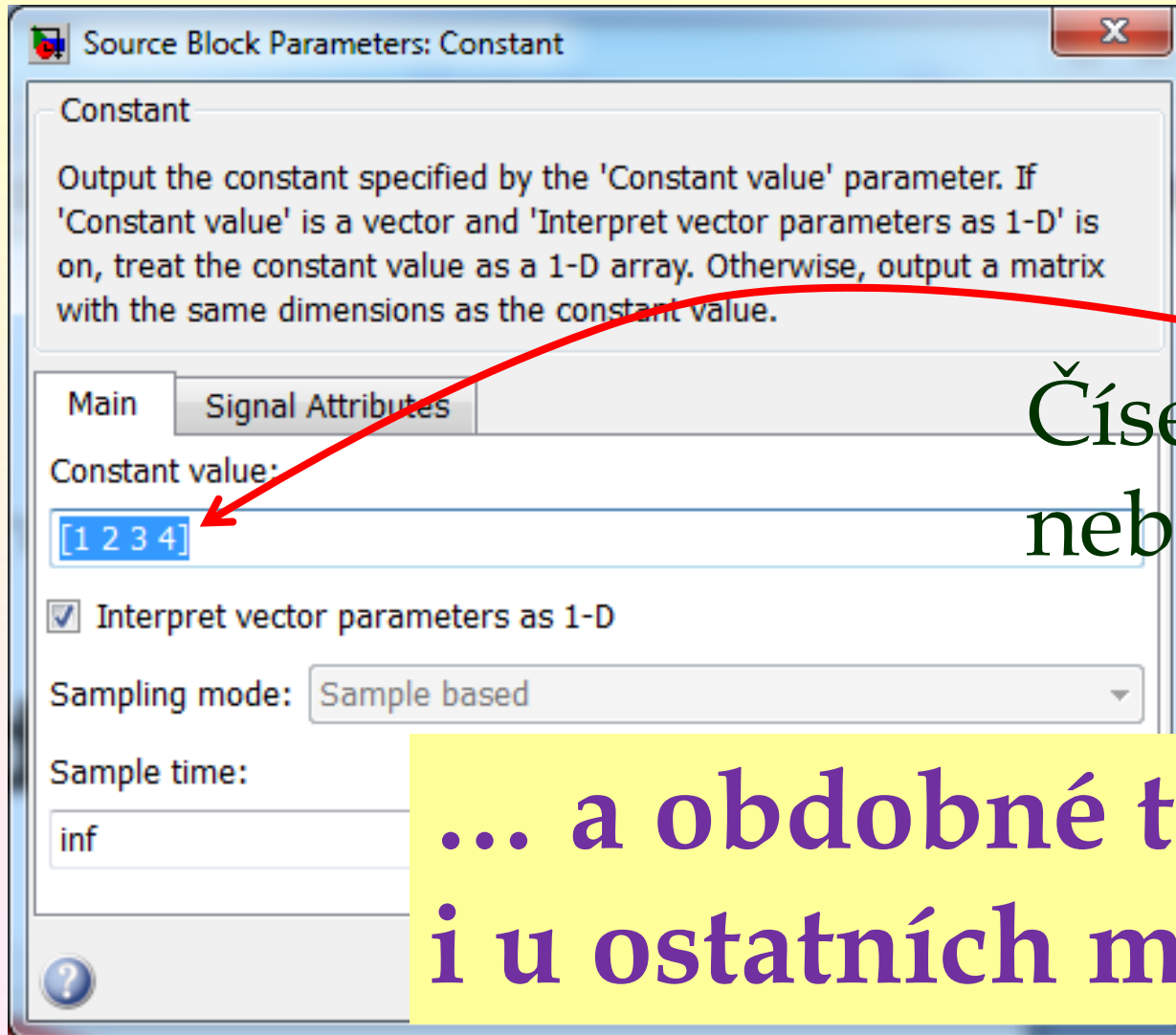
Úprava textu



Vložení hodnoty – u nás matice



SIMULINK



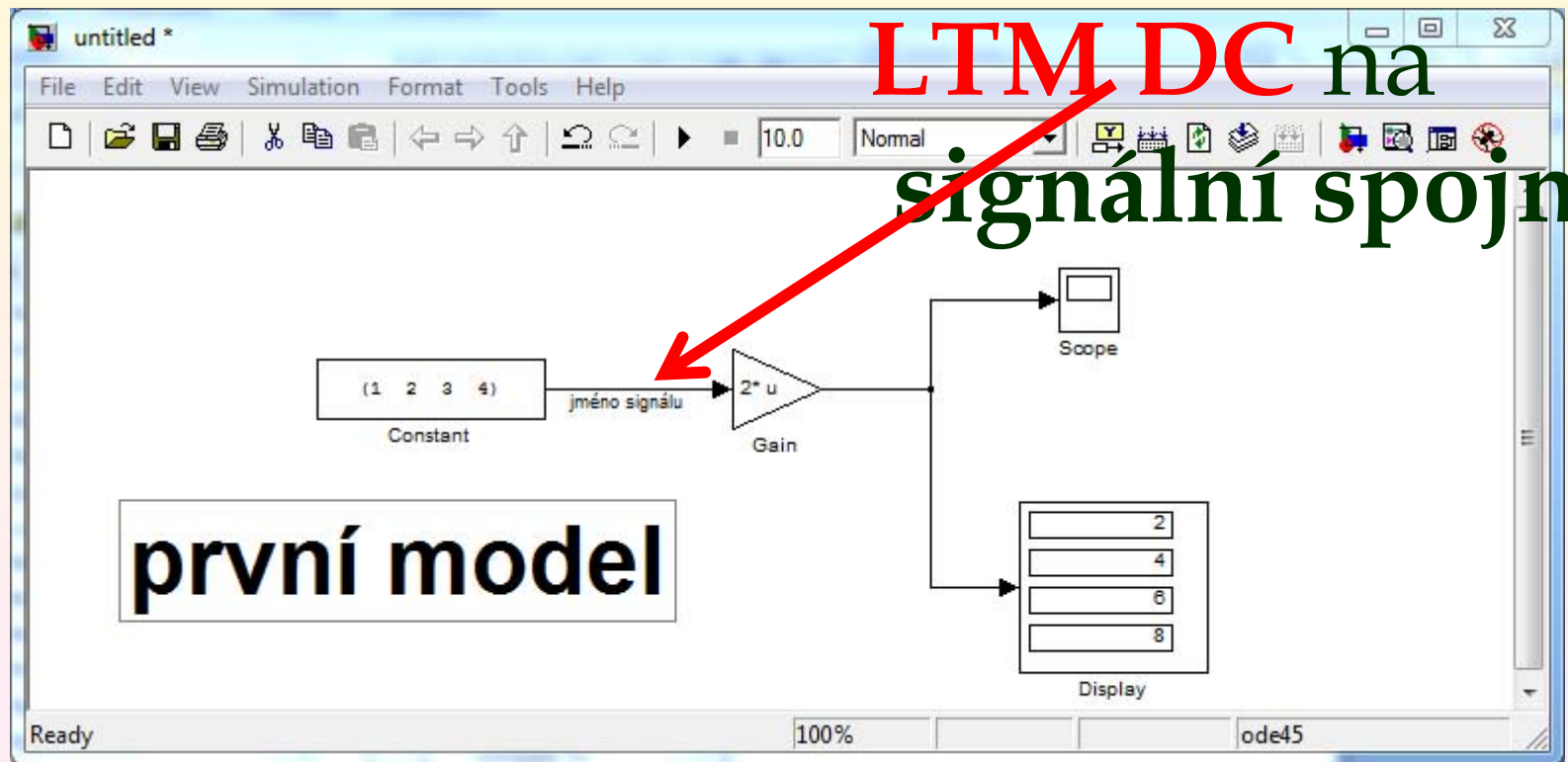
Číselná matice
nebo proměnná

... a obdobné to bude
i u ostatních modulů ...

SIMULINK

Název **signálu**

Pozor na diakritiku!!!



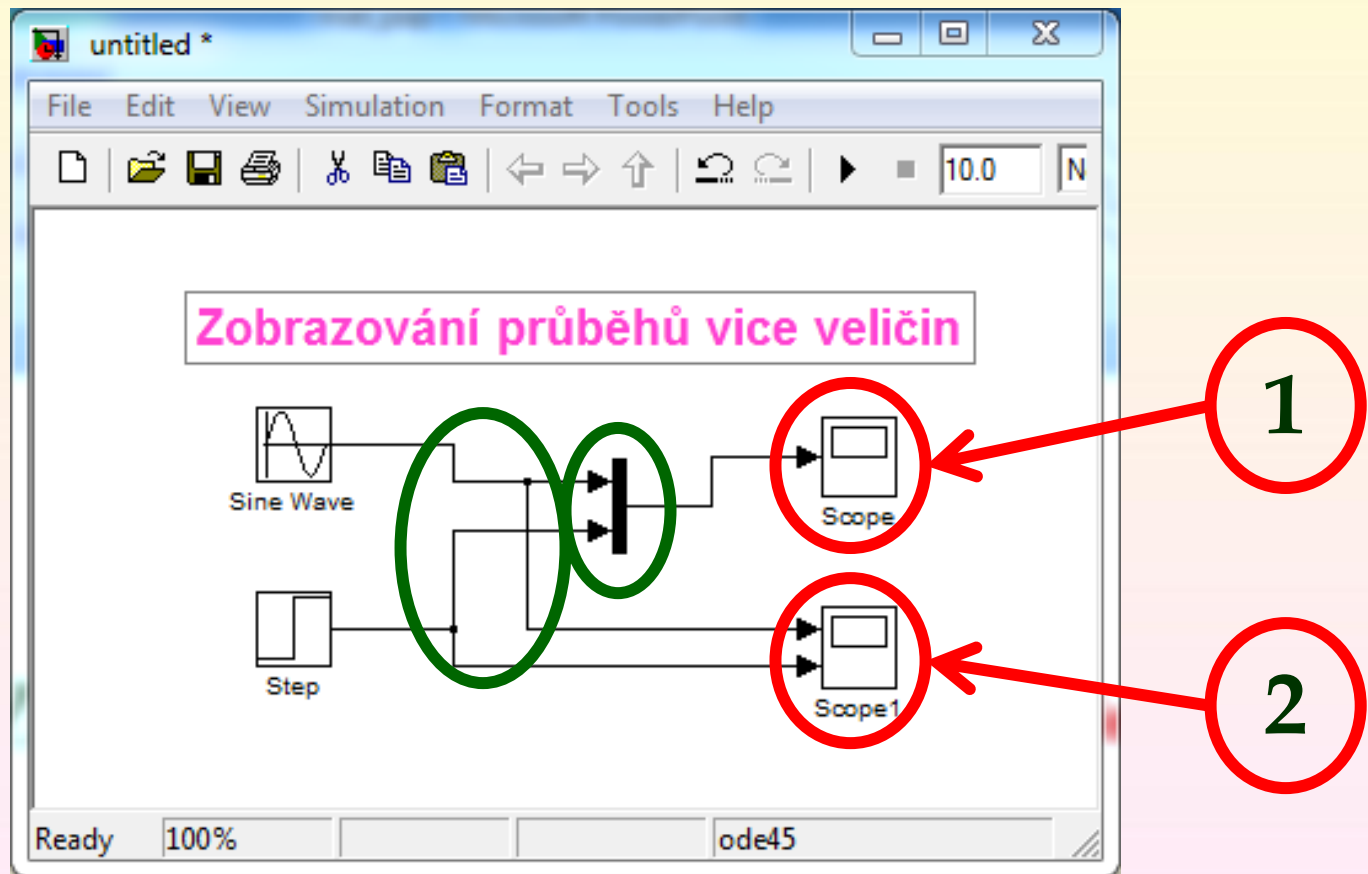
LTM DC na
signální spojnici

první model

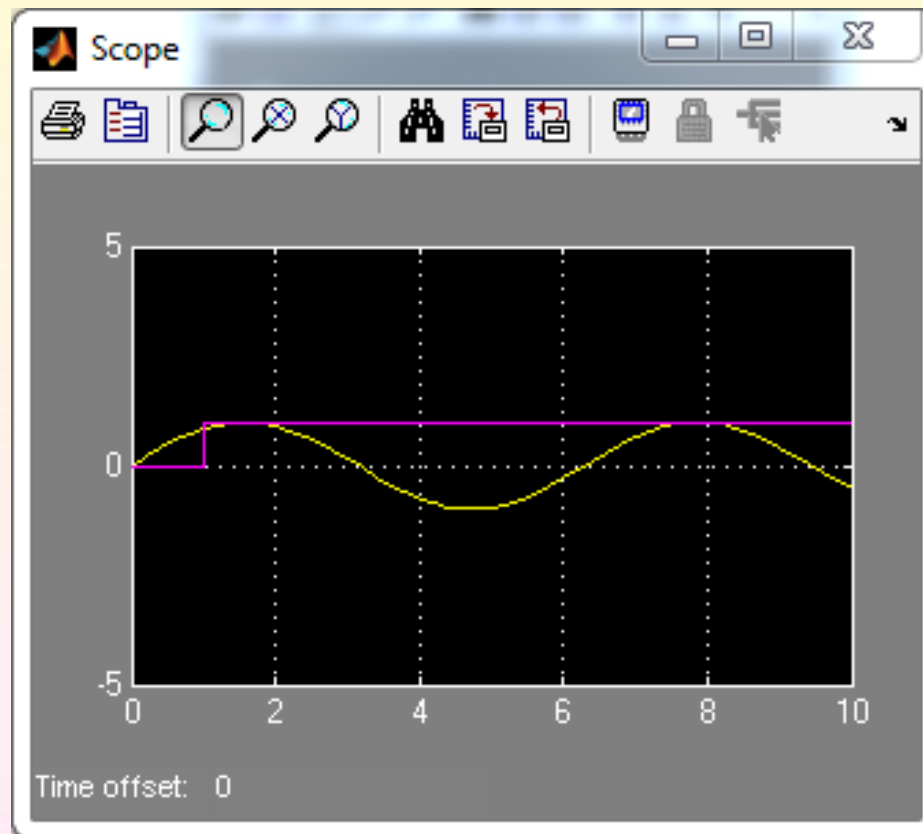
Úprava velikosti **displeje**.

Označím displej a velikost
upravíme tak, že budeme
táhat za **roh**y objektu.

Zobrazování průběhů na displeji.



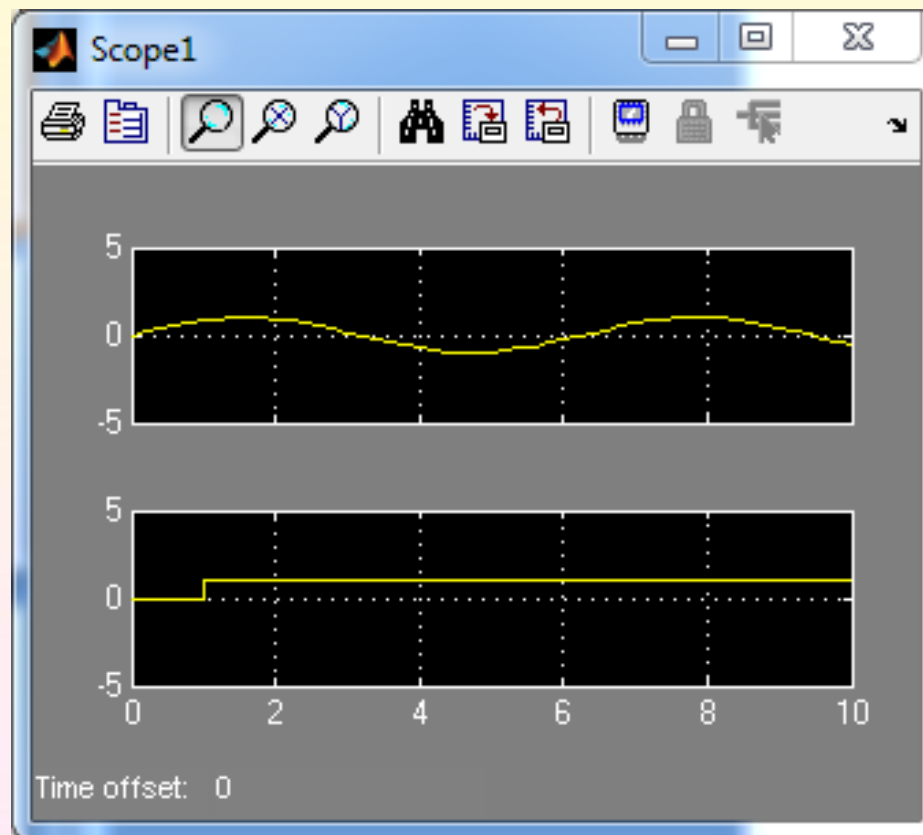
Zobrazování průběhů na **displeji**.



1

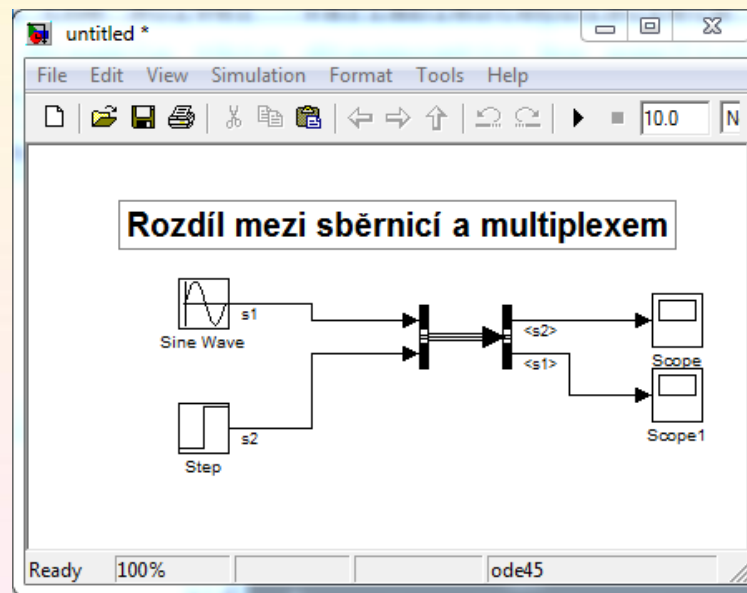
SIMULINK

Zobrazování průběhů na **displeji**.

