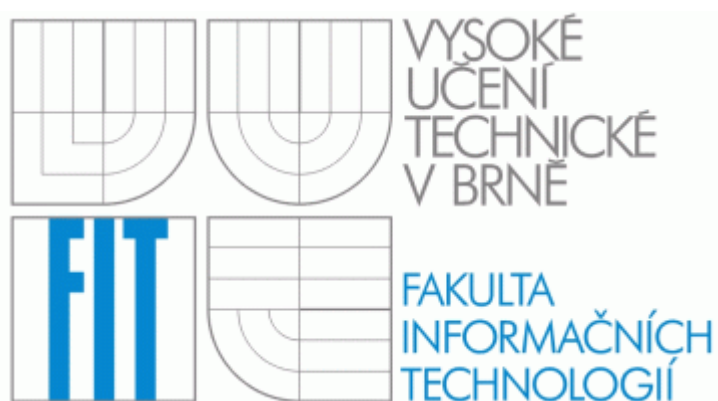


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



DIPLOMOVÁ PRÁCE

Prohlášení

Prohlašuji, že jsem tuto Diplomovou práci vypracoval samostatně pod vedením Doc. Ing. Jiřího Kunovského, CSc.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal poznatky pro zpracování této práce.

Poděkování

Chtěl bych poděkovat především Doc. Ing. Jiřímu Kunovskému, CSc. za jeho odborné vedení, dále za podnětné připomínky a za poskytnutou literaturu, kterou jsem využil pro tvorbu této diplomové práce. Dále také děkuji všem, kteří jakkoli pomohli, či přispěli k tomu, abych mohl tuto diplomovou práci vypracovat.

V Brně, 20. Ledna 2007

.....

Abstrakt

Tato diplomová práce se zabývá tematikou regulačních obvodů. V úvodu jsou základy teorie řízení a regulace - projekt je rozdělen do dvou základních tematických celků.

V teoretické části se definuje systém obecně, jsou zde popisovány různé definice systémů, jejich rozdělení, základní jevy a matematické prostředky pro jejich popis. Dále se pak popisují regulační obvody, jejich rozdělení, struktura a typy. Podrobně jsou zde rozebírány jednotlivé regulátory a typy regulovaných soustav.

Další kapitola je věnovaná matematickým prostředkům pro výpočet diferenciálních rovnic. Diferenciální rovnice tvoří základ pro popis jevů v daných systémech.

Poslední část se zabývá návrhem a implementací simulátorů regulačních soustav. S modely těchto soustav byla provedena řada experimentů s vyhodnocením.

Část práce je zpracována v programu Microsoft Power Point (.ppt). Jsou zde zahrnuty výsledky jednotlivých experimentů, které byly provedeny jak v TKSL, tak i v TKSL/C. Výstup simulace byl zpracován do grafů ve formátu Microsoft Excel.

Klíčová slova

Regulační obvod, regulace, řízení, regulátor, odezva na jednotkový skok, přechodná charakteristika, bloková algebra, diferenciální rovnice, Taylorova řada, simulační systém, TKSL, TKSL/C, A/D převodník, D/A převodník, řád soustavy, integrační krok výpočtu.

Abstract

The final year thesis deals with real-time modelling, especially with the control circuits, where we focus on the regulation circuit with auxiliary model of the regulated system. The introduction contains the Theory of Controls and regulation. The work is divided into two basic parts.

The first part deals with the problems from the theoretical point of view. The theoretical part is concerned with the general systems. Various definitions of systems, their partition and mathematical methods for their description are mentioned after introducing. A description the regulation circuits, their partition, structures and types follows. There are shown the particular regulators and different types of regulated systems in details. Mathematical methods applied to solve the differential equations that describe the models of the systems are summarized in the second part of the work. The end of this chapter is concerned with the TKSL system.

In the practical part I have produced Microsoft Power Point program which summarizes the results of experiments, made in the TKSL and TKSL/C system. The output of simulation was transformed into the graphs in the format of Microsoft Excel.

The last part deals with suggestion and implementation simulators of control circuits.

Keywords

Control circuit, regulation, controlling, regulator, step response, transient characteristic, block algebra, differential equations, Taylor series, simulation systems, TKSL, TKSL/C, A/D and D/A converter, level of system, integration step.

