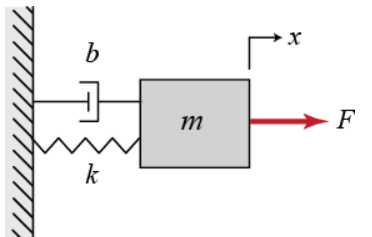
**Mass-Spring-Damper System**

**ZADANIE:**

* Uvažujte fyzikálno - matematický model dynamického systému, ktorý je popísaný lineárnou diferenciálnou rovnicou (LDR) 2. a vyššieho rádu.

**ÚLOHA:** Navrhnite m-file v simulačnom jazyku Matlab, ktorý umožní:

1. Zadefinovanie prenosovej funkcie opisujúcej LDS v s-oblasti (v polynomiálnom tvare, v tvare póly/nuly) a v stavovom priestore pomocou matíc A, B, C, D.
2. Konverziu modelov zo stavového priestoru do tvaru prenosovej funkcie a naopak.
3. Analýzu LTI DS v časovej (prechodová charakteristika, impulzná charakteristika, odozva na ľubovoľný vstupný signál) a frekvenčnej oblasti (Nyquistová, Nicholsová a Bodeho charakteristiky).
4. Vyhodnotenie stability uvažovaného LTI dynamického systému na základe získaných odoziev na rôzne typy vstupných signálov.

Parametre:

m [kg] – hmotnosť

k [N/m] – konštanta pružiny

b [Ns/m] v konštanta tlmenia

F [N] – vstupná sila

**Riešenie:**

1. Použitím laplaceovej transformácie získame diferenciálnu rovnicu v oblasti laplaceových obrazov:
2. Transfer function:
3. State Space:

Hodnoty parametrov pre výpočet v programovom prostredí MATLAB:

m = 1 kg

k = 1 N/m

b = 0.2 Ns/m

F = 1 N

**Riešenie menu v programovom prostredí MATLAB:**

**hlavny.m**

set(0, 'DefaultUIControlFontSize', 16);

options = menu('Main Menu', ...

'definovanie prenosovej funkcie ',' konverzia ','časové charakteristiky ',' frekvenčné charakteristiky ',' stabilita ',' koniec ');

% tvorba menu a jeho obsahu, prvý parameter je názov, ostatné možnosti (tlačidlá) menu

switch options

case 1, vstup

case 2, konverzia

case 3, casova

case 4, frekvencna

case 5, stabilita

case 6

end

Изображение выглядит как текст

Автоматически созданное описание

**vstup.m**

% PROGRAM NA ODSKUSANIE JEDNOTLIVYCH FUNKCII CONTROL TOOLBOXU PRE SYSTEM

% TELESO, PRUZINA, POTENCIOMETER, m=?, k=?, p=?

% 1. LDR v s-oblasti (v polynoilalnom tvare, v trave poly/nuly)

% a v stavovom priestore pomocou matic A, B, C, D.

function [sys] = vstup()

global sys a b c d;

options = menu("VSTUPY", ...

"V polynoilalnom tvare", "Pomocou Polov, Nul a Zosilnenia", "Pomocou matic");

switch options

case 1

num = input("Zadajte citatel ... num=");

den = input("Zadajte menovatel ... den=");

sys = tf(num, den);

disp(sys);

case 2

z = input("Zadajte nuly ... z=");

p = input("Zadajte poly ... p=");

k = input("Zadajte zosilnenie ... k=");

sys = zpk(z, p, k);

disp(sys);

case 3

a = input("Zadajte maticu A=");

b = input("Zadajte maticu B=");

c = input("Zadajte maticu C=");

d = input("Zadajte maticu D=");

sys = ss(a, b, c, d);

disp(sys);

end

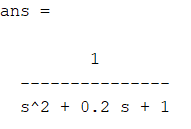
hlavny

Изображение выглядит как текст

Автоматически созданное описание

Zadaj čitateľa ... num = **1**

Zadaj menovateľa... den = **[1 0.2 1]**



Časové charakteristiky modelu otáčok motora:

Изображение выглядит как текст

Автоматически созданное описание

**casova.m**

% casova analyza

function [] = casova()

global sys;

options = menu("Casova oblast", 'step', 'impulse', "Odozva na lubovolny signal");

switch options

case 1

% Prechodova casova charakteristika

step(sys);

title("Prechodova casova charakteristika")

case 2

% Impulzna charakteristika

impulse(sys);

title("Impulzna charakteristika");

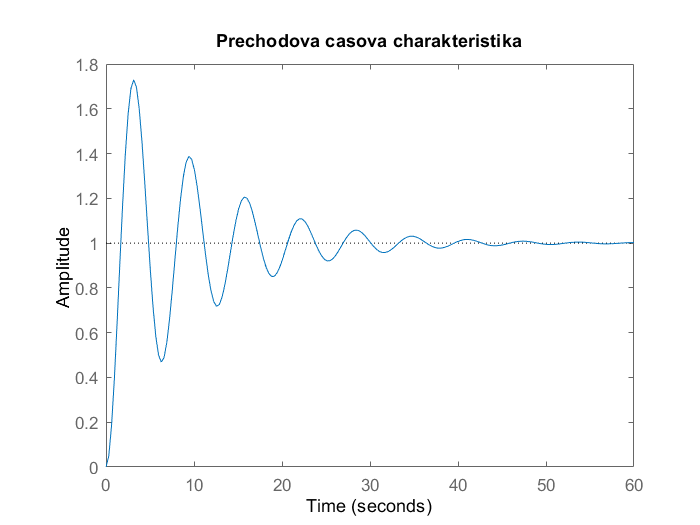
case 3

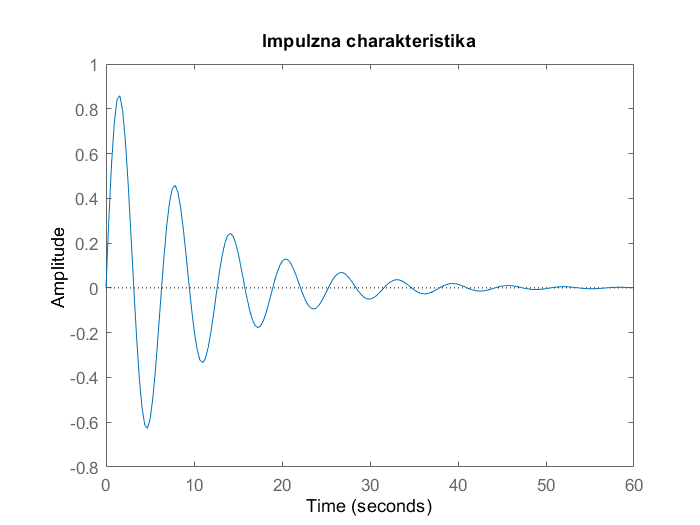
% Odozva na lubovolny signal

odozva

end

hlavny





Frekvenčné charakteristiky modelu otáčok motora:

**frekvencna.m**

function [] = frekvencna()

global sys;

options = menu("Frekvencna oblast", "Nyquistova", "Bodeho", "Nicholsonova");

switch options

case 1

% Nyquistova frekvencna charakteristika

nyquist(sys);

title("Nyquistova frekvencna charakteristika")

case 2

% Bodeho frekvencna charakteristika

bode(sys);

title("Bodeho frekvencna charakteristika")

case 3

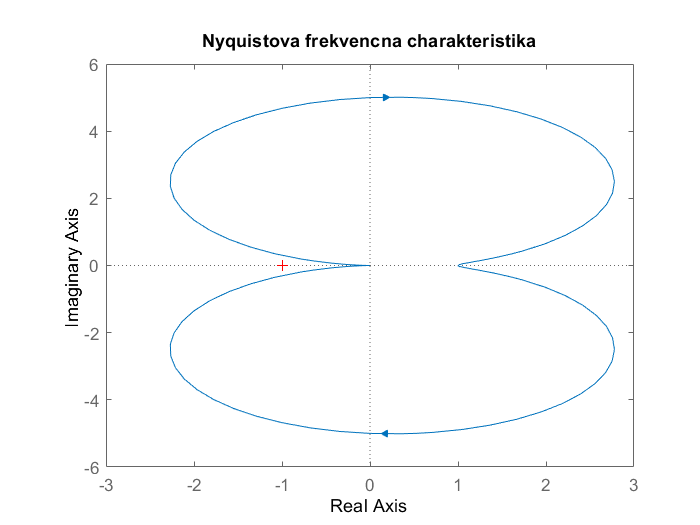
% Nicholsonova frekvencna charakteristika