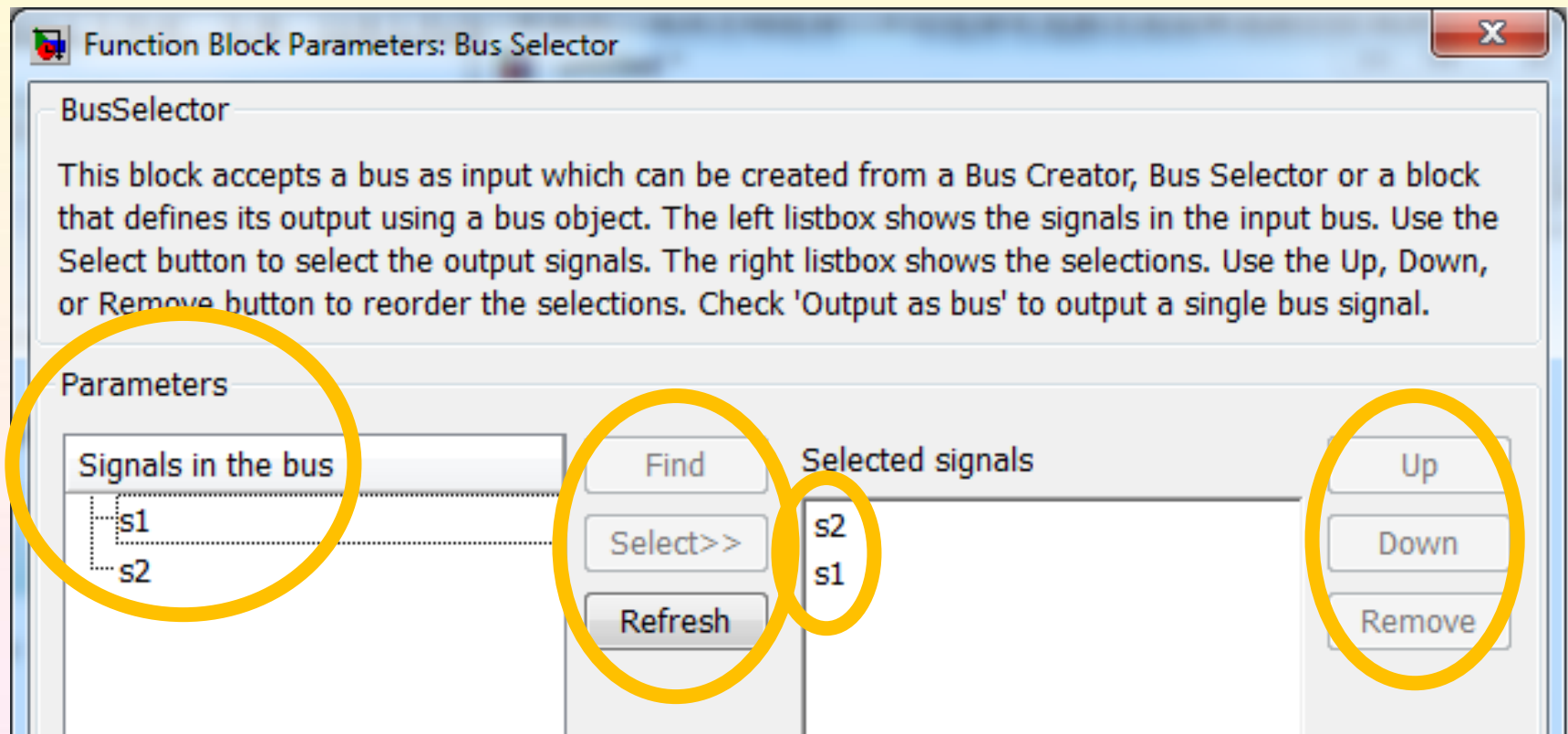


2

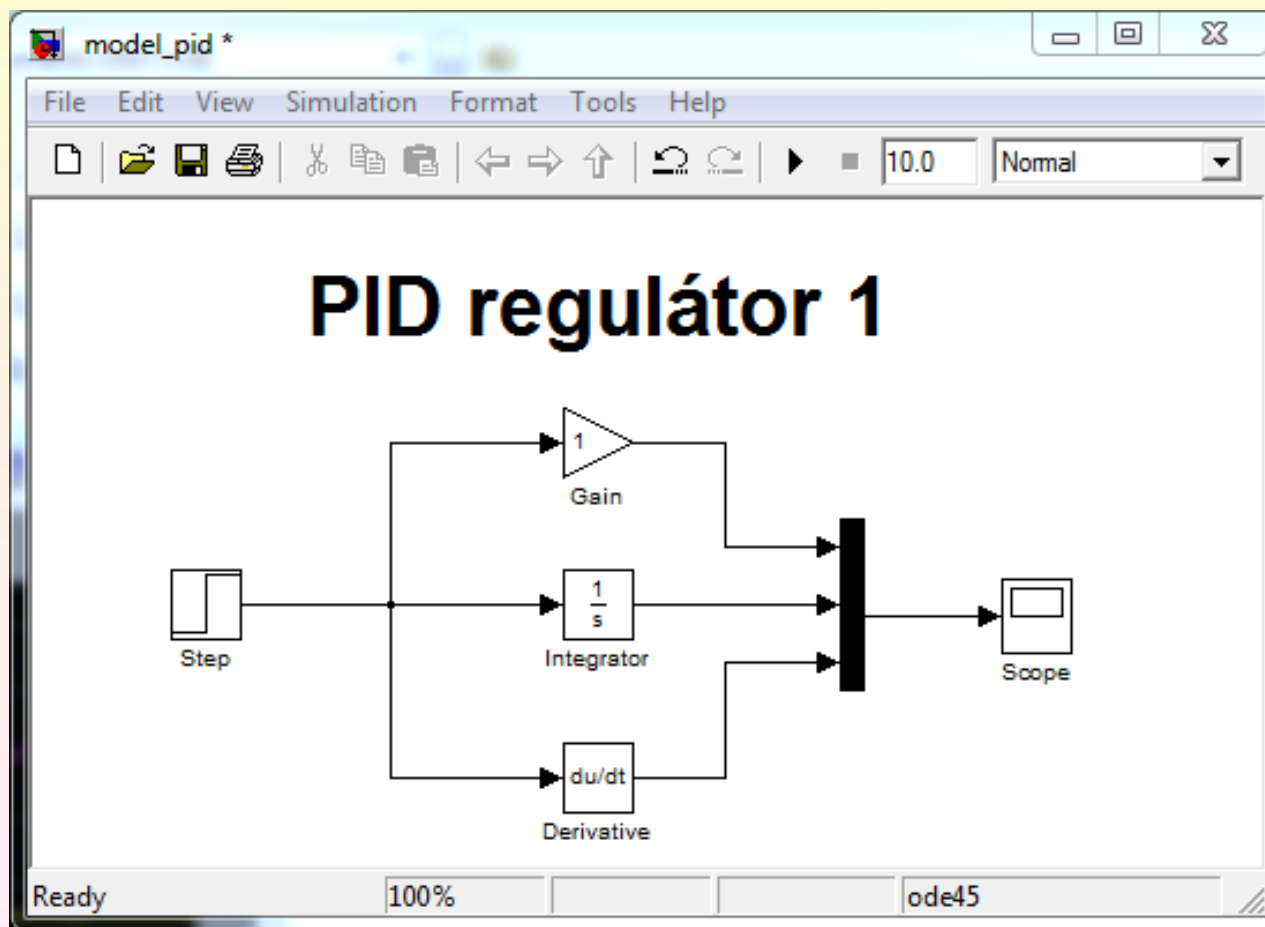
Zobrazování průběhů na displeji.

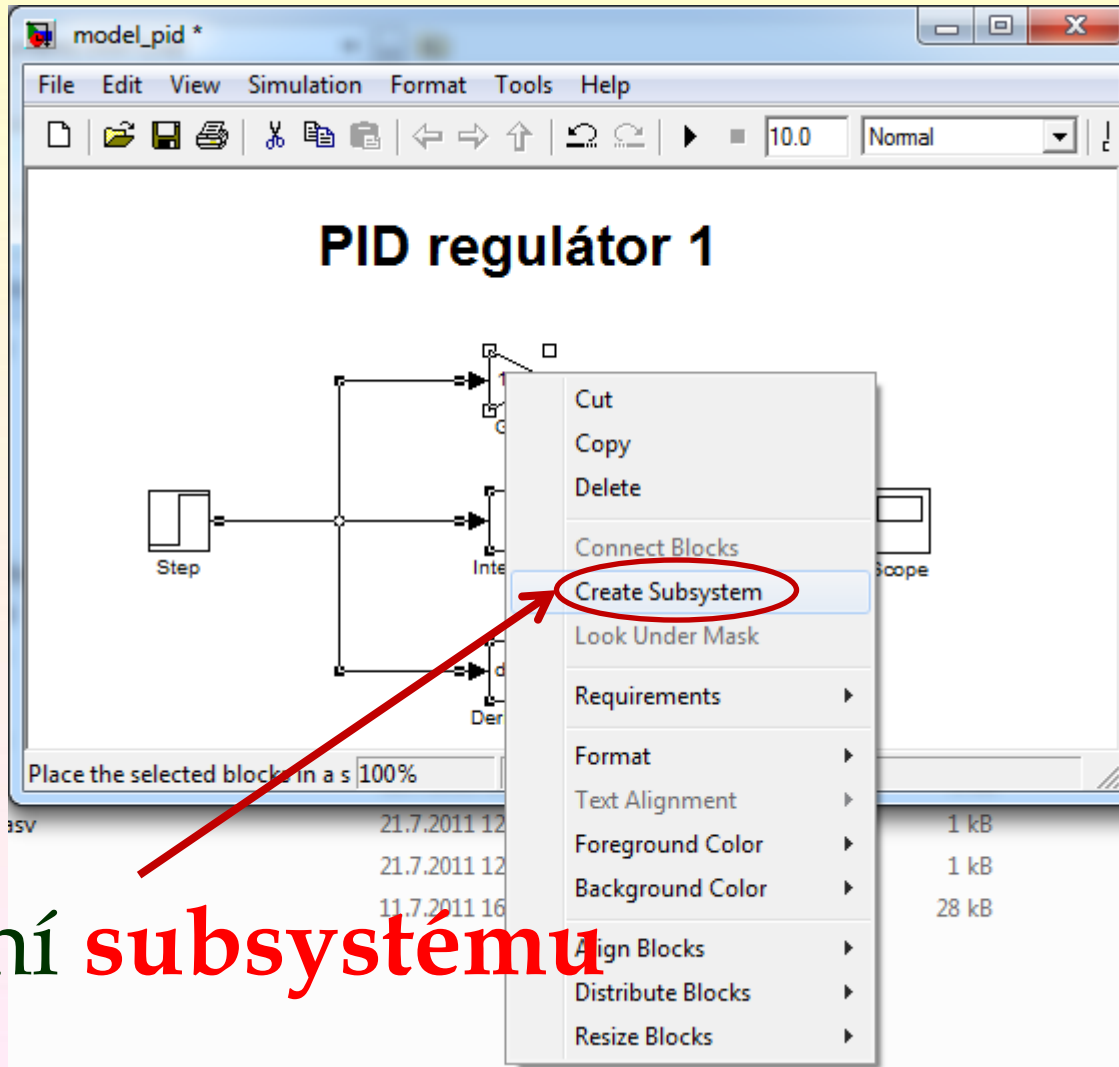


Zobrazování průběhů na displeji.