OBSAH

OBSAH	5
1. Úvod	7
2. Úvod do teorie systémů	8
3. Popis jevů v lineárních systémech	9
3.1 Vnější popisy spojitých lineárních systémů	9
3.2 Vnitřní popisy spojitých lineárních systémů	10
4. Teorie řízení a regulace	14
4.1 Základní pojmy	14
4.2 Struktura regulačních obvodů	14
• Regulovaná veličina (y):	15
• Akční veličina (x):	15
• Řídící veličina (w):	15
• Regulační odchylka (e):	15
• Poruchové veličiny (u_1 , u_2 , u_3 , u_4):	16
4.3 Regulátory	18
4.3.1 Přímochonné regulátory:	18
4.3.2 Regulátory s pomocným zdrojem energie:	18
4.4 Základní typy regulátorů:	18
4.4.1 Proporcionální P-regulátor	18
4.4.2 Integroační regulátor (I-regulátor)	19
4.4.3 Derivační regulátor (D-regulátor)	20
4.5 Složené typy regulátorů:	21
4.5.1 PD regulátor: proporcionálně derivační regulátor	21
4.5.2 PI regulátor: proporcionálně integroační regulátor	21
4.5.3 PID regulátor: proporcionálně derivačně integroační regulátor	22
4.6 Rozvětvené regulační obvody	22
5. Matematický úvod do problematiky	24
5.1 Metody pro výpočet diferenciálních rovnic	24
5.1.1 Analytické řešení	24
5.1.2 Numerické řešení	25
5.1.2.1 Jednokrokové metody	25
6. Systém TKSL	28
6.1 TKSL/386	28
6.2 TKSL/C	28
7. Simulace regulačních obvodů v TKSL	29
Operátorový přenos jednotlivých bloků	29
Program v TKSL	30
Výstup simulace (časový průběh regulované veličiny y) pro různá a je v obr 7.3 až obr 7.7.	30
Operátorový přenos jednotlivých bloků:	32
Program v TKSL	33
Operátorový přenos jednotlivých bloků	34
Program v TKSL	35
8. Experimentální regulovaná soustava	36

8.1	Neřízená soustava.....	36
8.2	Soustava s regulátorem.....	41
(8.59)	49
8.3	Implementace programu.....	50
8.3.1	Implementace Taylorovy metody.....	50
(8.60)	50
8.3.2	Implementace A/D a D/A převodníků.....	51
8.4	Ovládání programu.....	51
8.4.1	Program verze 1.....	52
8.4.2	Program verze 2.....	53
9.	Závěr.....	69
10.	Literatura.....	70
11.	Přílohy – slide v PowerPointu.....	71
11.1	Regulace regulátory P, I, D.....	71
11.2	Regulační obvody.....	73
11.3	Analýza regulačního obvodu.....	80

1. Úvod

Pro popis chování objektů reálného světa lze použít dva odlišné přístupy, spojitý a diskrétní. O tom, které prostředky budou vhodnější, rozhoduje úroveň abstrakce, kterou při modelování použijeme. Výběr popisu modelu proto závisí jak na účelu modelu, tak na úrovni našich znalostí o modelovaném systému. Jednou z nejpřirozenějších forem popisu dynamického systému je blokový diagram. V blokovém diagramu jsou jednotlivé elementy systému znázorněny pomocí bloků a vazeb mezi bloky.

Blokový diagram se dále převede na soustavu obyčejných diferenciálních, případně diferenčních nelineárních rovnic a tyto rovnice jsou řešeny běžnými numerickými metodami. K dispozici se nabízí např. metoda Runge-Kutta, Eulerova metoda, Taylorova metoda a jiné.

Tato diplomová práce se zabývá modelováním a simulací regulačních obvodů. Pro experimentování je zvolena soustava, ke které je připojen regulátor a následně se simuluje spolupráce modelu soustavy se soustavou reálnou. Součástí simulace je ověření vlivu převodníků A/D a D/A mezi soustavou a modelem.

Pro řešení diferenciálních rovnic, popisujících model systému, je využívána Taylorova metoda. Výhodou této nové metody je její rychlost a poměrně snadná implementace.

Součástí této práce je implementace simulačního systému regulačních soustav, včetně grafického uživatelského rozhraní. Program byl implementován v C++ Builderu na platformě Win32.

2. Úvod do teorie systémů

Velmi důležitou částí teorie systémů je zkoumání reality. Při zkoumání reality se obvykle omezíme pouze na tu část, která je pro nás zajímavá a tuto část nazveme objektem. Vše ostatní nazveme okolím objektu. Na zvoleném objektu pozorujeme určité veličiny, jejichž výběr záleží na cíli pozorování. Výběrem veličin a sledováním pro nás podstatných vlastností objektu definujeme na daném objektu systém. Jelikož je možné na určitý objekt pohlížet z několika různých hledisek, lze na daném objektu definovat větší množství systémů. Systém je vlastně abstraktním modelem reálného objektu. Systém formulovaný jako matematický popis studovaných projevů objektu může v procesu poznání v mnoha směrech zastupovat původní objekt a sehrát tak obdobnou úlohu jako fyzikální modely v experimentech. Objekt musíme také specifikovat co do podrobnosti, s níž má být zkoumán. Zanedbávání detailů za jistou zvolenou mez je nezbytným kompromisem mezi přesností a složitostí definovaného systému.