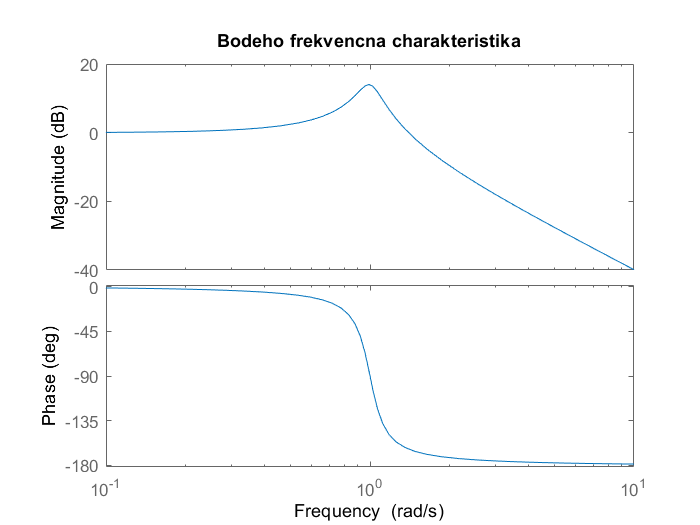
nichols(sys);

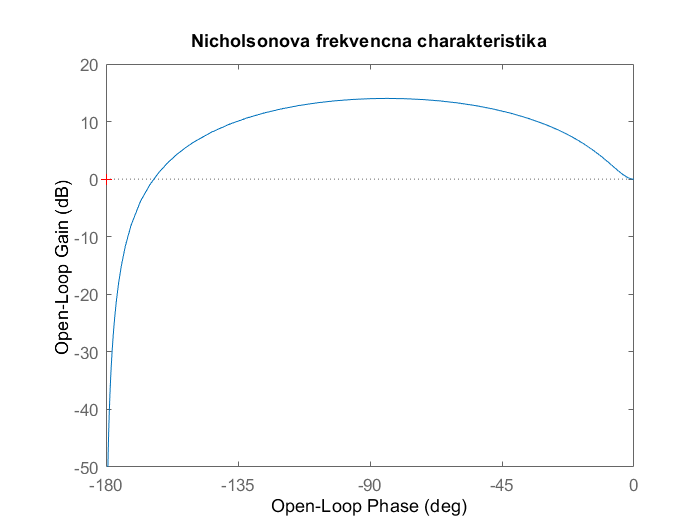
title("Nicholsonova frekvencna charakteristika")

end

hlavny







Odozva na ľubovoľný signál:

**odozva.m**

% Odozva na ľubovoľný signál

function [] = odozva()

global sys;

type = input("Zadaj typ signalu napr. 'sine', 'pulse', 'square': ");

Tau = input("zadaj periodu vzorkovania (5): ");

Tf = input("zadaj celkovu dobu simulacie (30): ");

Ts = input("zadaj vzorkovaci cas Ts (0.1): ");

[u, t] = gensig(type, Tau, Tf, Ts);

lsim(sys, u, t);

grid;

title("odozva na lubovolny vstupny signal")

xlabel("cas(t)"), ylabel("y(t)");

hlavny

return

Stabilita:

**stabilita.m**

% vyhodnotenie stability

function [] = stabilita()

global sys;

[num, den] = tfdata(sys, 'v');

r = roots(den);

max = size(r);

test = 1;

for a=1:max(:,1)

if r(a) > 0

test = 0;

end

end

if test == 0

disp("nestabilny")

else

disp("stabilny alebo na hranici")

end

hlavny

return