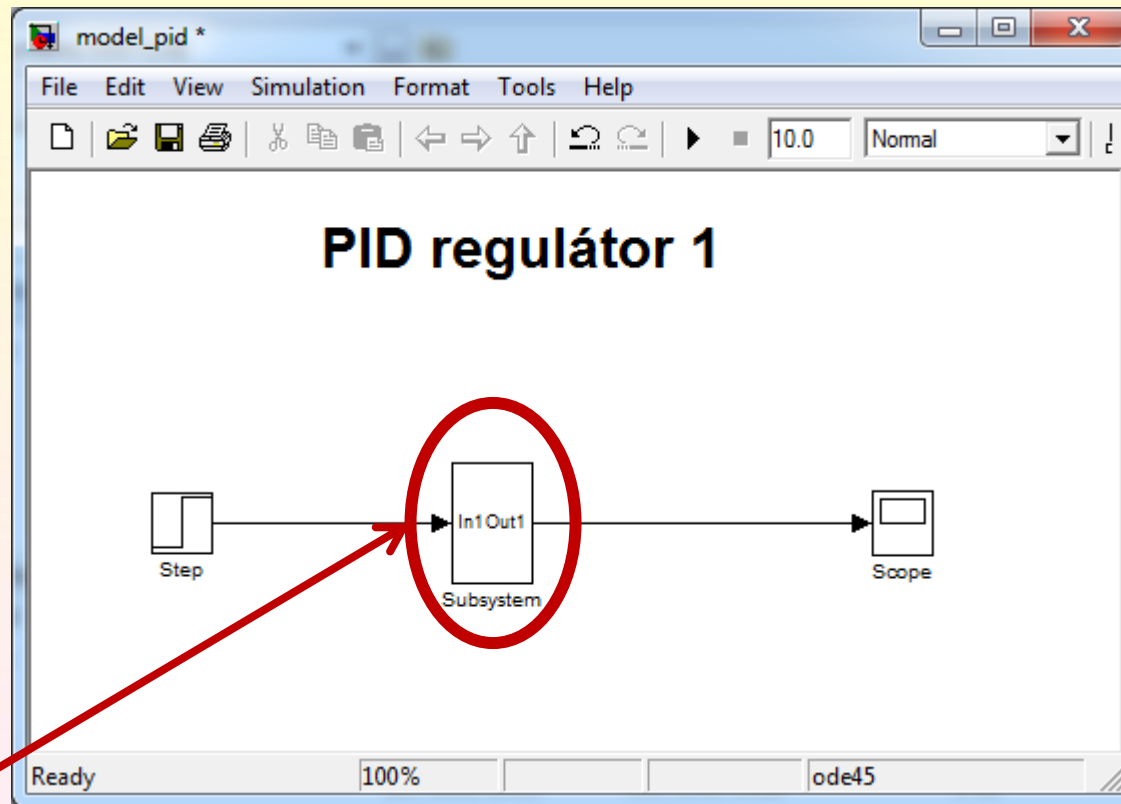
Vytvoření **subsystému**





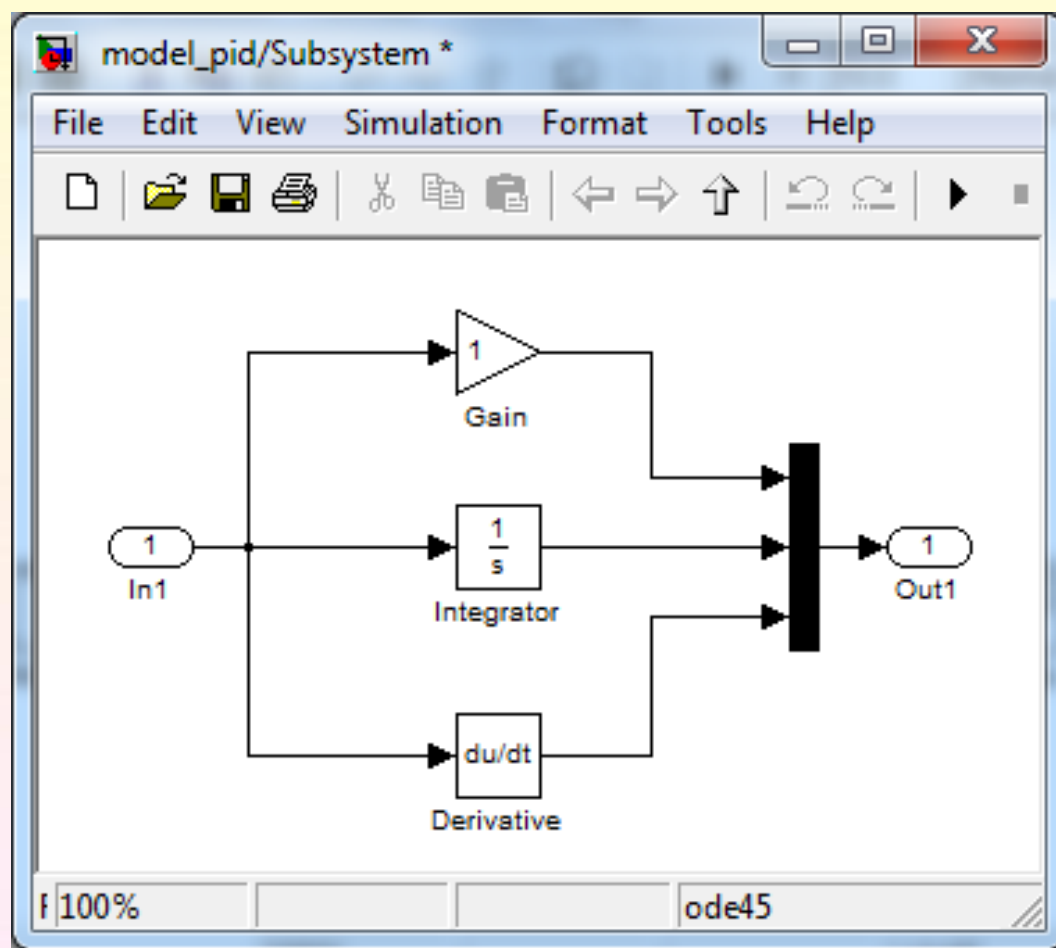
k

Vytvoření **subsystému**

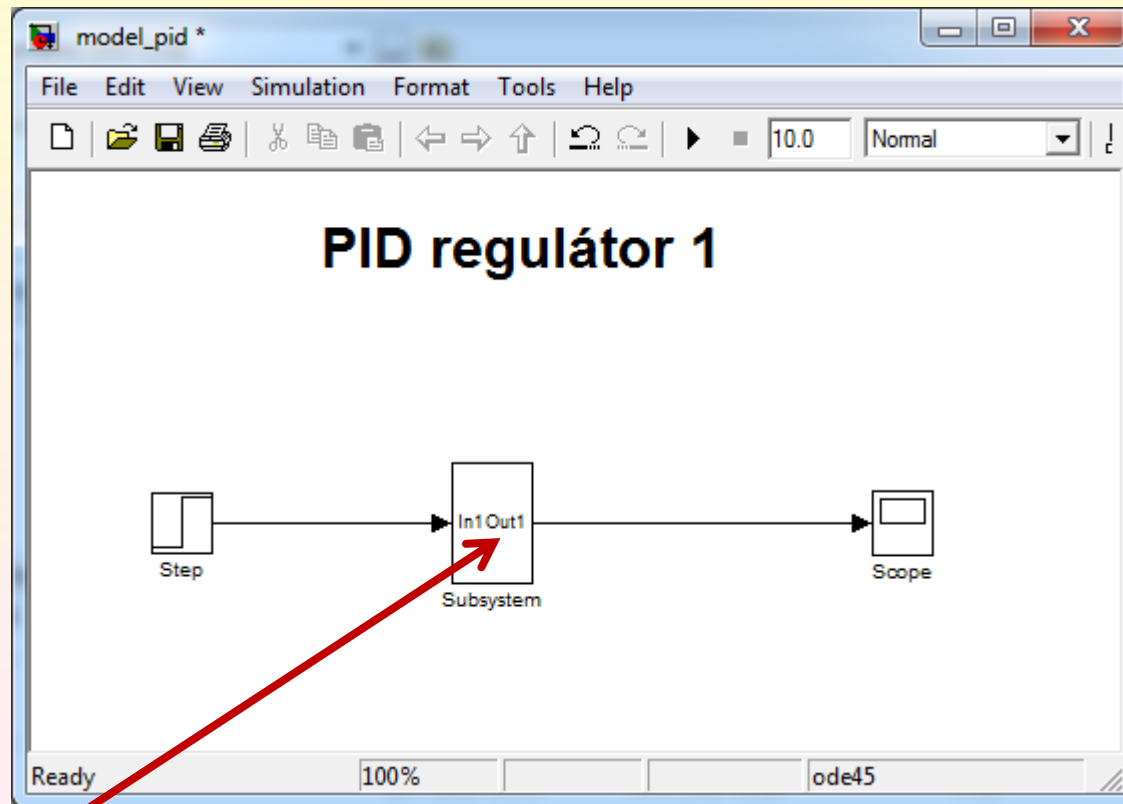


LTM DC a otevře se nám **subsystému**

Vytvoření **subsystemu**

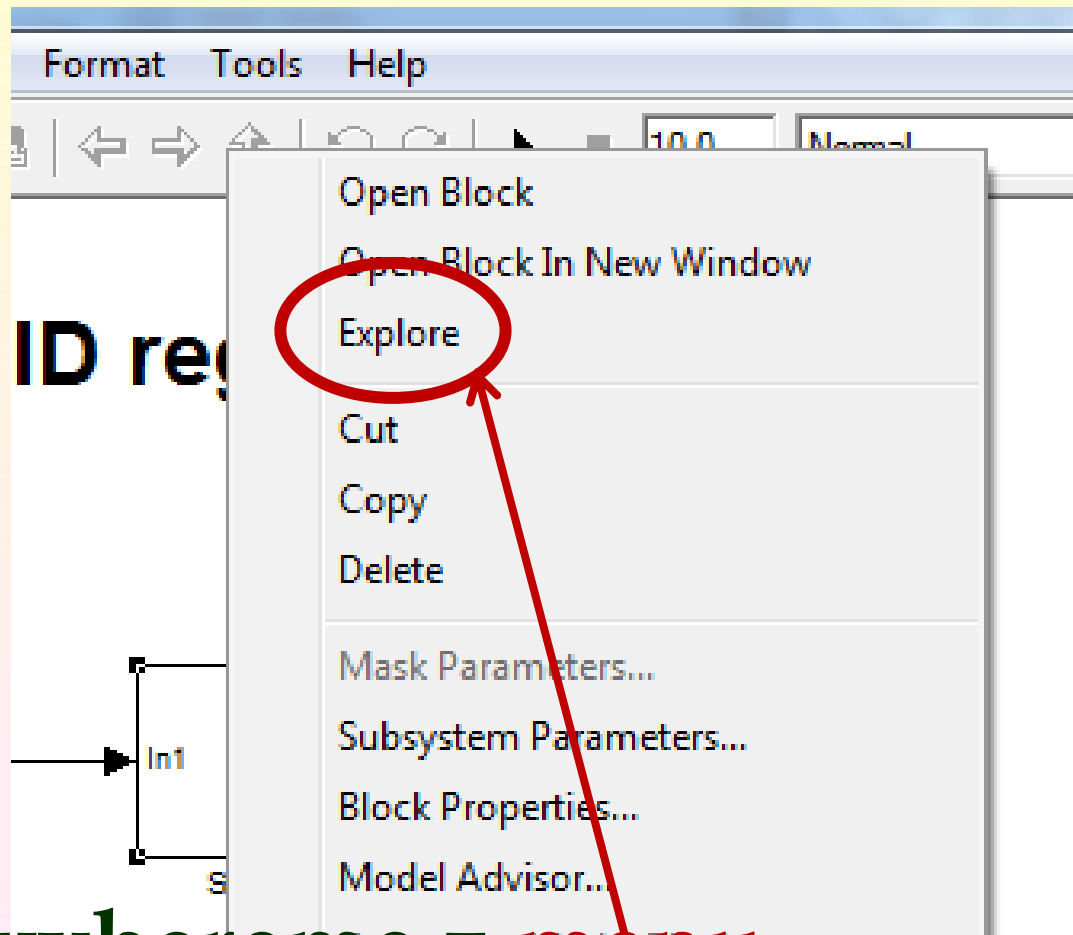


Vytvoření **subsystému**



PTM a otevře se nám **menu**

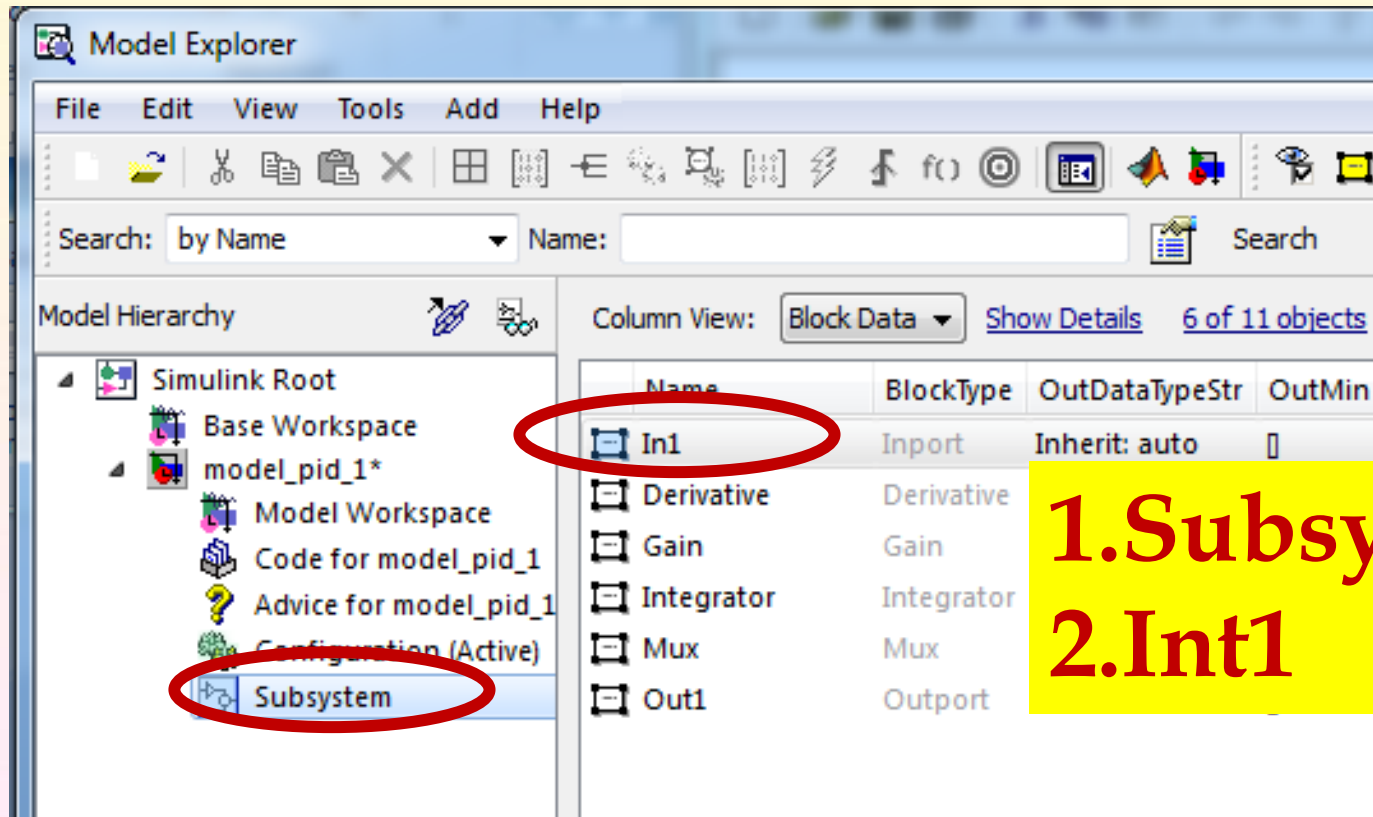
Vytvoření **subsystému**



a vybereme z **menu**

Vytvoření **subsystemu**

to samé můžeme udělat i s Out1



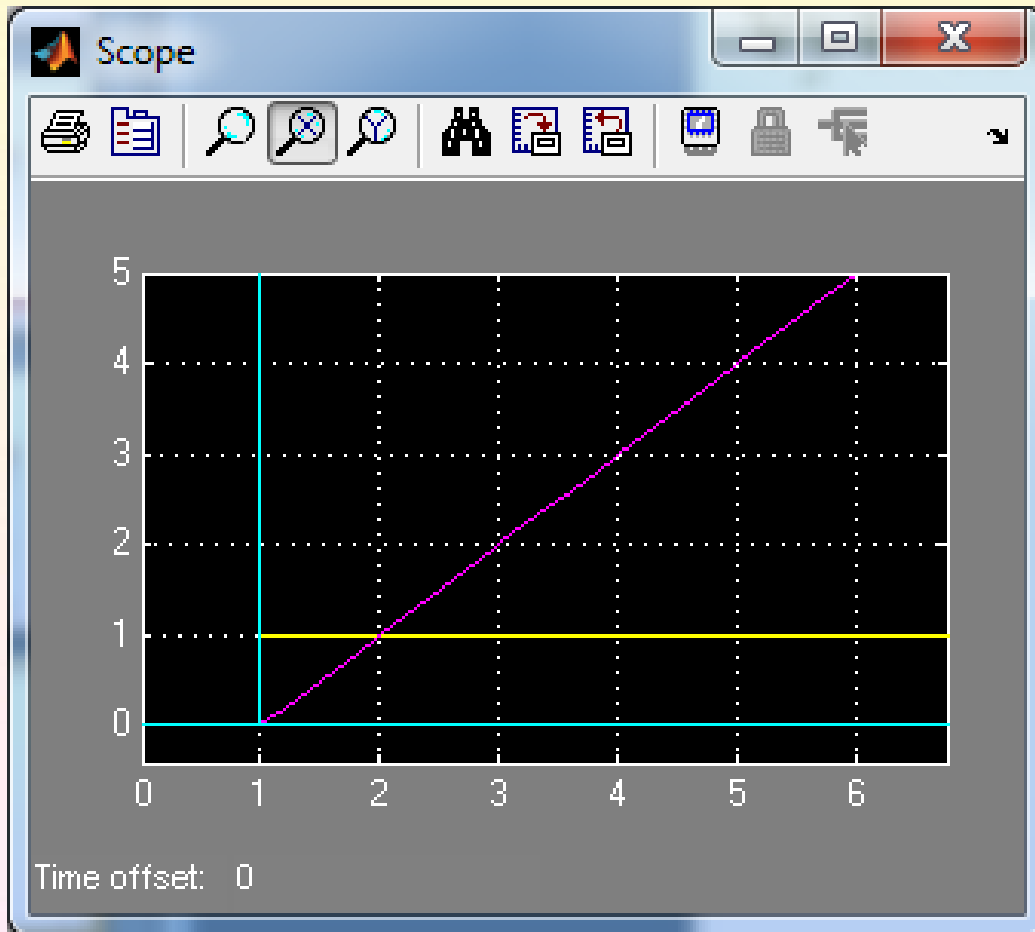
1.Subsystem
2.Int1

Vytvoření **subsystemu**

**přenesení subsystemu
do jiného modelu**

Ctrl+S – Ctrl+V

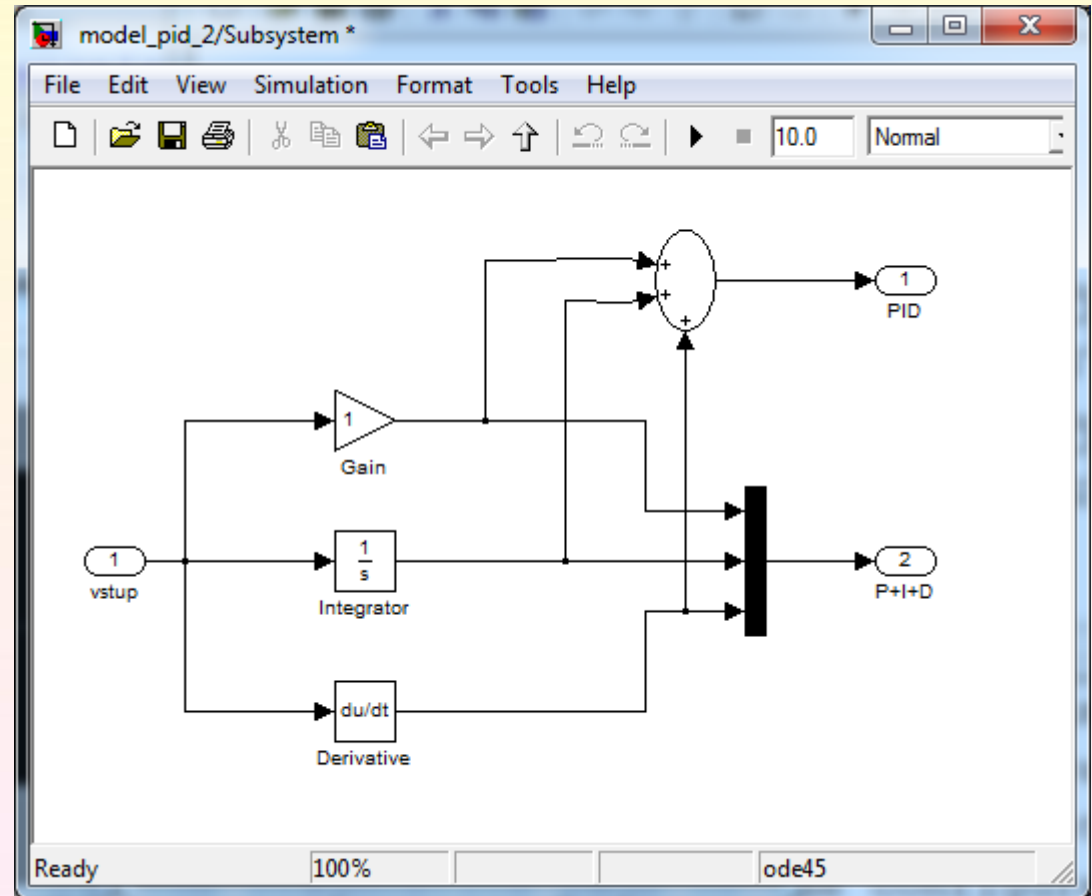
Vytvoření subsystému



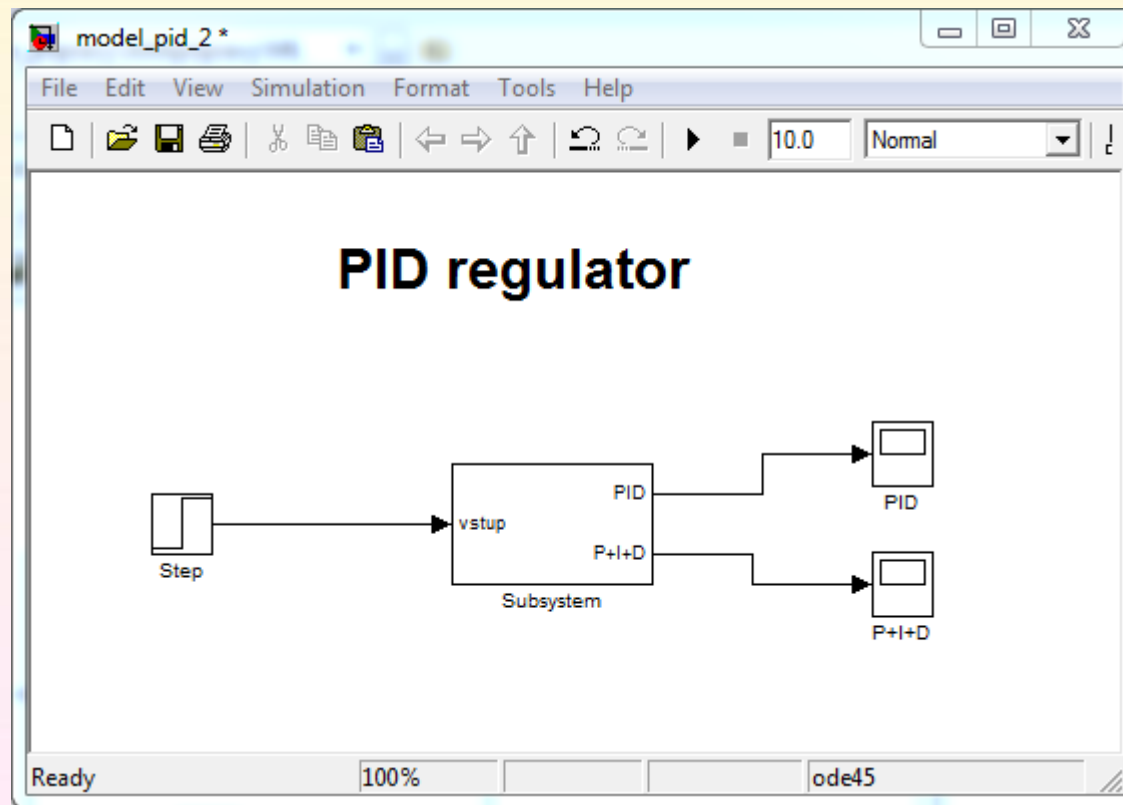
Zde se nám
zobrazila každá
složka zvlášť
a to **není** regulátor
PID

Vytvoření **subsystemu**

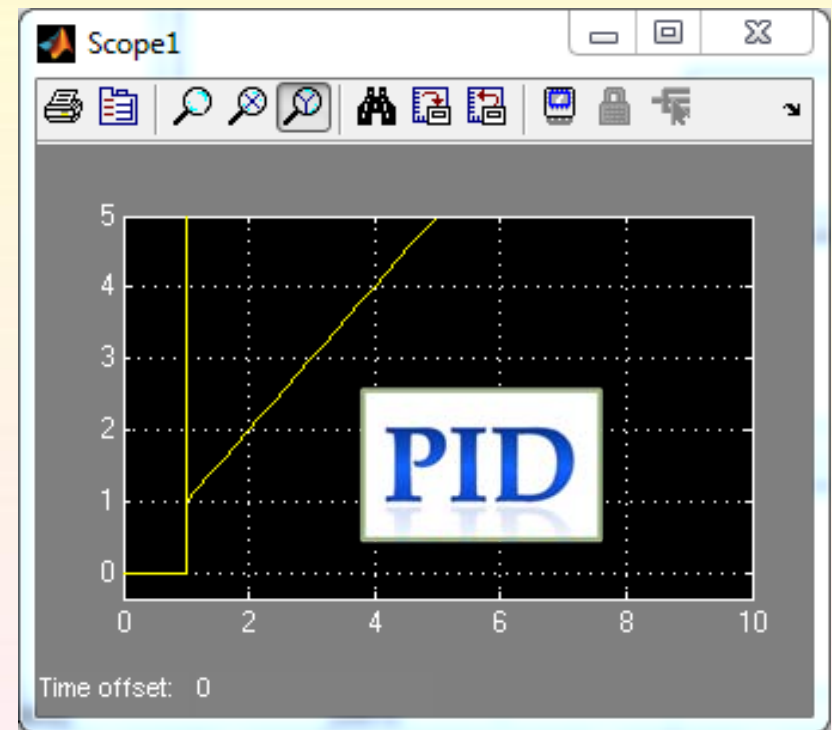
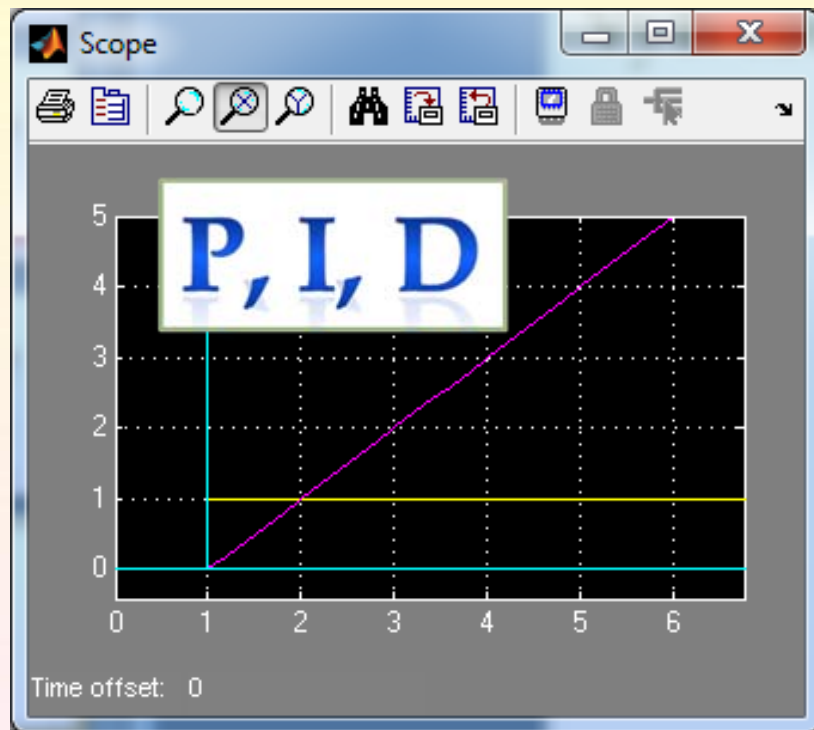
Úpravou
subsystemu
dostaneme
následující
schéma



Vytvoření **subsystemu** a takto vypadá upravený model



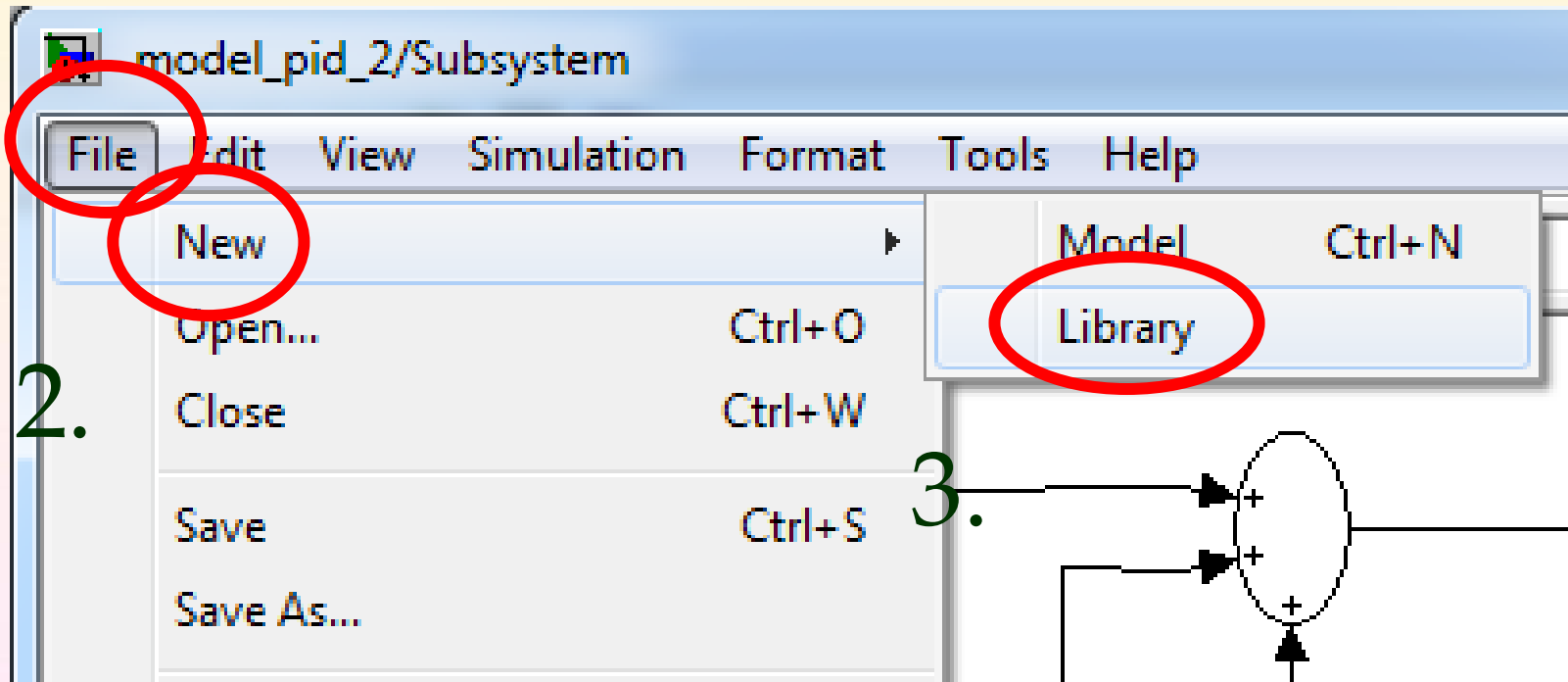
Vytvoření subsystému



Toto je rozdíl.