3. Popis jevů v lineárních systémech

V lineárních systémech existují dvě metody jak přistupovat k popisu jevů a to metodou vnějšího popisu nebo vnitřního popisu.

Vnější popis systému je založen na vztahu výstupních a vstupních veličin. Vnitřní popis systému je založen na tom, že „vidíme dovnitř systému“ a máme možnost přesně zkoumat vliv vstupních veličin na stavové veličiny a současný vliv vstupních a stavových veličin na výstup.

3.1 Vnější popisy spojitých lineárních systémů

Vztah mezi vstupem a výstupem systému může být vyjádřen různými způsoby. Vstup budeme označovat jako $u(t)$ a výstup jako $y(t)$.

Lineární stacionární spojitý systém se vstupem $u(t)$ a výstupem $y(t)$ je popsán diferenciální rovnicí typu:

$$a_n y^{(n)}(t) + a_{n-1} y^{(n-1)}(t) + \dots + a_1 y'(t) + a_0 y(t) = b_m u^{(m)}(t) + b_{m-1} u^{(m-1)}(t) + \dots + b_1 u'(t) + b_0 u(t) \quad (3.1)$$

kde, $a(i)$, $i = 0, 1, 2, \dots, n$ a $b(j)$, $j = 0, 1, 2, \dots, m$ jsou konstantní koeficienty.

Operátorový přenos systému je roven poměru Laplaceova obrazu výstupního signálu k Laplaceově obrazu vstupu, za předpokladu nulových počátečních podmínek. Přenos systému popsaného diferenciální rovnicí je:

$$F(p) = \frac{Y(p)}{U(p)} = \frac{b_m p^m + b_{m-1} p^{m-1} + \dots + b_1 p + b_0}{a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0} \quad (3.2)$$

Z podmínky fyzikální realizovatelnosti plyne, že stupeň polynomu v čitateli musí být menší nebo roven stupni polynomu ve jmenovateli přenosu. Pro analýzu systémů je důležitý právě polynom ve jmenovateli přenosu, který lze též vyjádřit jako součin kořenových činitelů :

$$a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0 = a_n (p - p_1)(p - p_2) \dots (p - p_n) \quad (3.3)$$

p_1, \dots, p_n jsou póly přenosu systému.

Kořeny polynomu v čitateli:

$$B(p) = b_m p^m + b_{m-1} p^{m-1} + \dots + b_1 p + b_0 = 0 \quad (3.4)$$

označujeme jako nuly přenosu systému a lze psát :

$$b_m p^m + b_{m-1} p^{m-1} + \dots + b_1 p + b_0 = b_m (p - n_1)(p - n_2) \dots (p - n_m) \quad (3.5)$$

S využitím takto určených pólů a nul můžeme přenos vyjádřit v tomto tvaru:

$$F(p) = \frac{b_m (p - n_1)(p - n_2) \dots (p - n_m)}{a_n (p - p_1)(p - p_2) \dots (p - p_n)} \quad (3.6)$$

3.2 Vnitřní popisy spojitých lineárních systémů

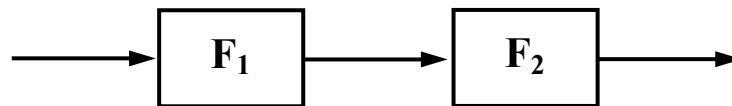
Je to nejobecnější popis systému. Vychází z vnitřních dějů systému, a proto umožňuje odhalit některé vlastnosti systému, které nelze vyčíst z jeho vnějšího popisu.

V teorii řízení je vnitřní popis založen na blokové algebře. Každý systém je složen z vzájemně propojených podsystémů (bloků). Proto bývá důležité určit, jaké vlastnosti bude mít složitý systém, známe-li vlastnosti všech jeho podsystémů a způsob jejich vzájemného propojení.

Existují tři základní způsoby propojení dvou systémů: sériové, paralelní a zpětnovazební.

- Sériové spojení:

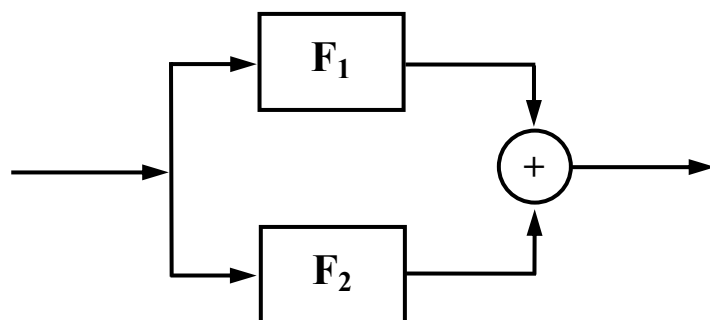
$$F(p) = F_1(p) \cdot F_2(p) \quad (3.7)$$