Vytvoření vlastní knihovny

1.

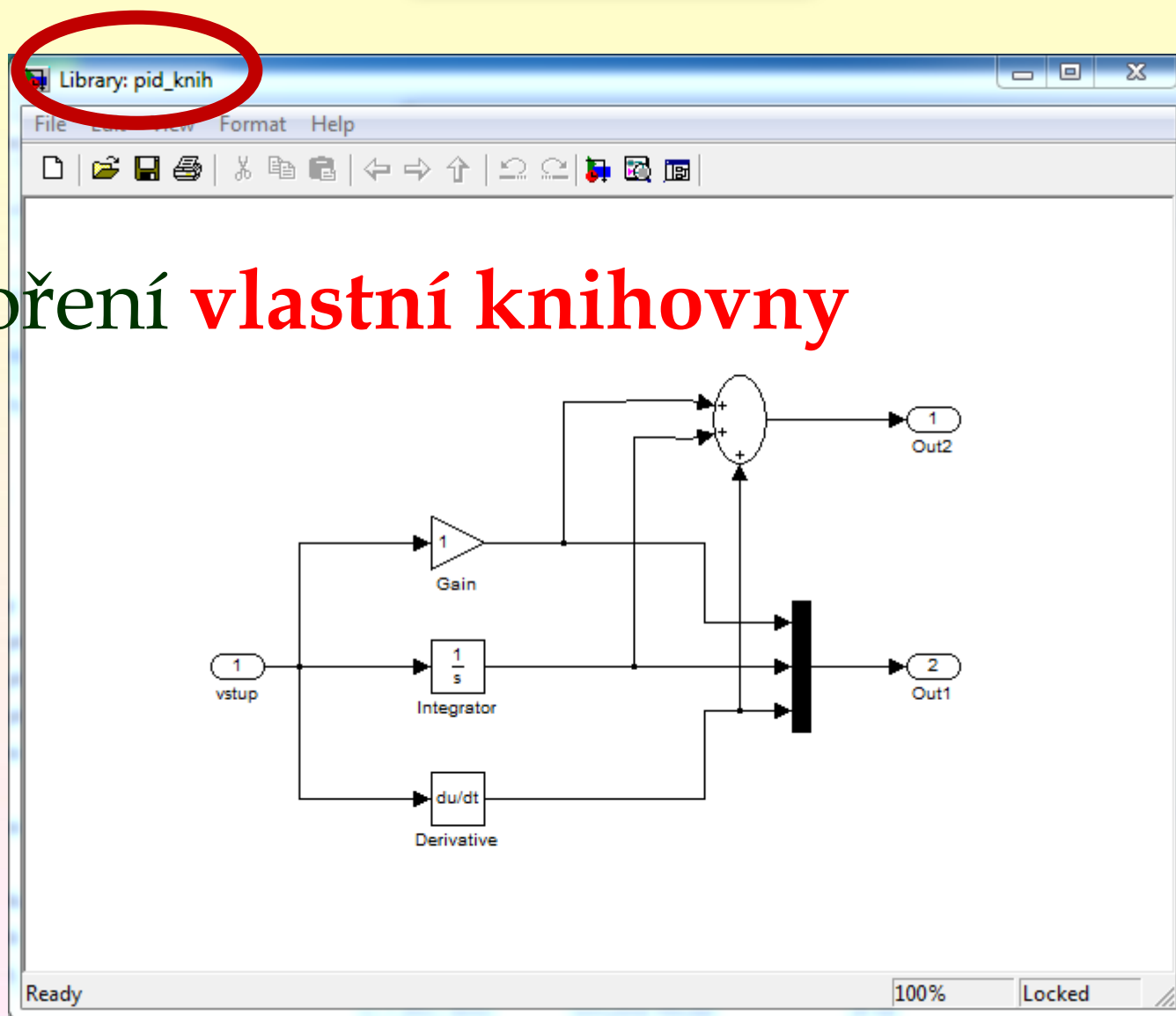


Vytvoření **vlastní knihovny**

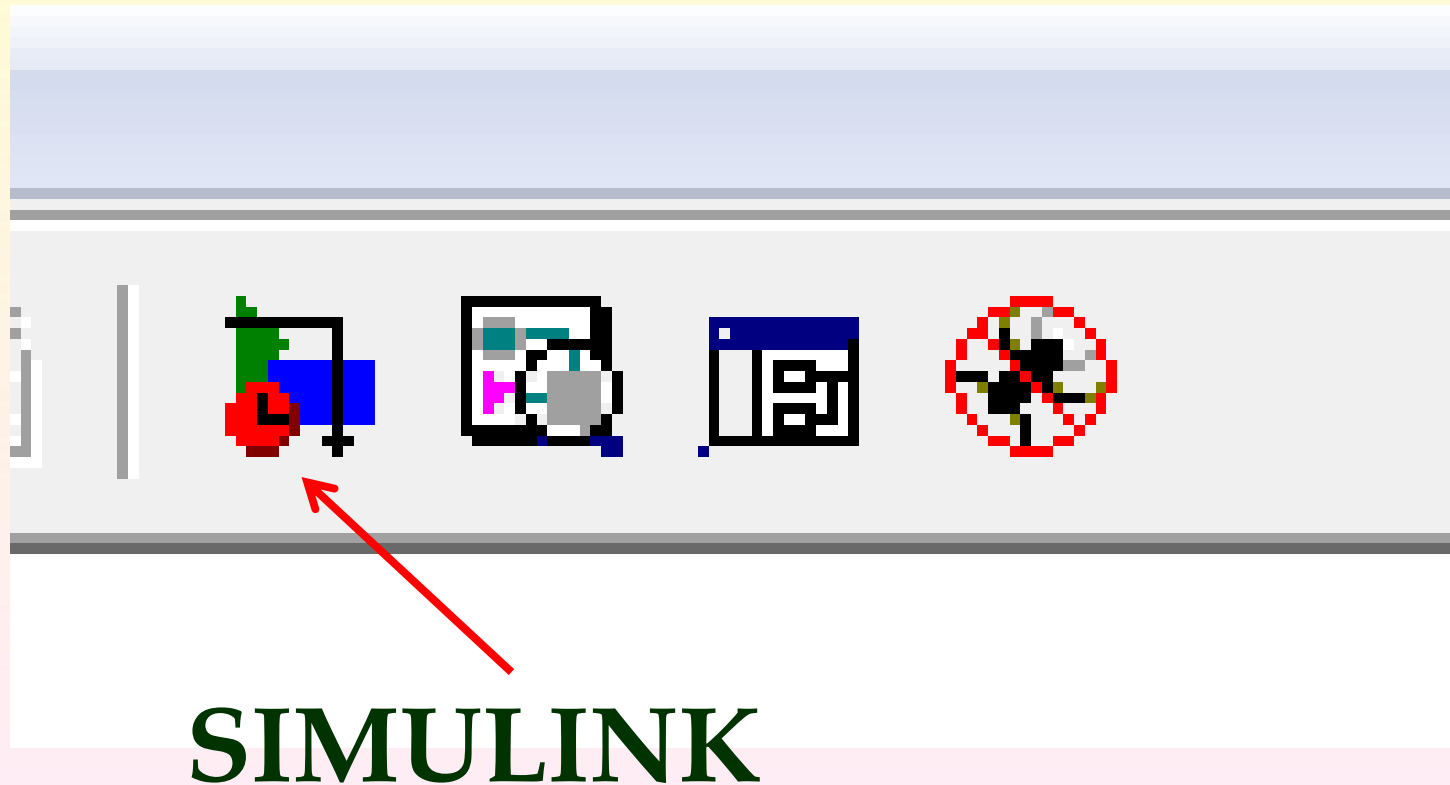
např. ze **subsystému** vyberu
to co budu chtít mít
v **knihovně** a zkopíruji
to do ní

SIMULINK

Vytvoření vlastní knihovny

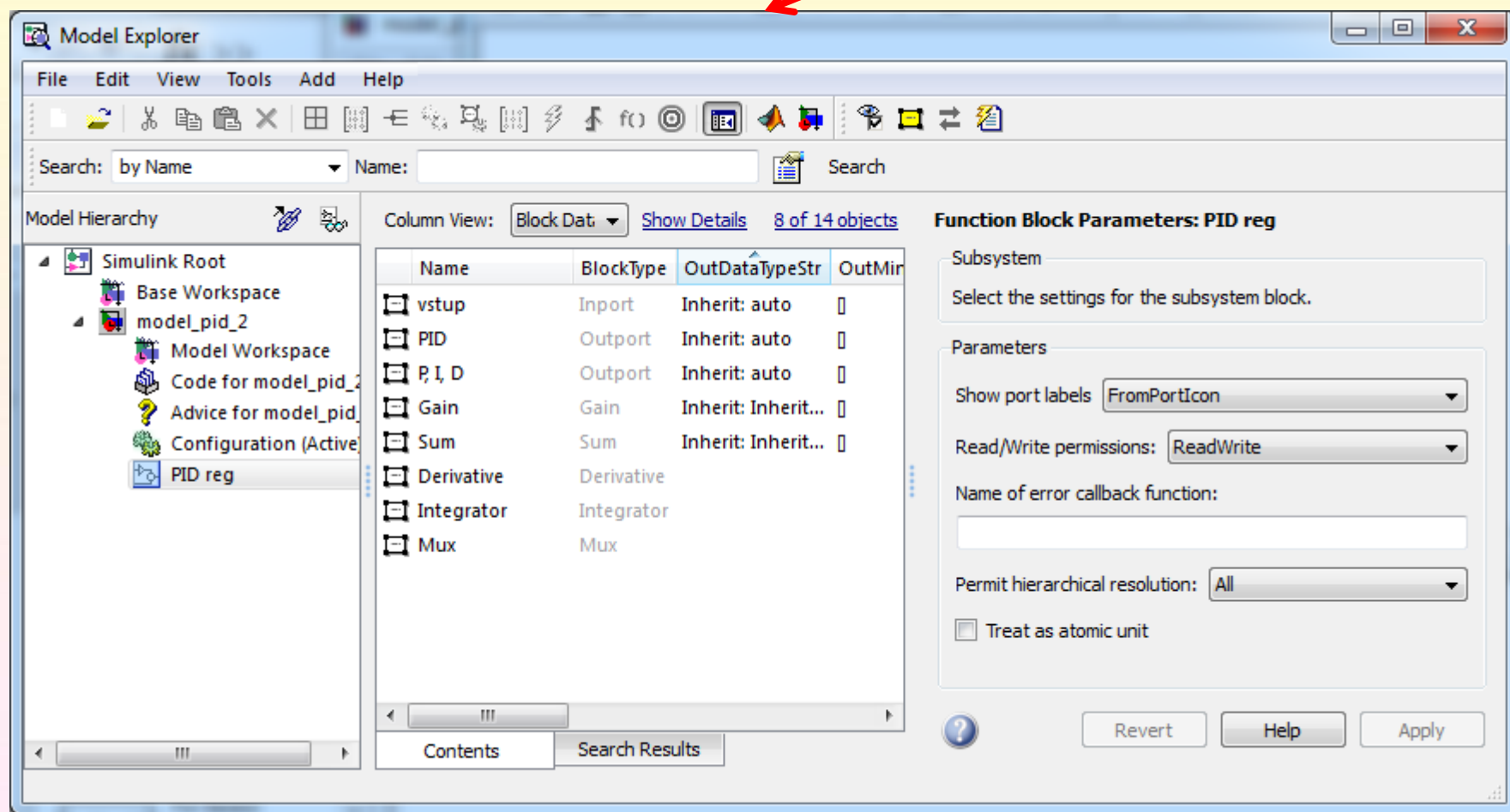


Něco o Panelu Nástrojů



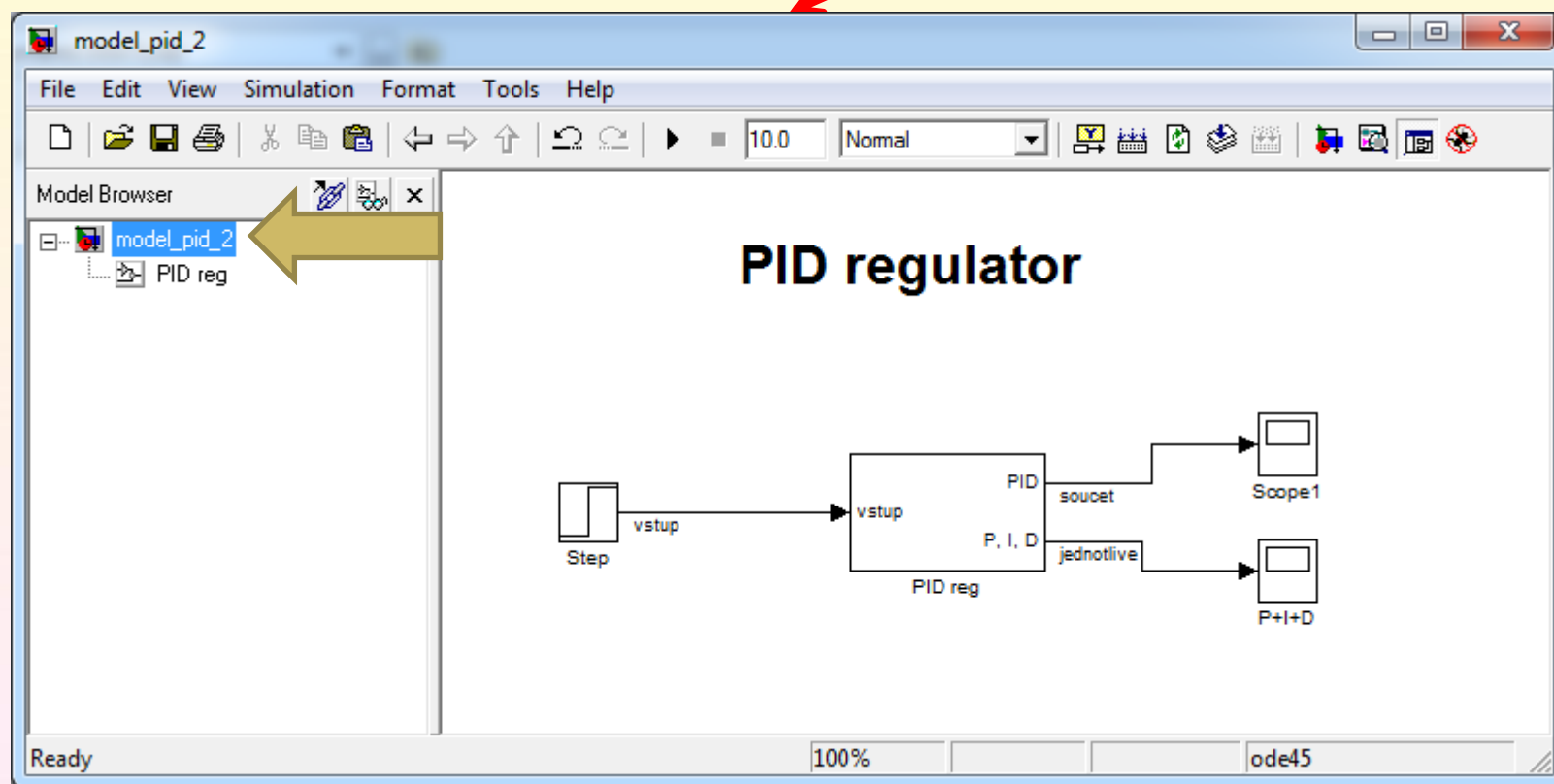
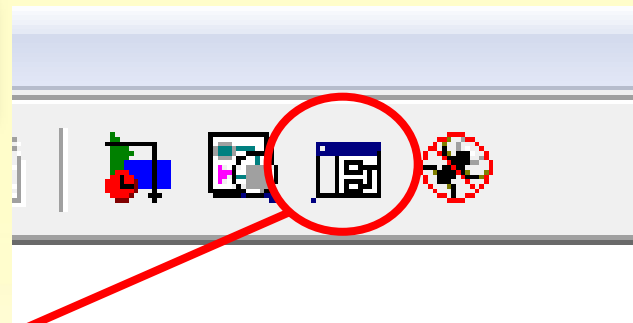
SIMULINK

Něco o Panelu Nástrojů



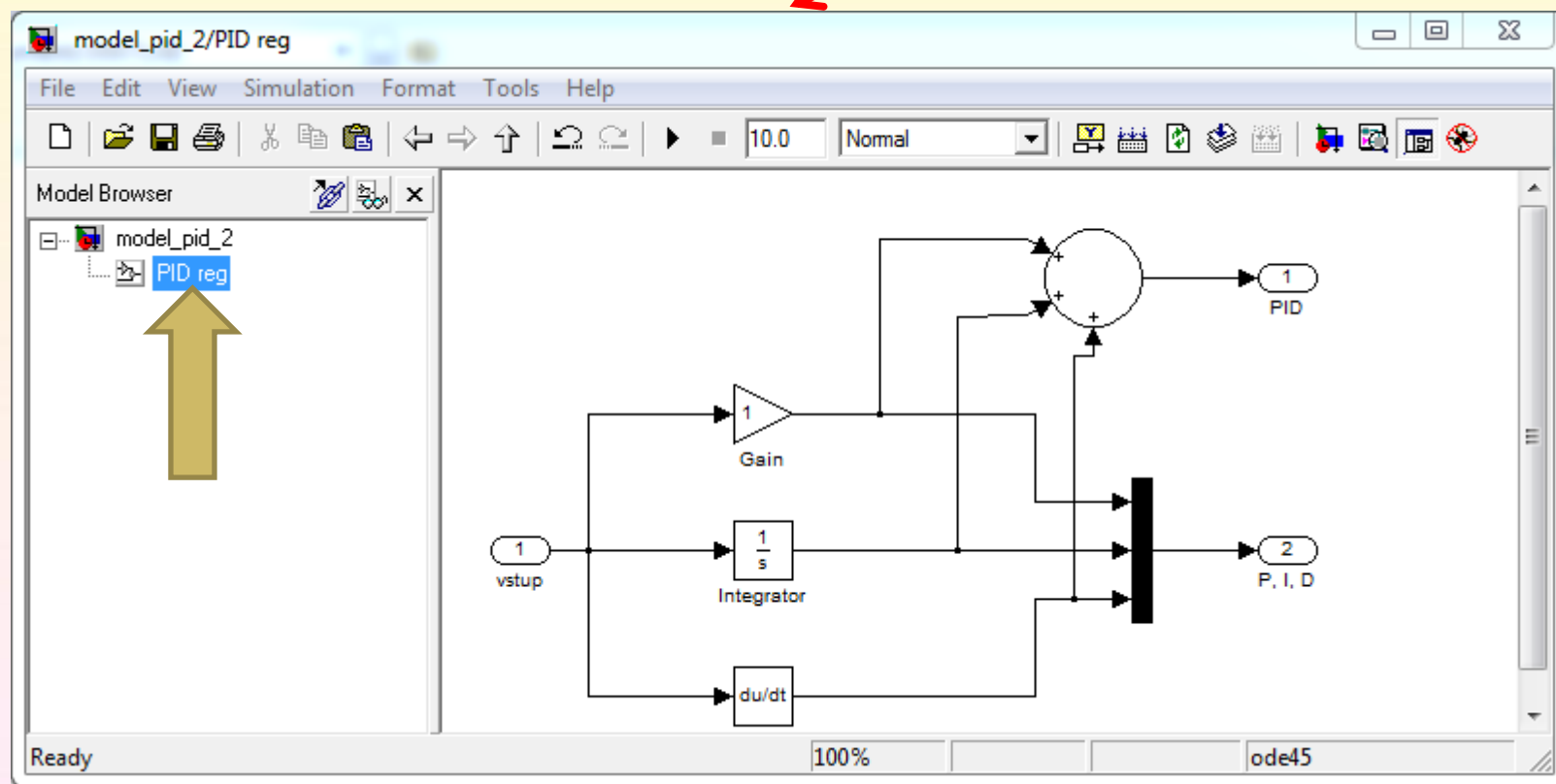
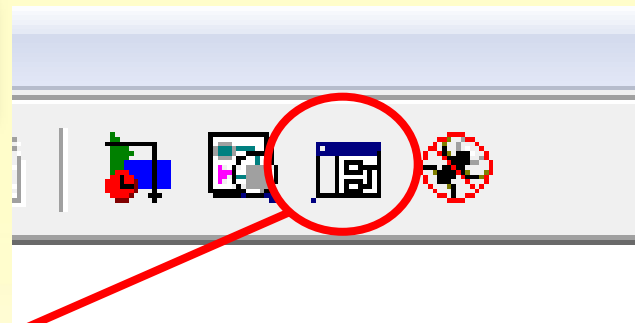
SIMULINK

Něco o Panelu Nástrojů

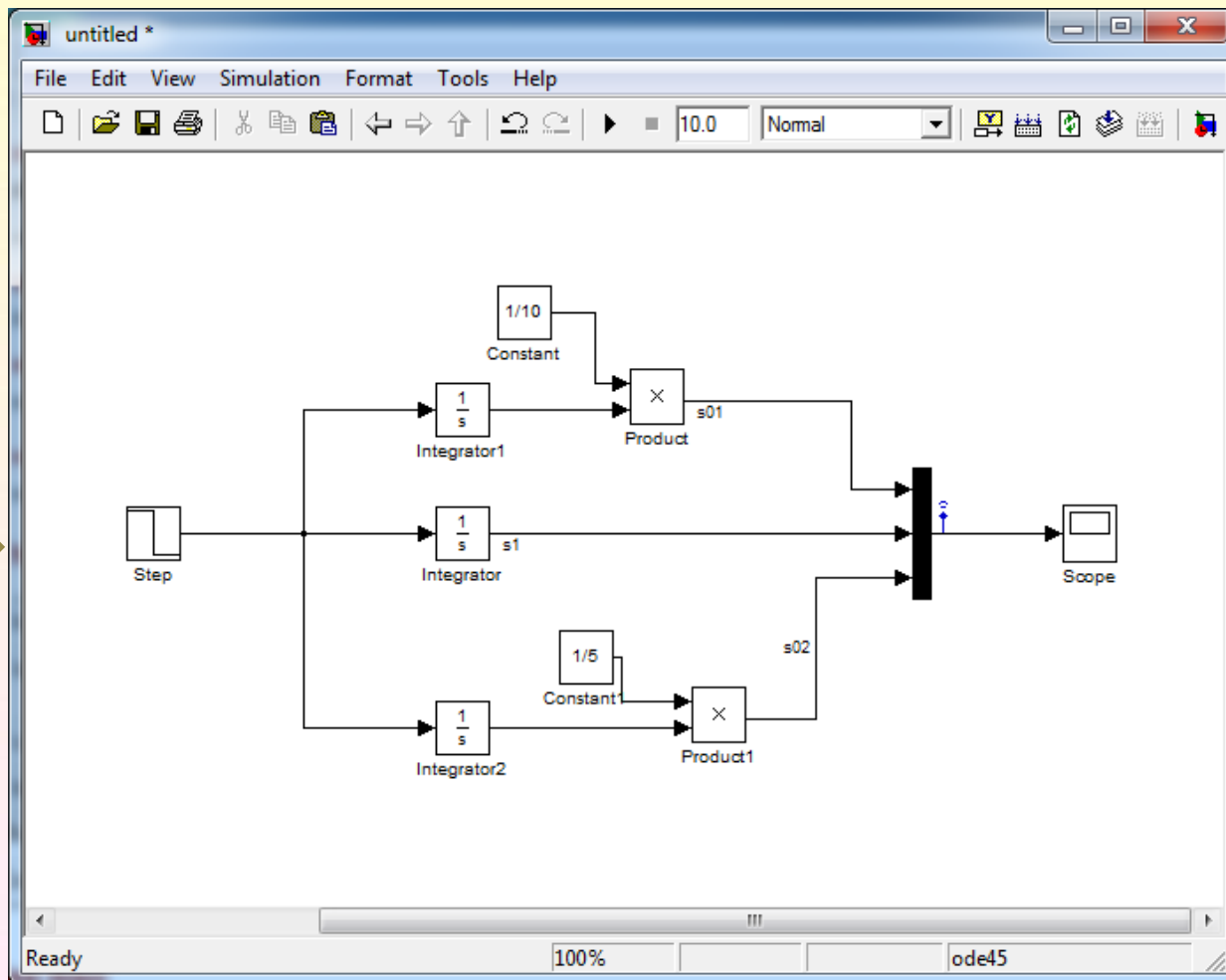


SIMULINK

Něco o Panelu Nástrojů



Jak s integrační (derivační) složkou???



Nechceš,
nemusíš,
ale je to
DŮ

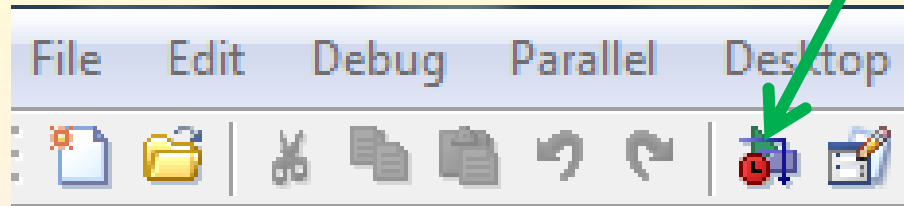
**Budeme řešit diferenciální rovnici –
např. –**

$$u = 2y + 3y' + 5y'' + y'''$$

**což není nic jiného než RS statická 3°
- tříkapacitní**

SIMULINK

Z PN vybereme tuto **ikonu**

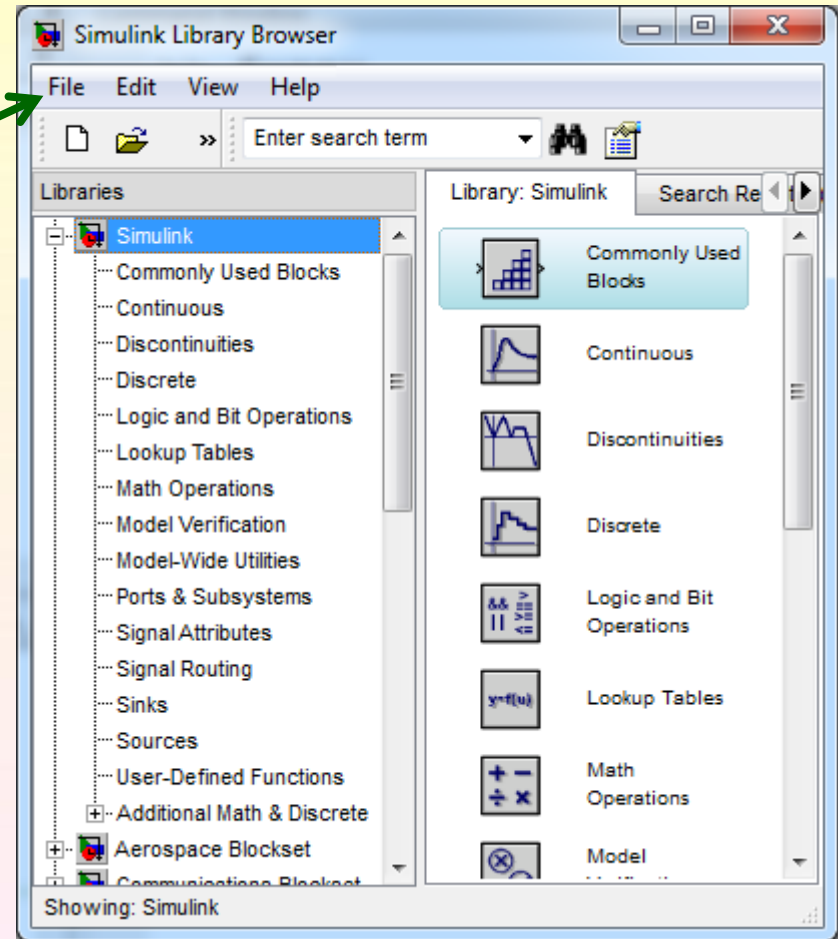


nebo napíšeme do příkazového okna
simulink
a to bude výsledek

SIMULINK

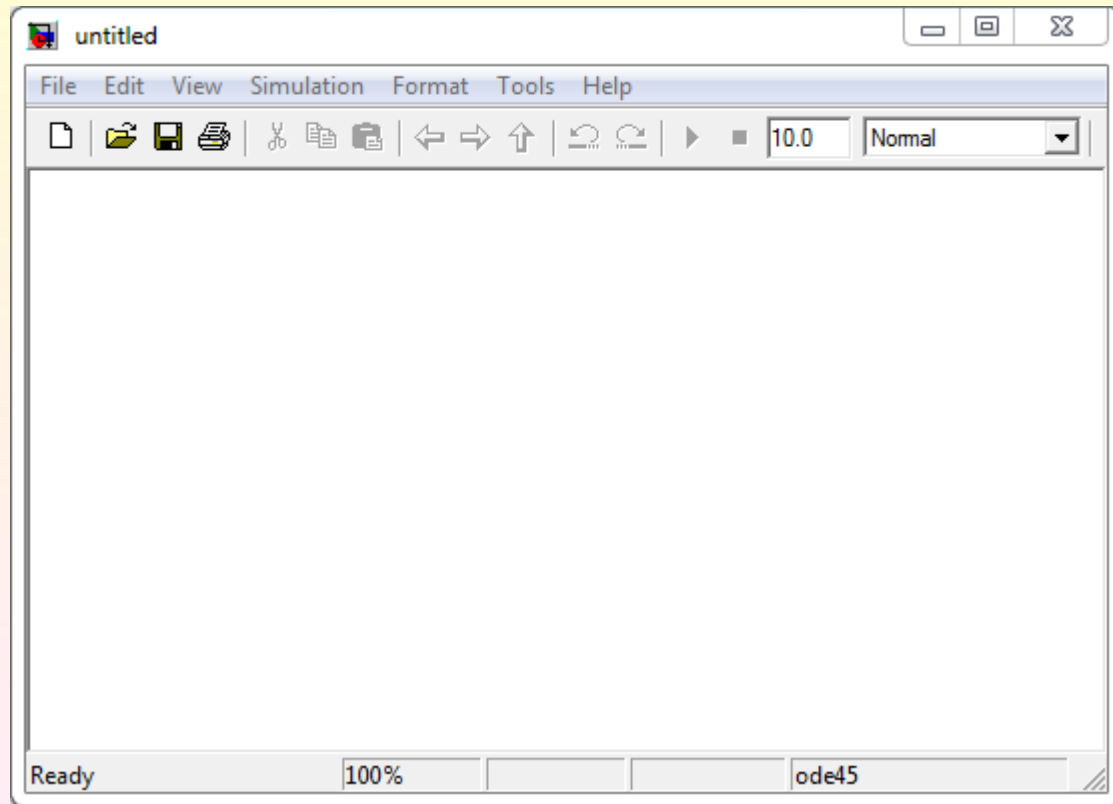
Z HNM vybereme
File/new/model
a tím získáme okno
pro grafické
řešení DR

Tím se nám objeví
velký seznam knihoven,
ze kterých si můžeme
vybírat dle libosti.



A nyní ve spolupráci
s knihovnamy si
„poskládáme“
DR.

Takto vypadá
okno pro nový
„model“



Nejprve provedeme úpravu DR
na následující tvar -

$$y''' = u - 2y - 3y' - 5y''$$

$$y = \iiint u - 2 \iiint y - 3 \iint y - 5 \int y$$

Nyní si provedeme zobecnění DR.

PROČ?

$$a_3 y''' = b_0 u - a_0 y - a_1 y' - a_2 y''$$

$$a_3 y = b_0 \iiint u - a_0 \iiint y - a_1 \iint y - a_2 \int y$$

a toto máme již upravený výraz!!!!!!!!!!!!!!

$$y = \frac{b_0}{a_3} \iiint u - \frac{a_0}{a_3} \iiint y - \frac{a_1}{a_3} \iint y - \frac{a_2}{a_3} \int y$$

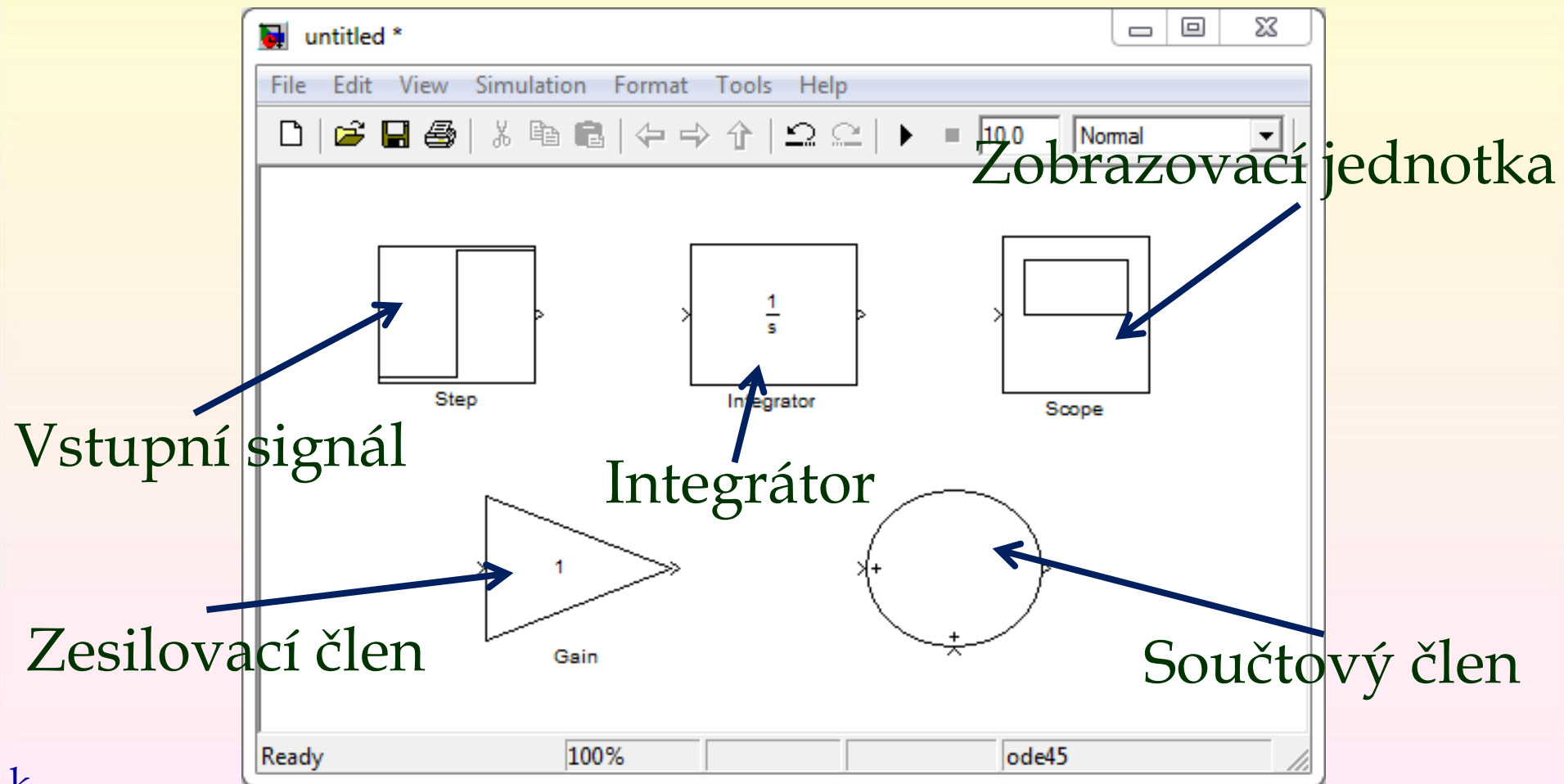
a toto jsou naše skutečné hodnoty

$$a_3 = 1; a_2 = 5; a_1 = 3;$$

$$a_0 = 2; b_0 = 1$$

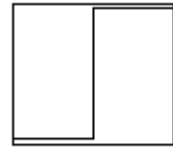
SIMULINK

Nyní si vybereme potřebné moduly.

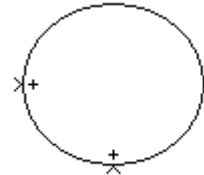


Kolik kterých modulů bude potřeba?

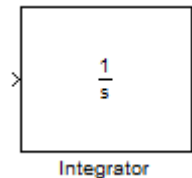
Vstupní signál - 1x



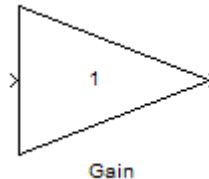
Součtový člen - 3x



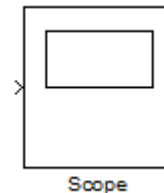
Integrátor - 3x



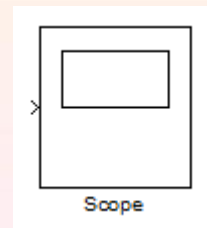
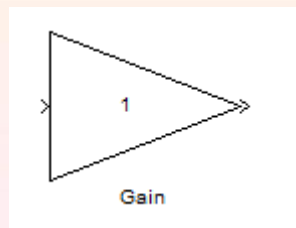
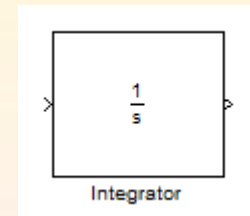
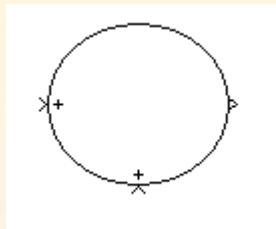
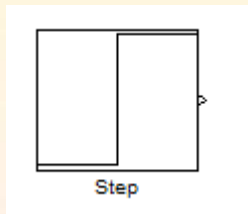
Zesilovací člen - 3x



Zobrazovací jednotka - 1x



A teď si řekneme, jak se editují jednotlivé moduly.