Obrázek 3.1: Sériové spojení

- Paralelní spojení:

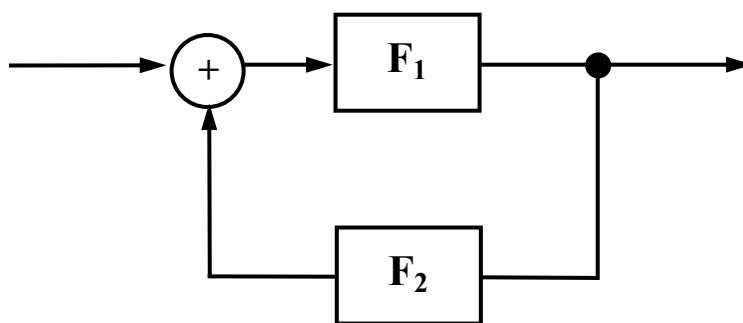
$$F(p) = F_1(p) + F_2(p) \quad (3.8)$$



Obrázek 3.2: Paralelní spojení

- Zpětnovazební spojení:

$$F(p) = \frac{F_1(p)}{1 - F_1(p) \cdot F_2(p)} \quad (3.9)$$

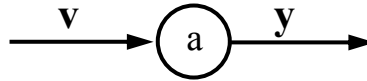


Obrázek 3.3: Zpětnovazební spojení

Nástrojem pro realizaci pravidel blokové algebry jsou grafy signálových toků. Základní operace lineárního systému jsou:

- Násobení signálu konstantou:

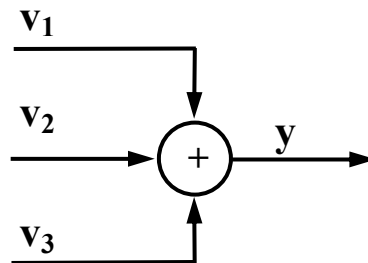
$$y = a \cdot v \quad (3.10)$$



Obrázek 3.4: Násobení signálu konstantou

- Sčítání signálů (sumace):

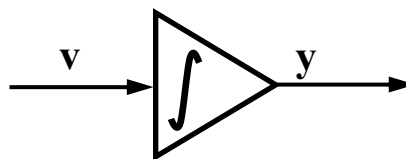
$$y = v_1 + v_2 + \dots + v_k \quad (3.11)$$



Obrázek 3.5: Sčítání signálu

- Integrace signálu: elementární blok integrátor

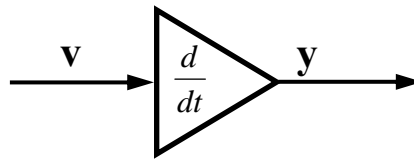
$$y = \int v \cdot dt \quad (3.12)$$



Obrázek 3.6: Integrace signálu

- Derivace signálu: elementární blok derivátor

$$y = \frac{dv}{dt} \quad (3.13)$$



Obrázek 3.7: Derivace signálu

Bloková algebra je základní metodou používanou v teorii řízení a regulaci.

4. Teorie řízení a regulace

V důsledku rozvoje průmyslové výroby dochází k zavádění speciálních zařízení, tzv. automatů, které nahrazují činnost člověka. Proces nahrazování lidské činnosti činností různých přístrojů a zařízení nazýváme automatizací. Teoretickým základem automatizace je teorie automatického řízení, která se zabývá popisem a rozbořem vlastností řídicích a regulačních obvodů a návrhy regulátorů pro dosažení zadaných požadavků.

4.1 Základní pojmy

Mechanizace je proces, jež zaváděním strojních mechanismů osvobozuje člověka od namáhavé, fyzicky náročné práce.

Automatizace je vývojové stádium rozvoje techniky, ve kterém automatizační zařízení vykonává nejen fyzickou, ale i řídicí práci.

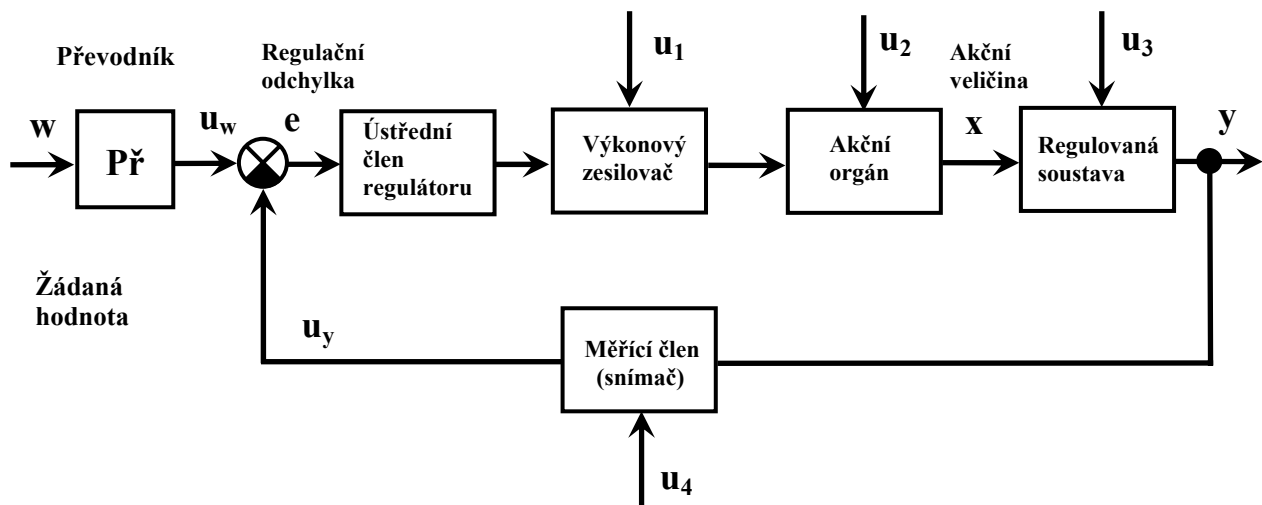
Řízení je každé cílevědomé působení na řízený objekt tak, aby bylo dosaženo určitého předepsaného cíle. Podle toho, jak je řízení prováděno, rozlišujeme řízení ruční a automatické.

Ovládání nebo též řízení bez zpětné vazby je takové řízení, které probíhá na základě zadaných pravidel (algoritmů) bez zpětné kontroly výsledků řídicích zásahů měření.

Regulace nebo též řízení se zpětnou vazbou probíhá tak, že velikost zvolené veličiny je měřena, porovnávána s žádanou hodnotou a řízení probíhá v závislosti na vypočtené odchylce. Toto řízení je daleko dokonalejší než v předchozím případě tudíž jsou dosahované výsledky mnohem kvalitnější.

4.2 Struktura regulačních obvodů

Teorie automatického řízení se zabývá analýzou a syntézou regulačních obvodů se zpětnou vazbou. Jsou to takové systémy, ve kterých je velikost regulované veličiny měřena a porovnávána s žádanou hodnotou. Vznikne tak obvod, který je blokově znázorněn pod tímto textem. Jsou v něm vyznačeny některé základní veličiny všech regulačních obvodů.



Obrázek 4.1: Regulační obvod se zpětnou vazbou

• **Regulovaná veličina (y):**

je výstupem regulované soustavy. Udržování velikosti této veličiny na požadované hodnotě je právě úkolem regulačního obvodu.

• **Akční veličina (x):**

je vstupem regulované soustavy. Je realizovaná akčním orgánem, který je napájen výkonovým zesilovačem. Ústřední člen regulátoru určuje algoritmus řízení, což znamená, že v něm probíhají požadované matematické operace.

• **Řídící veličina (w):**

též žádaná hodnota, je nositelem informace o tom, jaká hodnota regulované veličiny má být nastavena.

• **Regulační odchylka (e):**

je definována jako rozdíl žádané hodnoty a regulované veličiny $e = w - y$. Regulační odchylku zpracovává ústřední člen regulátoru, pro který je odchylka vstupem. Rozdíl $w - y$ se realizuje v diferenčním členu. K tomu je ovšem třeba, aby obě veličiny měly stejný fyzikální rozměr. Proto je do

obvodu zařazen převodník (Př), který převede řídicí veličinu na stejný tvar jaký má výstup z měřícího členu.

• **Poruchové veličiny (u_1 , u_2 , u_3 , u_4):**

mohou obecně působit v kterémkoliv místě celého obvodu. Nejčastěji se však uplatňují vlivy poruch přímo v regulované soustavě.

Zpětnovazební regulační obvod má tedy dva úkoly:

- Zabezpečit, aby regulovaná veličina co nejlépe sledovala časový průběh řídicí veličiny. Splnění tohoto požadavku charakterizují vlastnosti obvodu z hlediska řízení.
- Kompenzovat působení poruchových signálů tak, aby se jejich vliv projevil na regulované soustavě co nejméně.

Oba tyto požadavky lze shrnout do jednoho s použitím pojmu regulační odchylka.

Úlohou regulátoru v uzavřeném regulačním obvodu je řídit soustavu tak, aby regulační odchylka byla co nejmenší (v ideálním případě nulová). Naší snahou je splnění této podmínky nejen v ustáleném stavu, ale i v průběhu přechodného děje.

Podle časového průběhu řídicí veličiny rozpoznáváme několik druhů regulace:

Regulace na konstantní hodnotu:

jedná se o v praxi nejčastější případ, kdy je žádaná hodnota konstantní. U těchto systémů je proto důležitá zvláště funkce kompenzace poruch.