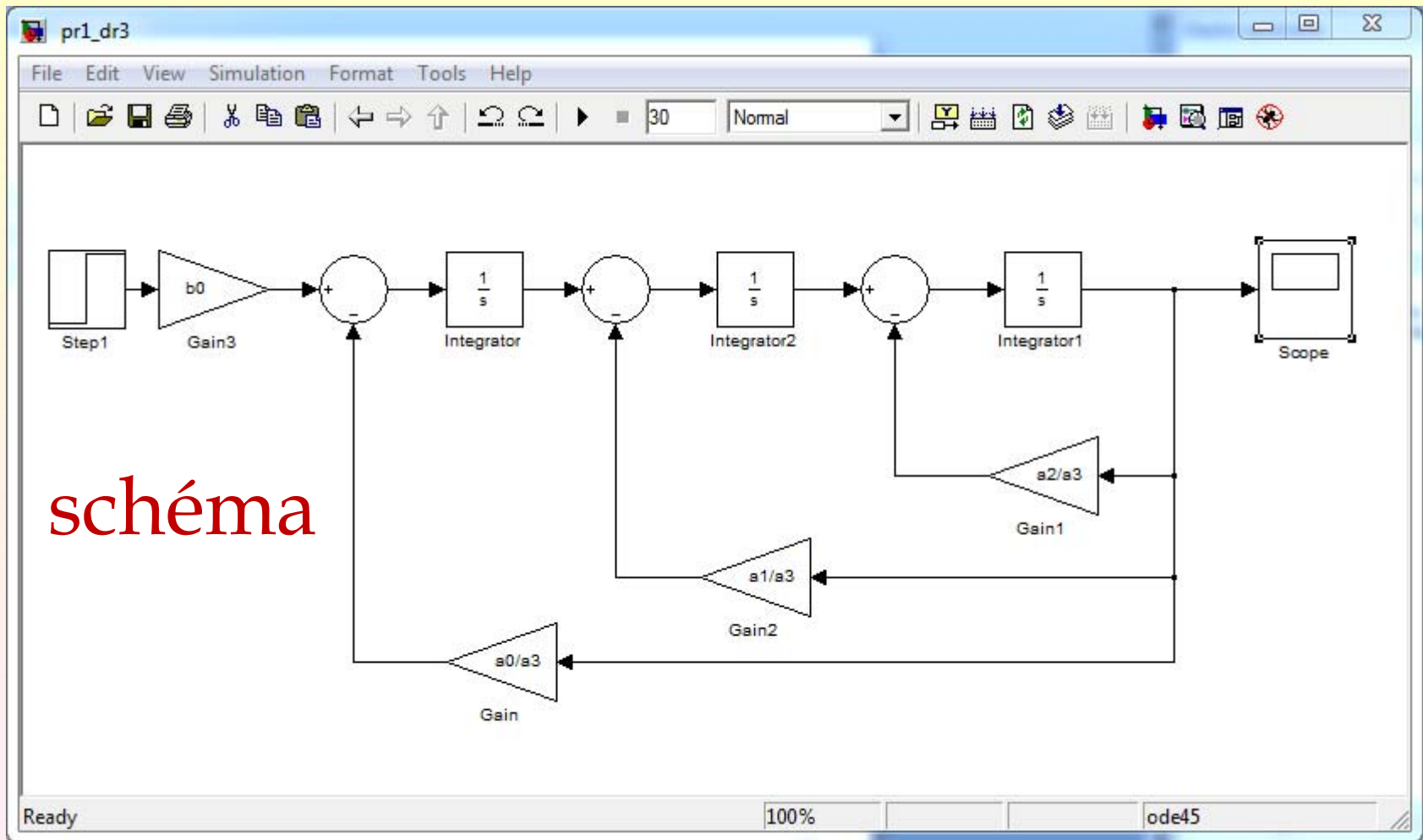
u dalších je to obdobné

Než spustíme simulaci, je třeba nastavit správně konstanty a a b a to v ML.

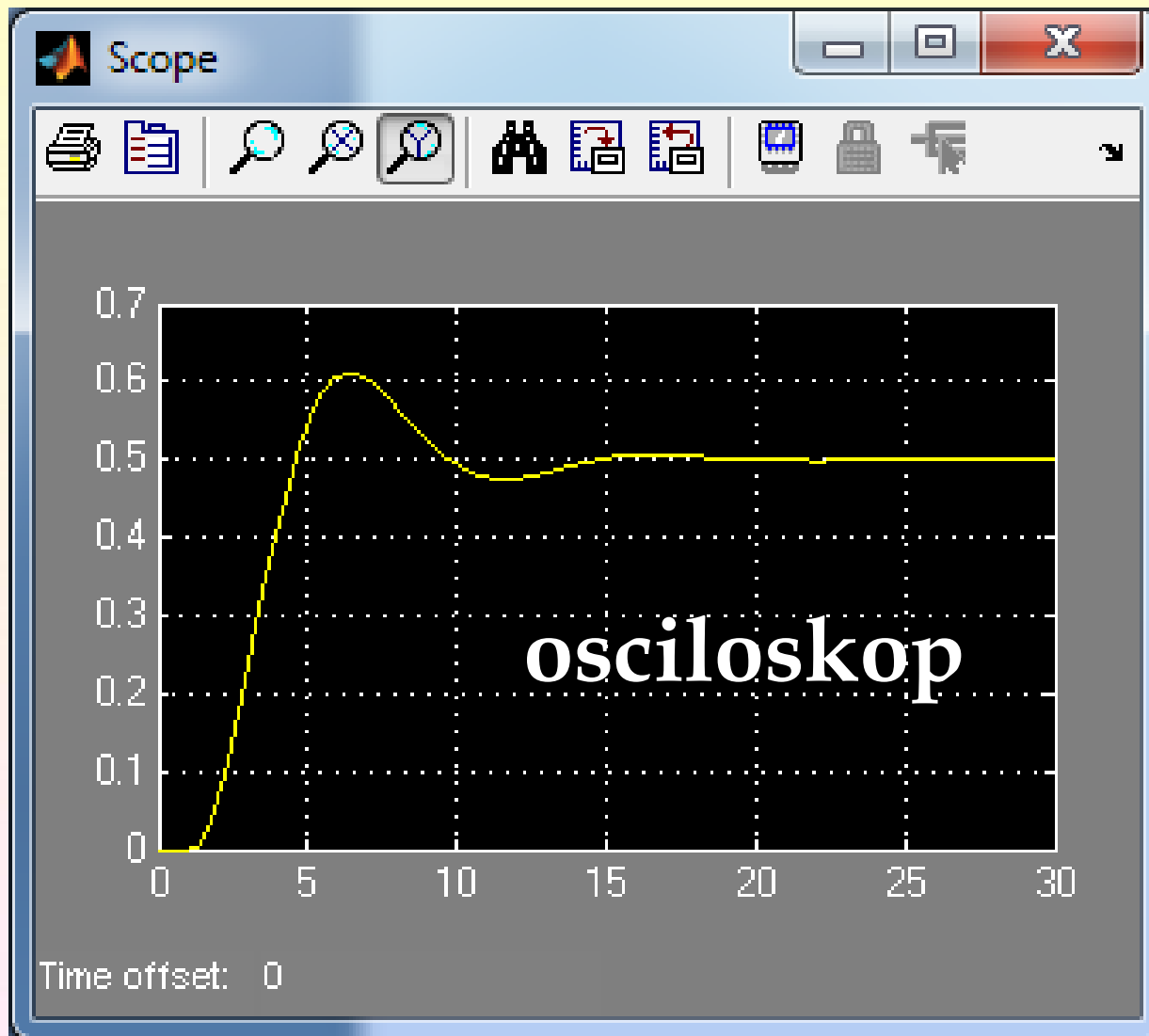
b0=1;a0=2;a1=3;a2=5;a3=1;

???

SIMULINK



SIMULINK



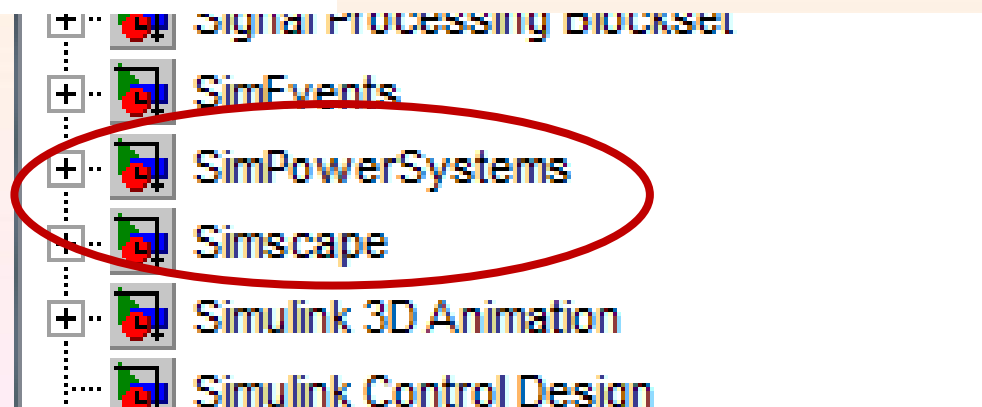
Existuje ještě jeden způsob jak
se to vše dá realizovat,
to bude váš DÚ – *povinný*!!!

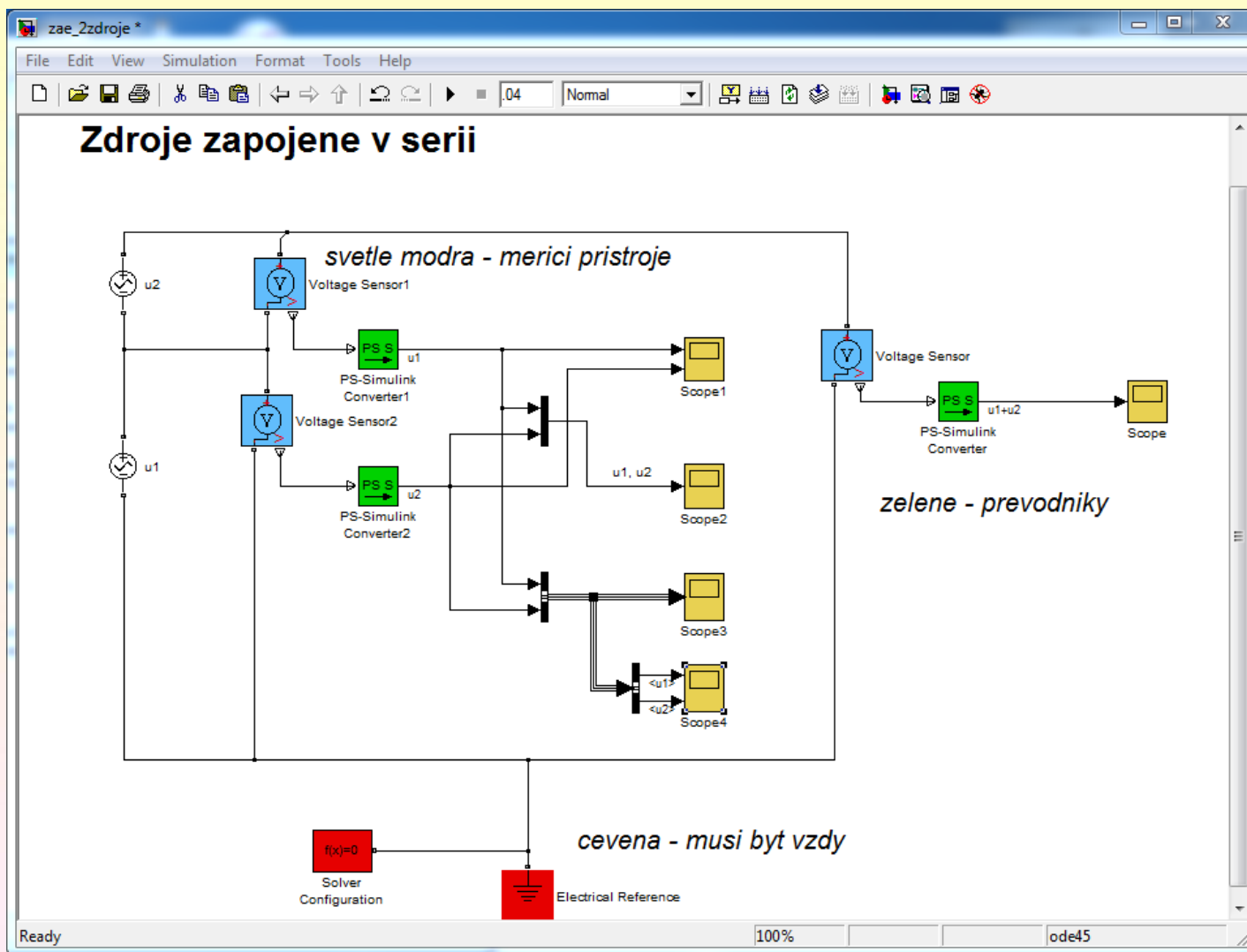
bude napsán a nakreslen ručně
- KAŽDÝM STUDENTEM -

Elektronické obvody

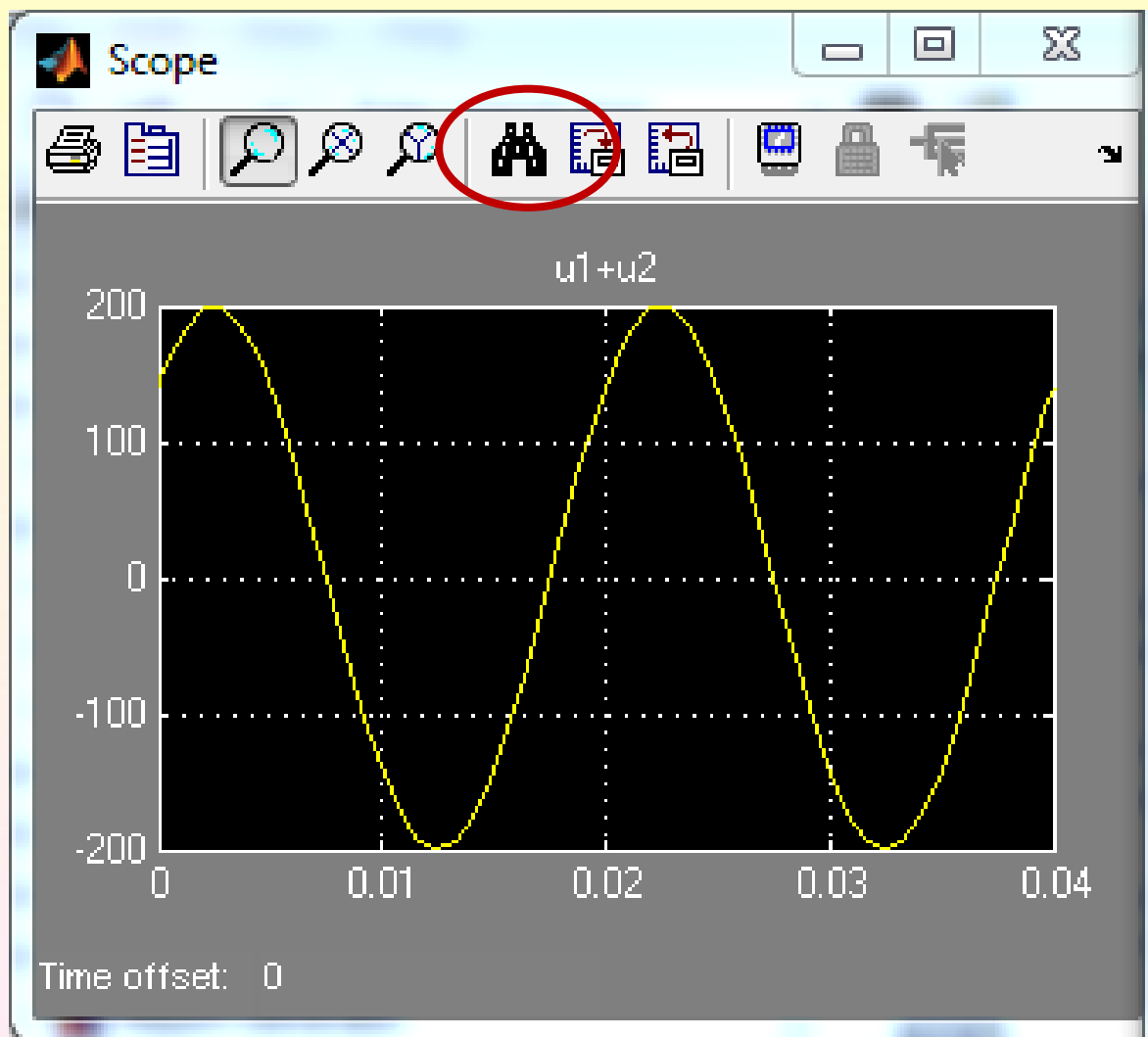
Kde to najdu?