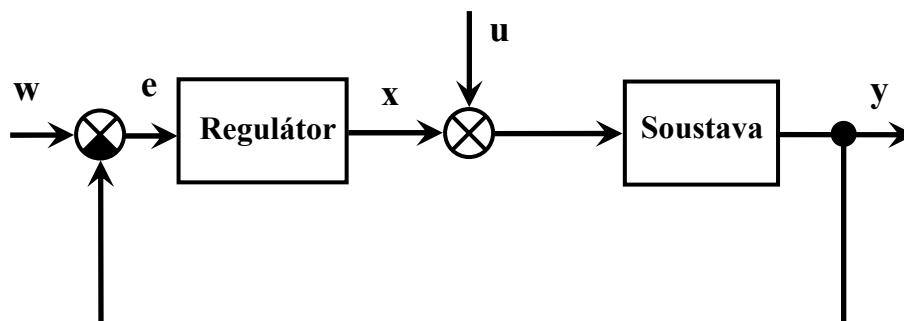
Programová regulace:

je charakteristická tím, že řídicí veličina se mění podle předem daného známého průběhu (programu). V průběhu řídicí veličiny mohou být jak konstantní úseky, tak úseky lineárně nebo kvadraticky proměnné s časem, případně skokové změny. V takovém případě jsou požadavky, kladené na regulační obvod z hlediska řídicí veličiny, daleko přísnější.

Vlečná regulace:

U tohoto typu regulace je řídicí veličina proměnná a její průběh předem neznáme. Speciálním případem vlečné regulace je regulace poměrová, kdy regulovaná veličina má být stále v určitém poměru k další zvolené nezávislé proměnné.

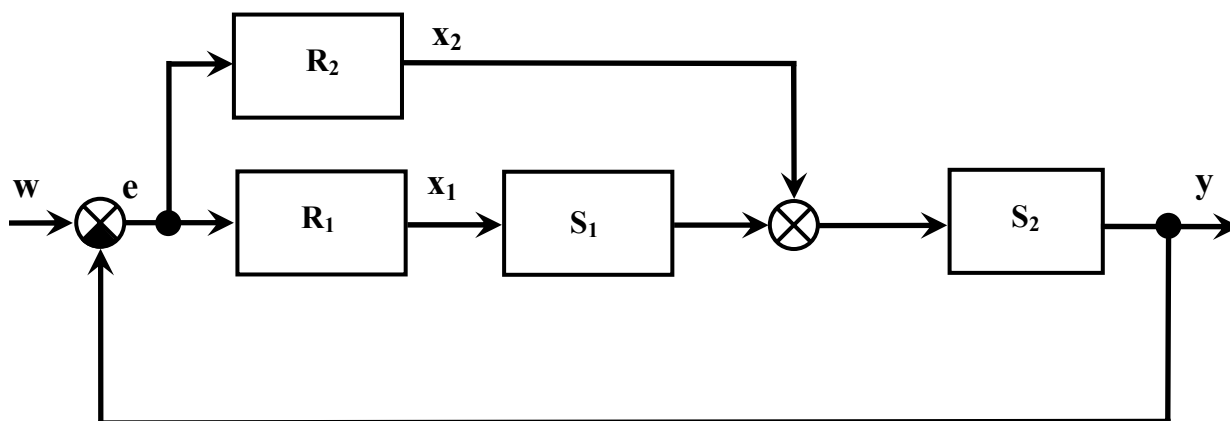
Nejjednodušší schéma obvodu zpětnovazební regulace:



Obrázek 4.2: Regulační obvod se zpětnou vazbou

Příklad rozvětveného obvodu, který je charakterizovaný větším počtem zpětných i dopředných vazeb.

Regulovaná soustava je složená ze dvou částí S_1 a S_2 , má dvě akční veličiny x_1 a x_2 , řízení provádějí dva regulátory R_1 a R_2 , které zpracovávají stejnou regulační odchylku e .



Obrázek 4.3: Rozvětvený regulační obvod s pomocnou akční veličinou

4.3 Regulátory

Regulátor je zařízení, které pomocí akční veličiny působí na soustavu tak, aby odchylka regulované veličiny od žádané hodnoty byla co nejmenší. Všechny členy zpětnovazebního regulačního obvodu, s výjimkou regulované soustavy, zahrnujeme pod pojem regulátor. Jeho úlohou je měřit velikost regulované veličiny, porovnat ji s velikostí žádané hodnoty, tuto regulační odchylku vhodně zesílit a případně upravit a po výkonovém zesílení pomocí akčního členu působit na soustavu. Pokud neuvažujeme poruchové signály, má regulátor dva vstupy $y(t)$ a $w(t)$ a jeden výstup $x(t)$.