Elektrické obvody

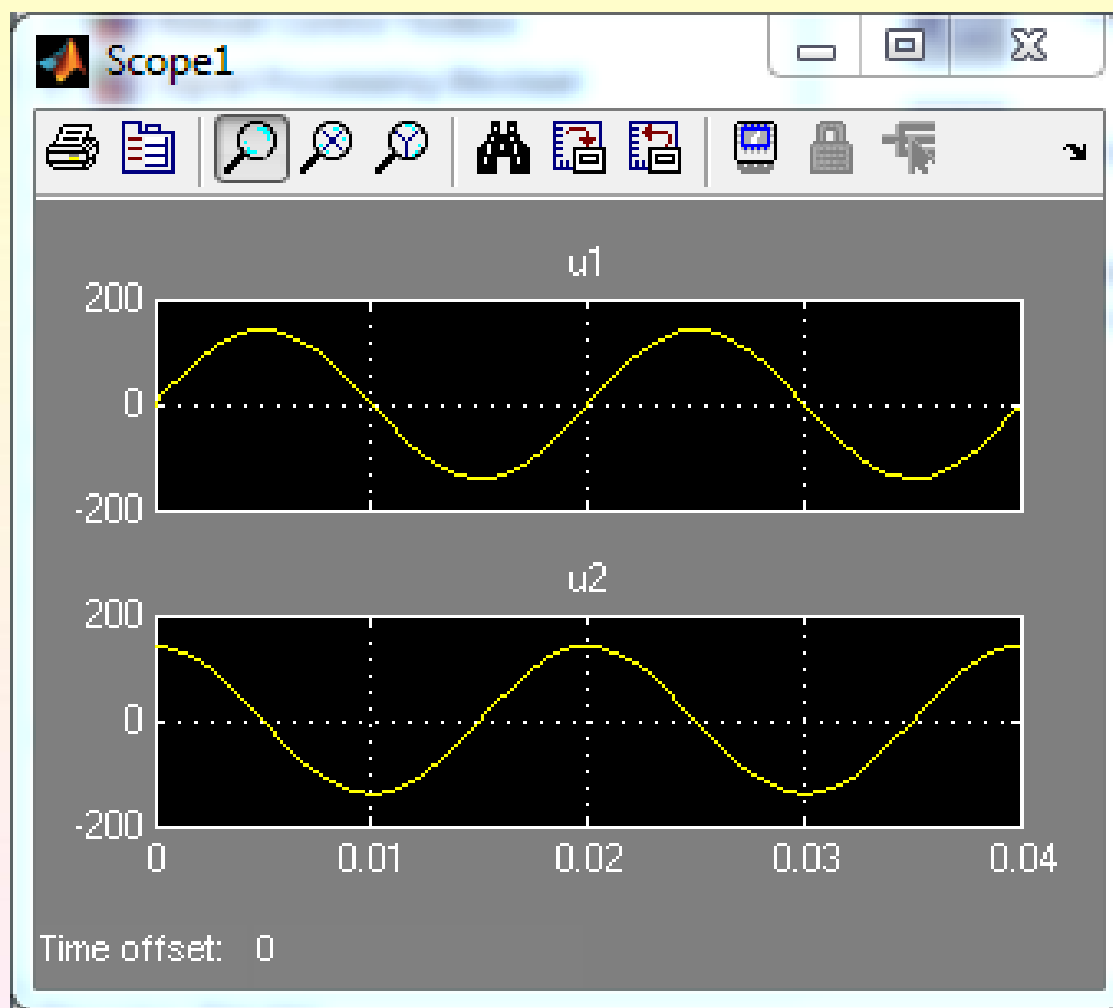




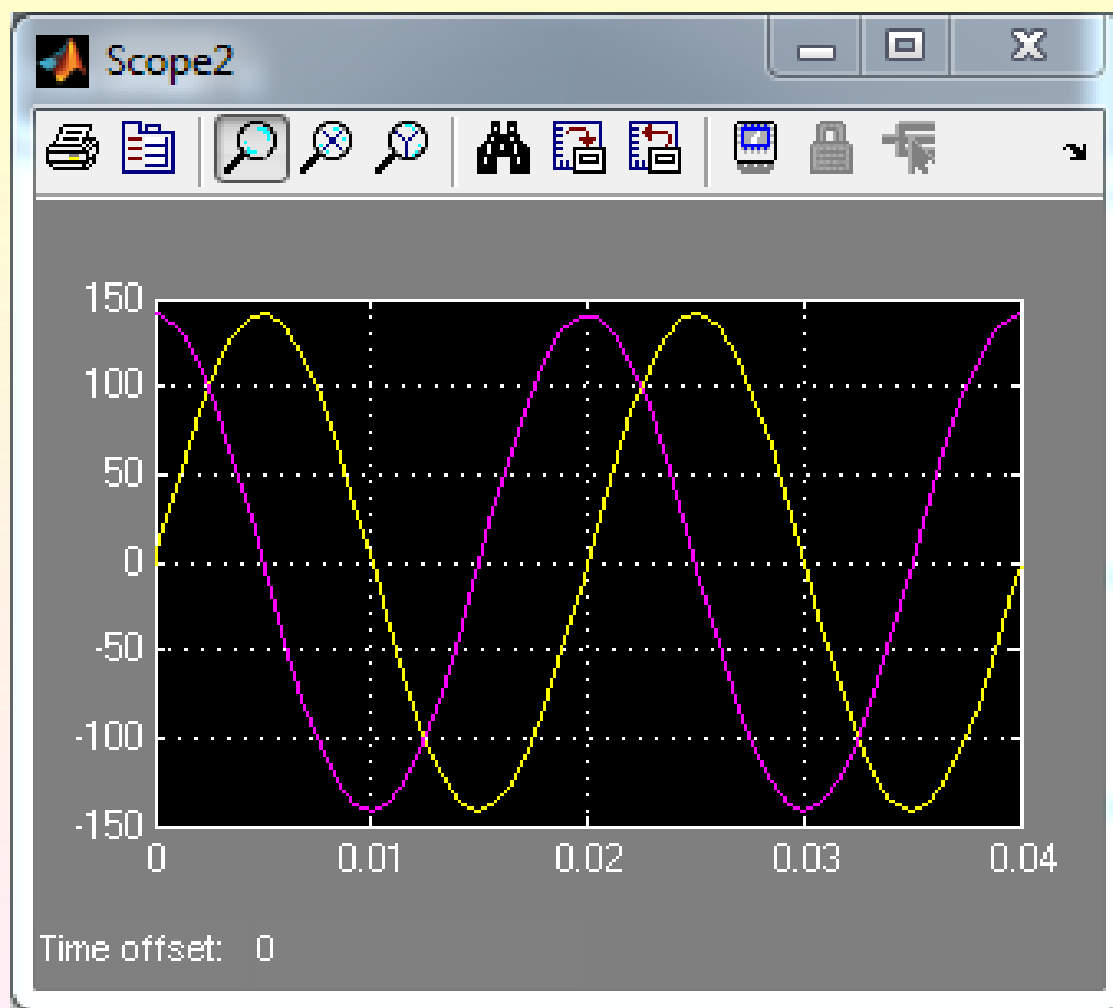
SIMULINK



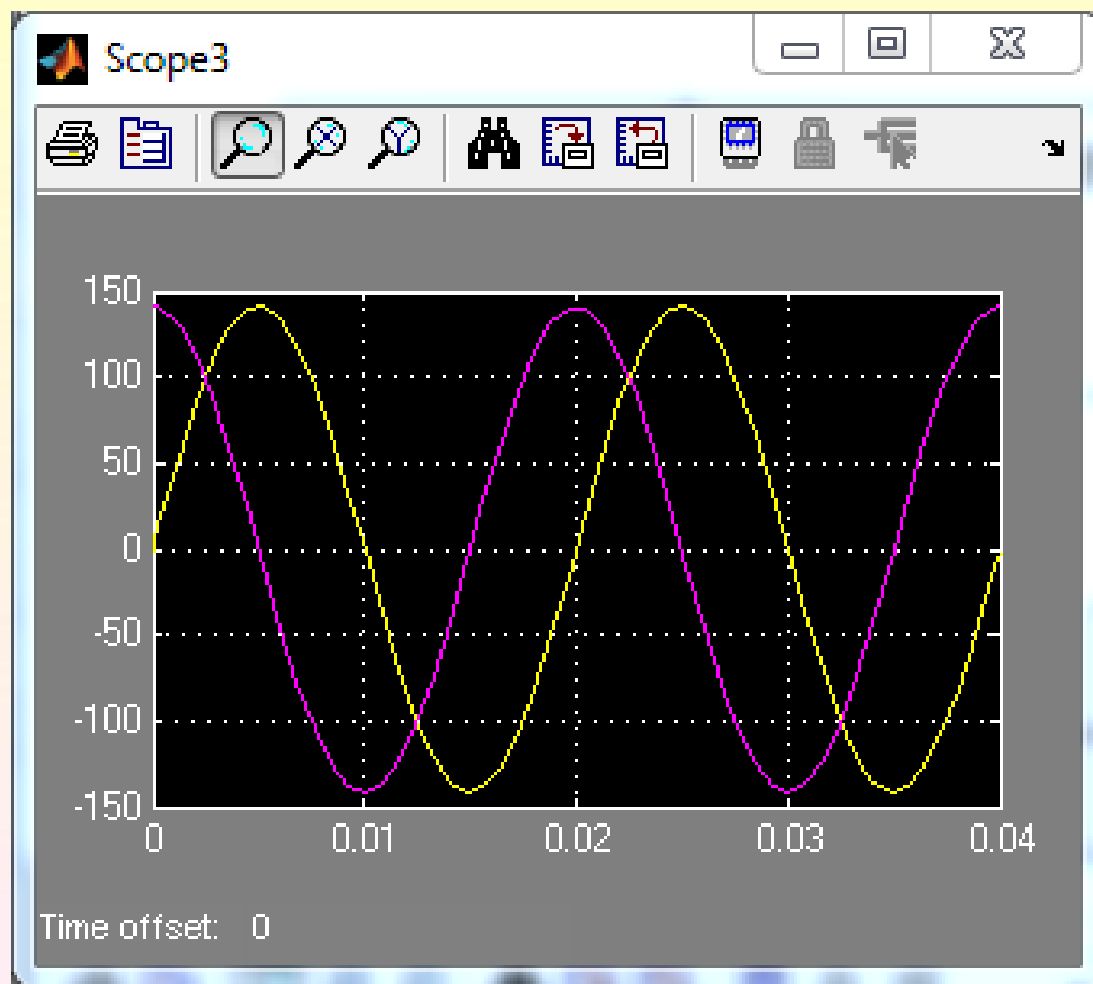
SIMULINK



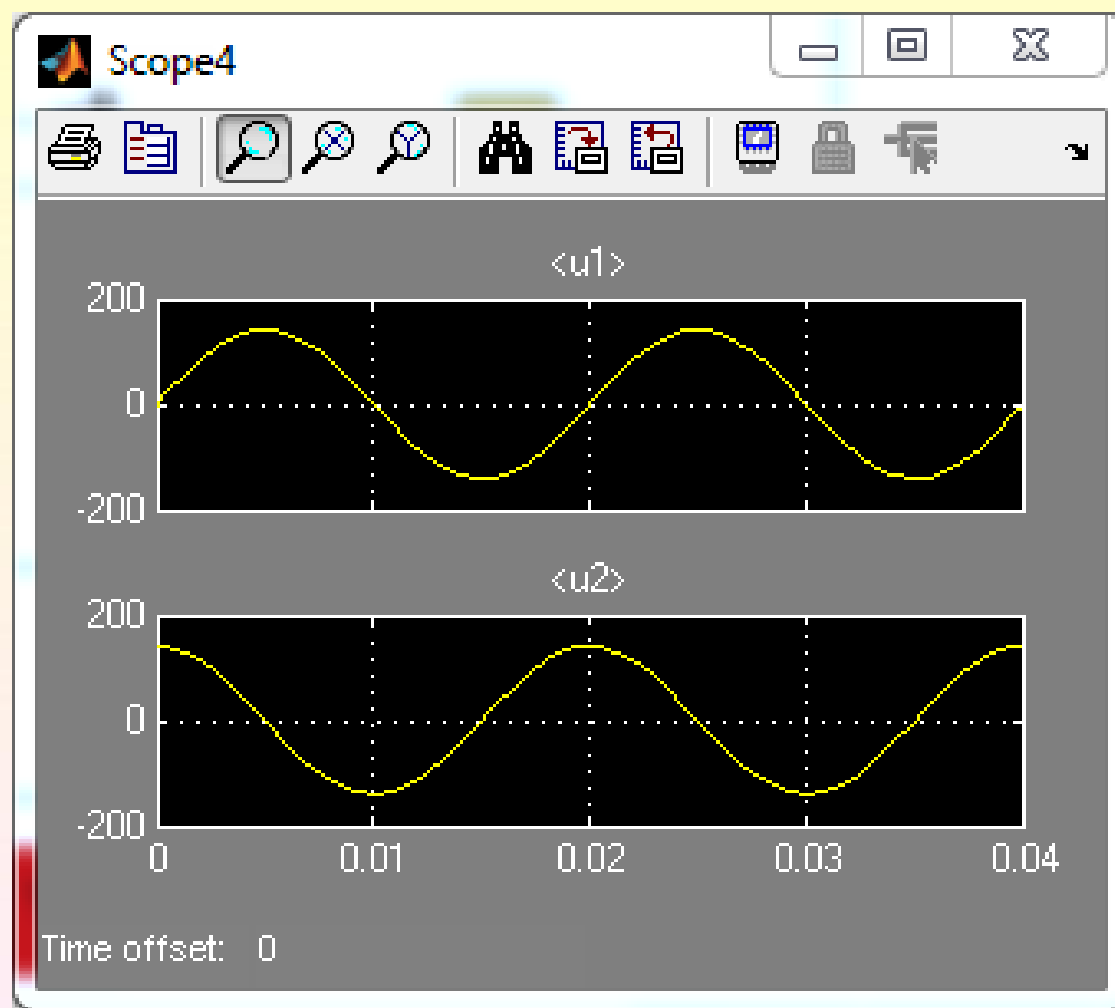
SIMULINK



SIMULINK



SIMULINK



SIMULINK

a to nejdůležitější
zase nyní,
nikdy nezapomeň
na

HELPOVNÍK

to když už nebudeš vědět kudy kam

a už je zase
konec roku
a někteří
se těší
na maturitu

SEZNAM PŘÍKLADŮ

Př. 1 – komplexní čísla

Př. 2 – převod mezi stupni a radiány a naopak

Př. 3 – tvorba sloupcových a řádkových matic

Př. 4 – vytvoření jednotkové matice

Př. 4 – grafické znázornění jednotkové matice

Př. 5 – vytvoření nulové matice

Př. 6 – vytvoření matice s **1** v hlavní diagonále

Př. 7 – sčítání a odčítání s maticemi

Př. 7 – matice **+** konstanta

Př. 7 – násobení matic

Př. 7 – dělení matic 1

Př. 7 – dělení matic 2

Př. 7 – mocniny s maticemi

SEZNAM PŘÍKLADŮ

Př. 7 – řešení systému rovni – o třech neznámých

Př. 7 – využití v el. technice

Př. 8 – determinant a inverzní matice

Př. 8 – rozměry matice - determinantu

Př. 9 – minimální hodnota matice

Př. 9 – maximální hodnota a medián matice

Př. 9 – směrodatná odchylka a rozptyl

Př. 9 – definice směrodatné odchylky

Př. 9 – třídění matic

Př. 10 – datové struktury

Př. 11 – převody řetězců 1

Př. 11 – převody řetězců 2

Př. 11 – převody řetězců 3

SEZNAM PŘÍKLADŮ

Př. 11 – převody řetězců 4

Použitá literatura

skripta ze školení **ML-1,2, Simulink** –
fy Humusoft

www.humusoft.cz:

Kupka, L.: Matlab, Simulink – úvod do
používání

helpovník **MatLabu**:

PŘÍKLAD 1

[zpět](#)

Zadání:

Proveď základní matematické operace s komplexními čísly – součet, rozdíl, podíl, násobení, mocninu a dále zjisti absolutní hodnu, číslo komplexně sdružené, reálnou a imaginární složku a je-li číslo reálné a fázový posun.

$$Z1=2-j, Z2=-3+2j.$$

Řešení:

1. Vytvoř proměnné $Z1$ a $Z2$.
2. Potom s nimi proveď příslušné matematické operace.



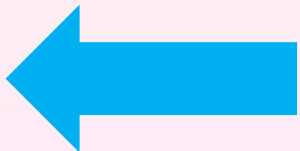
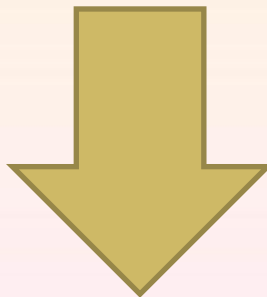
PŘÍKLAD 1

[zpět](#)

Používané příkazy:

$+$ $-$ $*$ $/$ $^$

To již bude provádět v ML a proto si ho
zavoláme.



PŘÍKLAD 2

[zpět](#)

Zadání:

Vyzkoušíme si nyní goniometrické fce. Budeme zkoušet jak počítání v radiánech, tak ve stupních. Nezapomeneme ani na inverzní fce. **Ukážeme si převod ze stupňů na radiány a naopak.**

Řešení:

$\text{rad}/180^\circ$ stupeň a opačně $\text{stupeň}/180^\circ$ rad

Příkazy:

$\sin(1)$, $\text{sin}(57)$, $\text{asin}(.5)$, ...

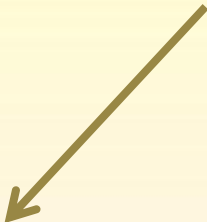
PŘÍKLAD 3

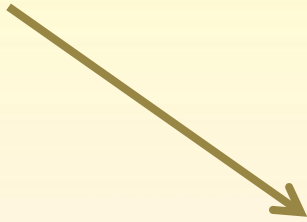
[zpět](#)

Zadání:

Vytvoř sloupcovou a řádkovou matici

Řešení:


$$B = [1; 2; 3]$$


$$A = [1 \ 2 \ 3]$$

$$B =$$

$$A =$$

Výsledek:

1
2
3

1 2 3

PŘÍKLAD 4

[zpět](#)

Zadání:

Vytvoř matice **ones(a)** a **ones(a,b,...)**.

Řešení:

ones(3)

ones(2,3,2)

Výsledek:

ans =

1	1	1
1	1	1
1	1	1

ans(:,:,1) =

1	1	1
1	1	1

ans(:,:,2) =

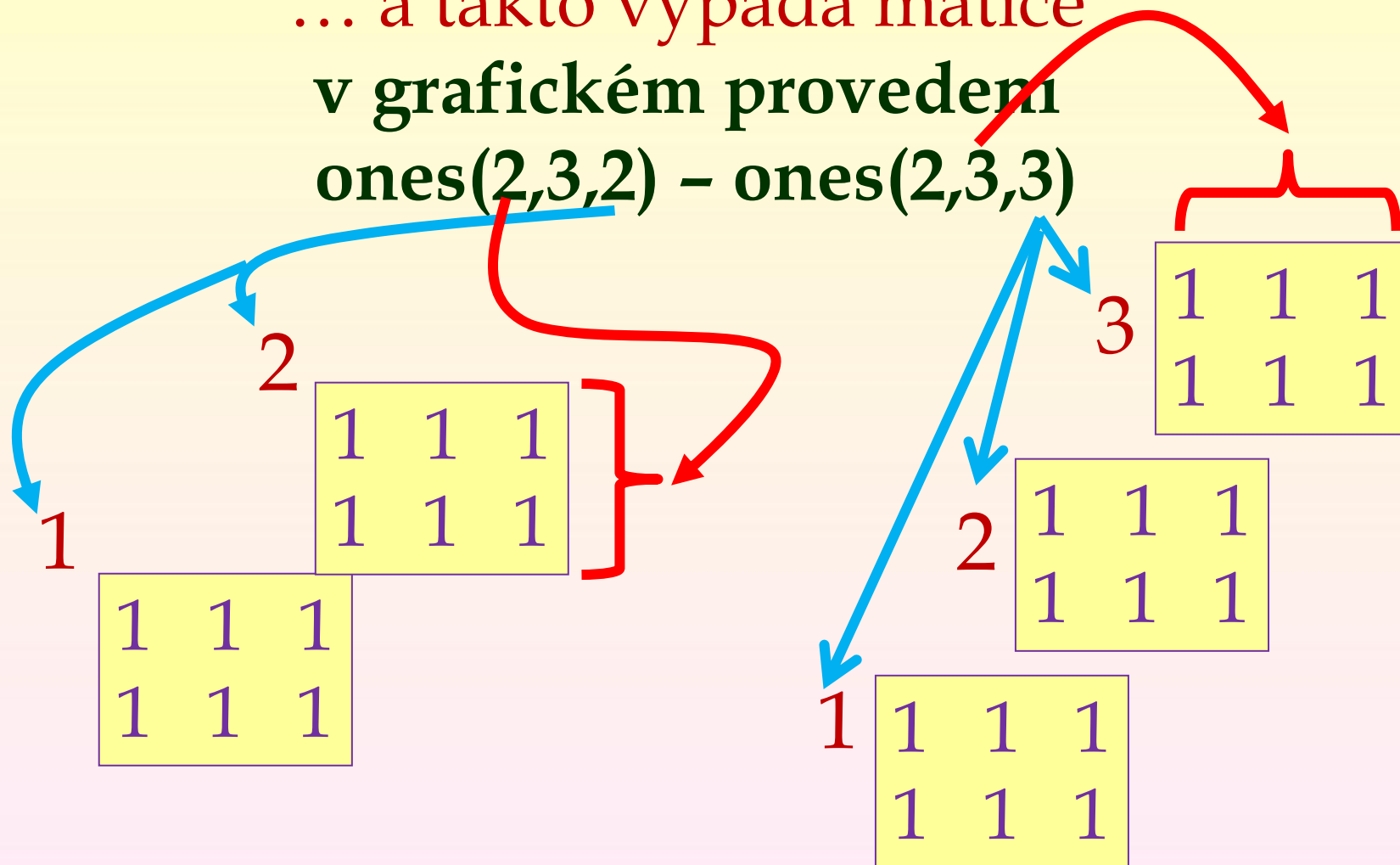
1	1	1
1	1	1

Chceš vidět jak
vypadá matice
ones(2,3,2)

PŘÍKLAD 4

[zpět](#)

... a takto vypadá matice
v grafickém provedení
 $\text{ones}(2,3,2) - \text{ones}(2,3,3)$



PŘÍKLAD 5

[zpět](#)

Zadání:

Vytvoř matici **zeros(a)**.

Řešení:



zeros(3)

Výsledek: **ans =**

0	0	0
0	0	0
0	0	0

PŘÍKLAD 6

[zpět](#)

Zadání:

Vytvoř matici **eye(a)**.

Řešení:


eye(3)

Výsledek:

ans =

1	0	0
0	1	0
0	0	1

*... a eye(a,b,c, ..)
má obdobné řešení
jako zeros(a,b,c,...)*

PŘÍKLAD 7

[zpět](#)

Zadání:

Sečti a odečti matice

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 7 & 4 & 1 \\ 9 & 5 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 2 & 1 \\ 7 & 4 & 1 \\ -6 & 5 & 1 \end{bmatrix}$$

Řešení:

$A+A$

$A-C$

Výsledek:

ans =

2	4	6
14	8	2
18	10	2

ans =

0	0	2
0	0	0
15	0	0

PŘÍČTENÍ KONSTANTY

*... pozor na
pravidla platící
pro matice ...*

PŘÍKLAD 7

[zpět](#)

Zadání:

Připočti konstantu $A = \begin{bmatrix} 1 & 2 & 3 \\ 7 & 4 & 1 \\ 9 & 5 & 1 \end{bmatrix}$

Řešení:

$$A + 3$$

Výsledek:

ans =

4	5	6
10	7	4
12	8	4

SOUČET - ROZDÍL

NÁSOBENÍ

PŘÍKLAD 7

[zpět](#)

Zadání:

Násob matice $A=[1\ 2\ 3;7\ 4\ 1;9\ 5\ 1]$, $B=[1;2;3]$

Řešení:

 $A*B$ $A*2$

Výsledek:

 $\text{ans} =$

14

18

22

 $\text{ans} =$

2 4 6

14 8 2

18 10 2

PŘÍČTENÍ KONSTANTY

DĚLENÍ

PŘÍKLAD 7

[zpět](#)

Zadání:

Děl matice $D=[1 \ 2; 2 \ 3]; E=[1; 2]$

$C=[1 \ 2 \ 1; 7 \ 4 \ 1; -6 \ 5 \ 1]; A=[1 \ 2 \ 3; 7 \ 4 \ 1; 9 \ 5 \ 1]$

Řešení:

$D \setminus E$

$D ./ 2$

Výsledek:

ans =

1

0

ans =

0.5000 1.0000

1.0000 1.5000

NÁSOBENÍ

DĚLENÍ 2

PŘÍKLAD 7

[zpět](#)

Zadání:

Děl matice $C=[1 \ 2 \ 1; 7 \ 4 \ 1; -6 \ 5 \ 1]$; $A=[1 \ 2 \ 3; 7 \ 4 \ 1; 9 \ 5 \ 1]$

Řešení:

A/C

Výsledek:

ans =

4.6875	-1.0625	-0.6250
0	1.0000	0
-0.4688	1.4063	0.0625

DĚLENÍ 1

UMOCŇOVÁNÍ

PŘÍKLAD 7

[zpět](#)

Zadání:

Umocni matici $A = \begin{bmatrix} 1 & 2 & 3 \\ 7 & 4 & 1 \\ 9 & 5 & 1 \end{bmatrix}$

Řešení:

A^2

$D.^2$

Výsledek:

ans =

42	25	8
44	35	26
53	43	33

ans =

1	4	9
49	16	1
81	25	1

DĚLENÍ 2

SYSTÉM ROVNIC

PŘÍKLAD 7

[zpět](#)

Zadání:

Vyřeš systém tří rovnic o třech neznámých.

$$3x-2y+z=2$$

$$x+y+z=6$$

$$-x-y+z=0$$

Řešení:

Systém rovnic upravíme na tři matice.

$$K \cdot X = B$$

K jsou koeficienty u neznámých **x**, **y** a **z**. **X** jsou
Neznáme **x**, **y** a **z**. **B** je pravá strana rovnic

PŘÍKLAD 7

[zpět](#)

Řešení:

$$K = \begin{vmatrix} 3 & -2 & 1 \\ 1 & 1 & 1 \\ -1 & -1 & 1 \end{vmatrix} \quad X = \begin{vmatrix} x \\ y \\ z \end{vmatrix} \quad B = \begin{vmatrix} 2 \\ 6 \\ 0 \end{vmatrix}$$

Výsledek:

$$X = \text{inv}(K) * B$$

$$X =$$

1.0000

2.0000

3.0000

ZADÁNÍ

PŘ. ze ZAE – el. obv.

PŘÍKLAD 7

[zpět](#)

Zadání:

Vyřeš el. obv. s pěti smyčkami. el. obv. je pro ST proud. Nakresli jeho schéma, sestav systém rovnic a vypočítej jednotlivé proudy.

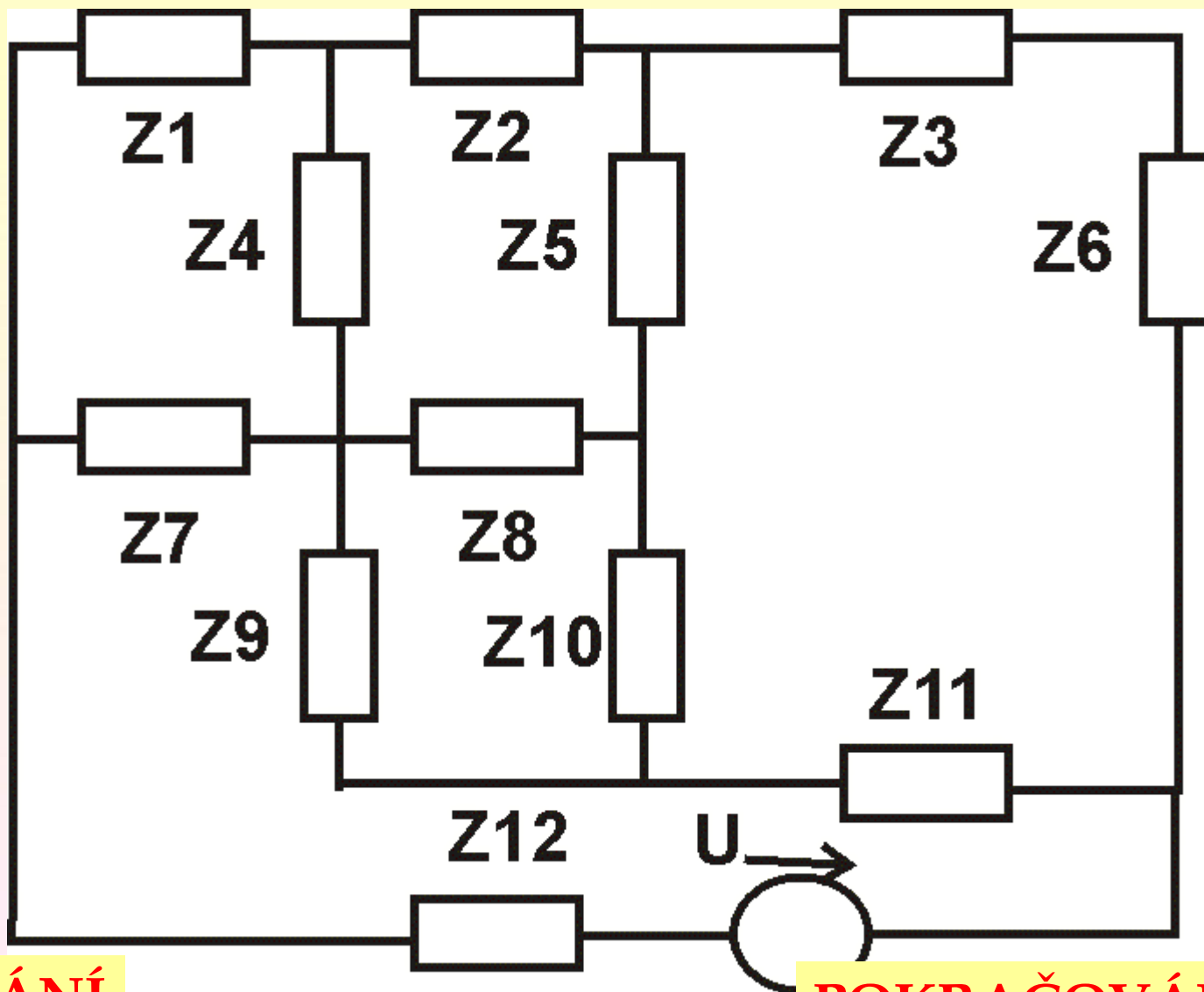
smyčky

- 
1. $Z1, Z4, Z7$
 2. $Z2, Z5, Z4, Z8$
 3. $Z3, Z6, Z11, Z10, Z5$
 4. $Z8, Z10, Z9$
 5. $Z7, Z9, Z11, Z12, U$

PŘÍKLAD 7

[zpět](#)

S
C
H
É
M
A



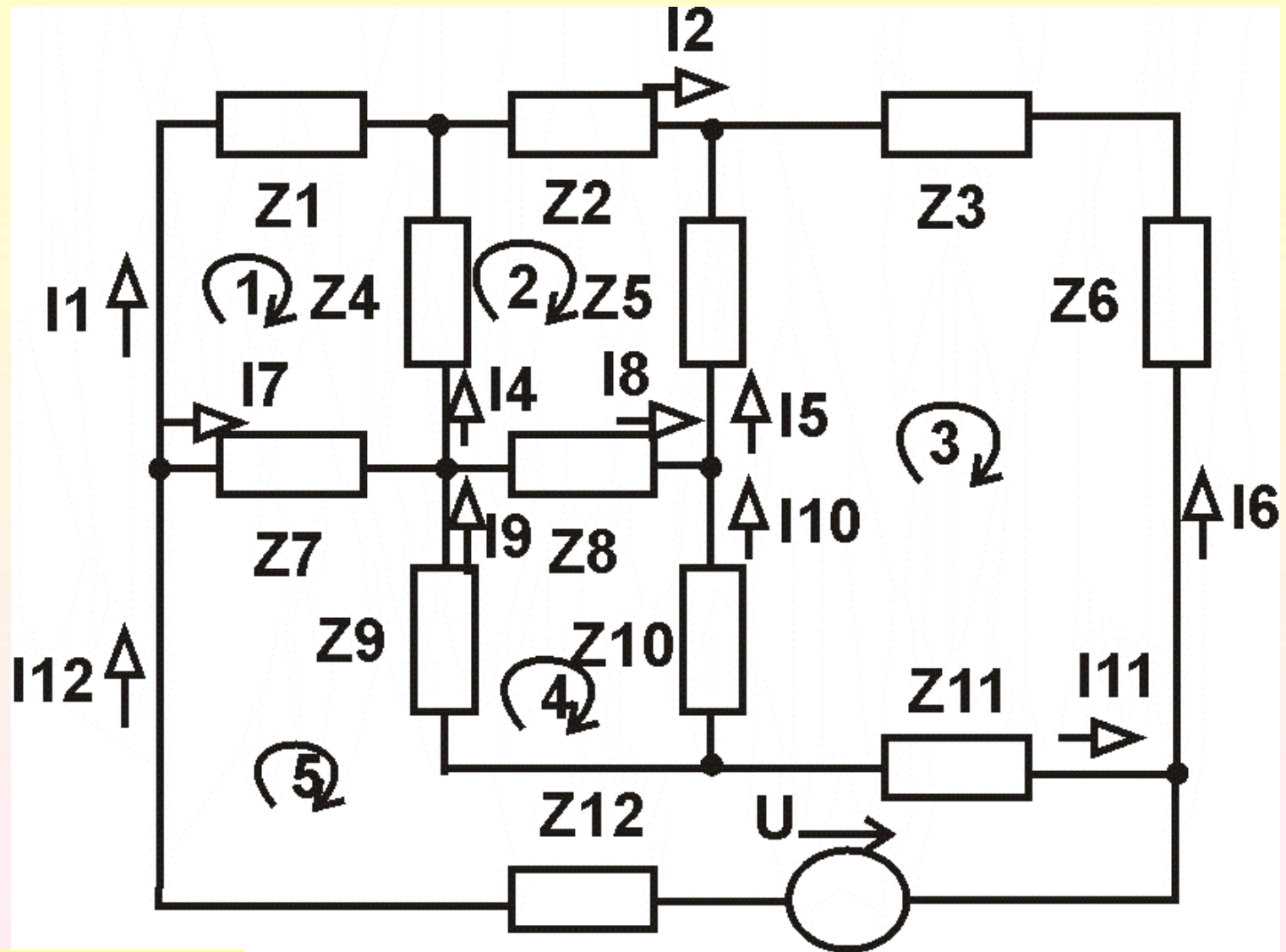
ZADÁNÍ

POKRAČOVÁNÍ

PŘÍKLAD 7

[zpět](#)

S
C
H
É
M
A



PŘEDCHOZÍ

POKRAČOVÁNÍ

PŘÍKLAD 7

[zpět](#)

Rovnice jednotlivých smyček

$$Z_1 I_a + Z_4 (I_a - I_b) + Z_7 (I_a - I_e) = 0$$

$$Z_2 I_b + Z_5 (I_b - I_c) + Z_8 (I_b - I_d) + Z_4 (I_b - I_a) = 0$$

$$(Z_3 + Z_6) I_c + Z_{11} (I_c - I_e) + Z_{10} (I_c - I_d) + Z_5 (I_c - I_b) = 0$$

$$Z_8 (I_d - I_b) + Z_{10} (I_b - I_c) + Z_9 (I_d - I_e) = 0$$

$$Z_7 (I_e - I_a) + Z_9 (I_e - I_d) + Z_{11} (I_e - I_c) + Z_{12} I_e = U$$

PŘEDCHOZÍ

POKRAČOVÁNÍ

PŘÍKLAD 7

[zpět](#)

Roznásobení rovnic

$$Z_1 I_a + Z_4 I_a - Z_4 I_b + Z_7 I_a - Z_7 I_e = 0$$

$$Z_2 I_b + Z_5 I_b - Z_5 I_c + Z_8 I_b - Z_8 I_d + Z_4 I_b - Z_4 I_a = 0$$

$$Z_3 I_c + Z_6 I_c + Z_{11} I_c - Z_{11} I_e + Z_{10} I_c - Z_{10} I_d + Z_5 I_c - Z_5 I_b = 0$$

$$Z_8 I_d - Z_8 I_b + Z_{10} I_d - Z_{10} I_c + Z_9 I_d - Z_9 I_e = 0$$

$$Z_7 I_e - Z_7 I_a + Z_9 I_e - Z_9 I_d + Z_{11} I_e - Z_{11} I_c + Z_{12} I_e = U$$

PŘEDCHOZÍ

POKRAČOVÁNÍ

PŘÍKLAD 7

[zpět](#)

Upravené rovnice dle proudů

$$(Z_1 + Z_4 + Z_7)I_a - Z_4I_b - Z_7I_e = 0$$

$$-Z_4I_a + (Z_2 + Z_4 + Z_5 + Z_8)I_b - Z_5I_c - Z_8I_d = 0$$

$$-Z_5I_b + (Z_3 + Z_5 + Z_6 + Z_{10} + Z_{11})I_c - Z_{10}I_d - Z_{11}I_e = 0$$

$$-Z_8I_b - Z_{10}I_c + (Z_8 + Z_9 + Z_{10})I_d - Z_9I_e = 0$$

$$-Z_7I_a - Z_{11}I_c - Z_9I_d + (Z_7 + Z_9 + Z_{11} + Z_{12})I_e = U$$

PŘEDCHOZÍ

POKRAČOVÁNÍ

PŘÍKLAD 7

[zpět](#)

Řešení pomocí matic

$$Z \cdot I = U$$

$$I = \begin{bmatrix} I_a \\ I_b \\ I_c \\ I_d \\ I_e \end{bmatrix} \quad U = \begin{bmatrix} U_a \\ U_b \\ U_c \\ U_d \\ U_e \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ U \end{bmatrix}$$

PŘEDCHOZÍ

POKRAČOVÁNÍ

PŘÍKLAD 7

[zpět](#)

Řešení pomocí matic

$$\mathbf{Z} = \begin{vmatrix} \mathbf{Z}_{33} & \mathbf{Z}_{32} \\ \mathbf{Z}_{23} & \mathbf{Z}_{22} \end{vmatrix}$$

[PŘEDCHOZÍ](#)

[POKRAČOVÁNÍ](#)

PŘÍKLAD 7

[zpět](#)

Řešení pomocí matic

Z_{33}

$$\begin{pmatrix} (Z_1 + Z_4 + Z_7) & -Z_4 & 0 \\ -Z_4 & (Z_2 + Z_4 + Z_5 + Z_8) & -Z_5 \\ 0 & -Z_5 & (Z_3 + Z_5 + Z_6 + Z_{10} + Z_{11}) \end{pmatrix}$$

PŘÍKLAD 7

[zpět](#)

Řešení pomocí matic

$$Z_{22}$$

$$\begin{pmatrix} Z_8 + Z_9 + Z_{10} & -Z_9 \\ -Z_9 & (Z_7 + Z_9 + Z_{11} + Z_{12}) \end{pmatrix}$$

k PŘEDCHOZÍ

POKRAČOVÁNÍ

PŘÍKLAD 7

[zpět](#)

Řešení pomocí matic

$$Z_{32}$$

$$0$$

$$Z_7$$

$$-Z_8$$

$$0$$

$$-Z_{10}$$

$$Z_{11}$$

[PŘEDCHOZÍ](#)

[POKRAČOVÁNÍ](#)

PŘÍKLAD 7

[zpět](#)

Řešení pomocí matic

$$Z_{23}$$

$$0$$

$$-Z_8$$

$$-Z_{10}$$

$$Z_7$$

$$0$$

$$-Z_{11}$$

PŘEDCHOZÍ

POKRAČOVÁNÍ

PŘÍKLAD 7

[zpět](#)

Řešení pomocí matic

PŘEDCHOZÍ

POKRAČOVÁNÍ

PŘÍKLAD 8

[zpět](#)

Zadání:

Vytvoř determinant a inverzní matici

$\det(C)$ a $\text{inv}(C)$ - $C = \begin{bmatrix} 1 & 2 & 1 \\ 7 & 4 & 1 \\ -6 & 5 & 1 \end{bmatrix}$.

Řešení:

$\text{inv}(C)$

$\det(C)$

Výsledek:

*... a vypočítej si
ho „ručně“*

ans =

ans =

32

-0.0312 0.0937 -0.0625

-0.4063 0.2188 0.1875

1.8438 -0.5313 -0.3125

ROZMĚRY MATICE

PŘÍKLAD 8

[zpět](#)

Zadání:

`length(A), size(A)`

Zjisti rozměry matice **`length(F), size(F)`**

- `F=[1 2 1;7 4 1]`.

Řešení:

`length(F)`

`size(F)`

Výsledek:

`ans =`
`3`

`ans =`
`2 3`

PŘÍKLAD 9

[zpět](#)

Zadání:

Zjisti minimální hodnotu z matice **A** a **A'**.

$A = [1 \ 2 \ 3; 7 \ 4 \ 1; 9 \ 5 \ 1]$

Řešení:

$\min(A)$

$\min(A')$

Výsledek:

*... a zjisti z čeho
ty hodnoty jsou
(z ř. nebo s.)*

ans =

1 2 1

ans =

1 1 1

MAXIMÁLNÍ HODNOTA

PŘÍKLAD 9

[zpět](#)

Zadání:

Zjisti maximální hodnotu a medián z matice **A**.

$A = [1 \ 2 \ 3; 7 \ 4 \ 1; 9 \ 5 \ 1]$

Řešení:

$\max(A)$

$\text{median}(A)$

Výsledek:

$\text{ans} =$

9 5 3

$\text{ans} =$

7 4 1

MINIMÁLNÍ HODNOTA

STD

PŘÍKLAD 9

[zpět](#)

Zadání:

Zjisti směrodatnou odchylku a rozptyl z matice **A**.

$A = [1 \ 2 \ 3; 7 \ 4 \ 1; 9 \ 5 \ 1]$

Řešení:

$SO = \text{std}(A)$

$R = SO.^2$

Výsledek:

ans =
4.1633 1.5275 1.1547

ans =
17.3333 2.3333 1.3333

PŘÍKLAD 9

[zpět](#)

Zadání:

Definování směrodatné odchylky.

Řešení:

$S = \text{std}(A, \text{flag})$ flag nabývá hodnoty **0** nebo **1**


$$s = \left(\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2 \right)^{1/2}$$

populační STD


$$s = \left(\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2 \right)^{1/2}$$

STD

TŘÍDĚNÍ

PŘÍKLAD 9

[zpět](#)

Zadání:

Seříd' matici.

$Q=[2\ 5\ 4\ 1\ 3\ 9\ 7]$

Řešení:

$[\text{hodnota}\ \text{index}]=\text{sort}(Q)$

Výsledek:

hodnota =

1 2 3 4 5 7 9

index =

4 1 5 3 2 7 6

PŘÍKLAD 10

[zpět](#)

Zadání:

Tvorba datové struktury – např. pro **Kruh**.

Řešení:

```
Kruh.r=5;Kruh.x=10;Kruh.y=-3;Kruh
```

Výsledek:

```
Kruh =
```

```
  r: 5
```

```
  x: 10
```

```
  y: -3
```

... a plocha kruhu ? ...

```
Kruh.r^2*pi()
```

```
ans =
```

```
78.5398
```

PŘÍKLAD 11

[zpět](#)

Zadání:

Převody řetězců a textů a zjisti co tyto příkazy ve skutečnosti dělají.

Řešení:

```
r='Karel';abs(r)
```

```
char(r)
```

Výsledek:

```
ans =
```

```
75  97 114 101 108
```

```
ans =  
Karel
```

POKRAČOVÁNÍ

PŘÍKLAD 11

[zpět](#)

Zadání:

Převody řetězců a textů a zjisti co tyto příkazy ve skutečnosti dělají.

Řešení:

`num2str(num)`

`int2str(int)`

Výsledek:

`int2str(123)`

`ans =`

123

`>> 5+ans`

`ans =`

54 55 56

`>> 123+5`

`ans =`

128

`num2str(3.24)`

`ans =`

3.24

`>> ans+0`

`ans =`

51 46 50

52

`>> ans+1`

`ans =`

52 47 51 53

PŘEDCHOZÍ

DALŠÍ

PŘÍKLAD 11

[zpět](#)

Zadání:

Převody řetězců
a textů a zjisti co tyto
Příkazy ve skutečnosti
dělají.

Řešení:

`str2double(r)`

Výsledek:

```
>> r='12.36'
```

```
r =
```

```
12.36
```

```
>> r+1
```

```
ans =
```

```
50 51 47 52 55
```

```
>> str2double(r)
```

```
ans =
```

```
12.3600
```

```
>> ans+10
```

```
ans =
```

```
22.3600
```

[DALŠÍ](#)[PŘEDCHOZÍ](#)

PŘÍKLAD 11

[zpět](#)

Zadání:

Převody řetězců a textů a zjisti co tyto příkazy ve skutečnosti dělají.

Řešení:

```
str2num(r)
```

```
>> str2num(r)
```

```
ans =
```

```
12.3600
```

```
>> 10+ans
```

```
ans =
```

```
22.3600
```

Výsledek:

PŘEDCHOZÍ



V 12.7 – 230 listů