Blok, označený názvem ústřední člen regulátoru, určuje, jakými časovými, frekvenčními či logickými úpravami je regulační odchylka $e(t)$ zpracována, než se změní ve výstupní signál regulátoru, tj. akční veličinu. Tento člen má z hlediska regulace mimořádný význam, neboť určuje řídicí algoritmus, který je spolu s vlastnostmi regulované soustavy určující pro výslednou kvalitu regulace. Podle druhu energie, která napájí samotný regulátor, dělíme regulátory na:

4.3.1 Přímočinné regulátory:

Nemají vlastní zdroj energie a ke své činnosti využívají pouze energii odebíranou z regulované soustavy. Do této skupiny patří velká většina jednoduchých průmyslových regulátorů. Tyto regulátory jsou většinou nelineární, akční veličina může nabývat pouze omezený počet hodnot (často pouze dvě: zapnuto – vypnuto). Jsou to známé reléové regulátory používané např. v chladničkách, žehličkách apod.

4.3.2 Regulátory s pomocným zdrojem energie:

Zde jde o složitější zařízení, jehož jádrem je vždy zesilovač. Dosahovaná kvalita regulace je podstatně vyšší, úměrně nákladům a složitosti. Statické vlastnosti těchto regulátorů považujeme v určitém pracovním rozsahu za lineární.

4.4 Základní typy regulátorů:

4.4.1 Proporcionální P-regulátor

Je to prostý zesilovač. Je tak nazýván, protože akční odchylka je v tomto případě proporcionálně úměrná regulační odchylce.

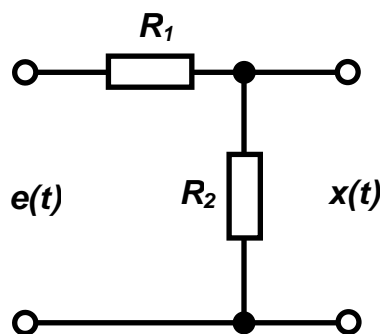
P-regulátor :

$$x(t) = r_0 \cdot e(t) \quad (4.1)$$

přenos je:

$$F_R(p) = \frac{X(p)}{E(p)} = r_0 \quad (4.2)$$

Příklad pasivního P-regulátoru je v obrázku 4.4:



Obrázek 4.4: Pasivní P-regulátor

4.4.2 Integrovní regulátor (I-regulátor)

Častým požadavkem na regulaci je, aby regulovaná veličina přesně souhlasila s řídicí veličinou, tzn. aby regulační odchylka byla v ustáleném stavu nulová. Přitom akční veličina nulová není, neboť je-li v obvodu statická soustava, musí nenulové regulované veličině odpovídat též nenulová akční veličina. Regulátor je pak článek, který i při nulovém vstupním signálu (odchylce) má nenulový výstup.

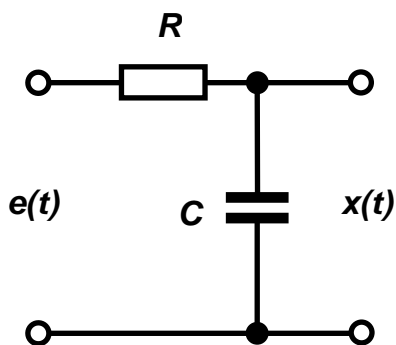
I-regulátor:

$$x(t) = r_i \cdot \int_0^t e(t) dt + x(0) \quad (4.3)$$

přenos je:

$$F_R(p) = \frac{X(p)}{E(p)} = \frac{r_i}{p} = \frac{1}{T_i p} \quad (4.4)$$

Příklad pasivního I-regulátoru je na obrázku 4.5.



Obrázek 4.5: Pasivní I-regulátor

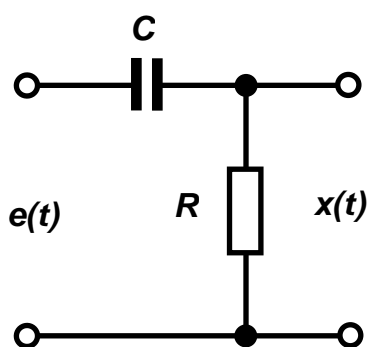
4.4.3 Derivační regulátor (D-regulátor)

Jeho výstupní signál je derivací vstupního signálu, neboli změna akční veličiny je úměrná rychlosti změny odchylky. Pro konstantní odchylku je nulová a regulátor se tedy chová jako rozpojený spínač. Proto jej nelze zařazovat do přímé větve regulačního obvodu samostatně. Může být pouze součástí regulátorů PD a PID. Samostatně jej lze použít do tzv. malé zpětnovazební smyčky, sloužící ke stabilizaci regulačního obvodu.

D-regulátor:
$$x(t) = T_d' \cdot e(t) \quad (4.5)$$

přenos je:
$$F_R(p) = \frac{X(p)}{E(p)} = T_d p \quad (4.6)$$

Příklad pasivního D-regulátoru je na obrázku 4.6.



Obrázek 4.6: Pasivní D-regulátor

4.5 Složené typy regulátorů:

Existují tři základní typy sdružených regulátorů:

4.5.1 PD regulátor: proporcionálně derivační regulátor

$$x(t) = r_0 \cdot e(t) = r_d \frac{d \cdot e(t)}{d(t)} \quad (4.7)$$

přenos je:
$$F_R(p) = \frac{X(p)}{E(p)} = r_0 + r_d = K_R (T_D p + 1) \quad (4.8)$$

platí:
$$K_R = r_0, \quad T_D = \frac{r_d}{r_0}$$

4.5.2 PI regulátor: proporcionálně integrační regulátor

$$x(t) = r_0 \cdot e(t) + r_i \cdot \int_0^t e(t) dt + x(0) \quad (4.9)$$

přenos je:
$$F_R(p) = \frac{X(p)}{E(p)} = r_0 + \frac{r_i}{p} = K_R \frac{(T_R p + 1)}{p} = \frac{T_R p + 1}{T_i p} \quad (4.10)$$

platí:
$$K_R = r_i = \frac{1}{T_i}, \quad T_R = \frac{r_0}{r_i}, \quad r_0 = \frac{T_R}{T_i}$$

4.5.3 PID regulátor: proporcionálně derivačně integrační regulátor

$$x(t) = r_0 \cdot e(t) + r_d \frac{d \cdot e(t)}{dt} + r_i \cdot \int_0^t e(t) dt + x(0) \quad (4.11)$$

přenos je:

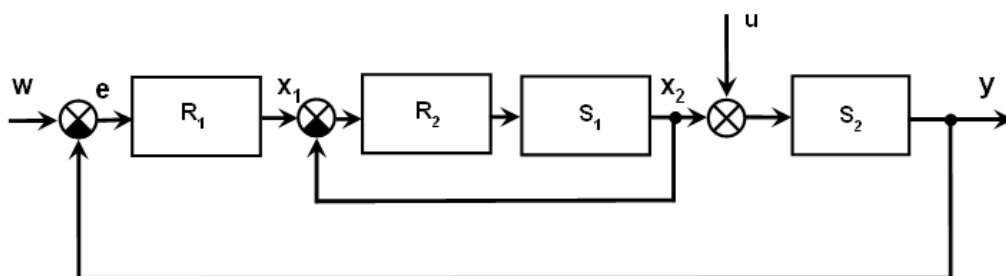
$$F_R(p) = \frac{X(p)}{E(p)} = r_0 + \frac{r_i}{p} + r_d p = K_R \left(1 + D \cdot p + \frac{1}{I \cdot p} \right) = k_r \left(i + T_d p + \frac{1}{T_i p} \right) = \frac{(T_1 p + 1)(T_2 p + 1)}{T_c p} \quad (4.12)$$

platí: $r_i = \frac{K_R}{I} = \frac{k_r \cdot i}{T_i}$, $r_d = K_R D = \frac{T_1 T_2}{T_c} = k_r T_d$, $r_0 = K_R = \frac{T_1 + T_2}{T_c} = k_i \cdot i$

4.6 Rozvětvené regulační obvody

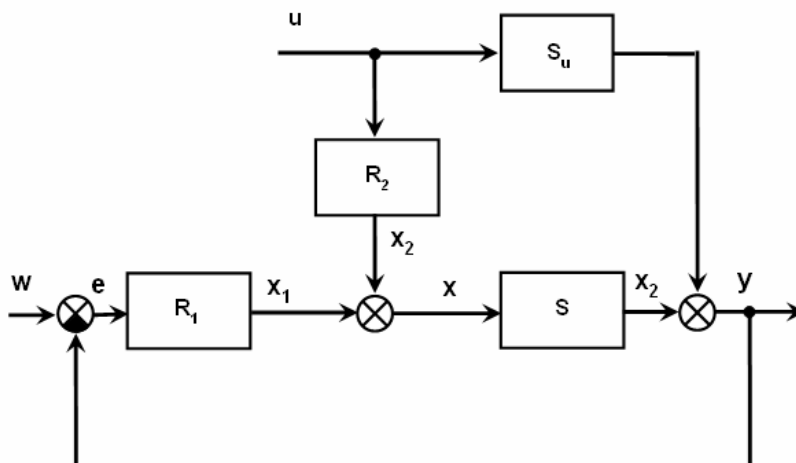
rozeznáváme tyto typy rozvětvených regulačních obvodů:

- Obvody s pomocnou regulovanou veličinou - obr 4.7
- Obvody s měřením poruchy - obr 4.8
- Obvody s modelem regulované soustavy - obr 4.9



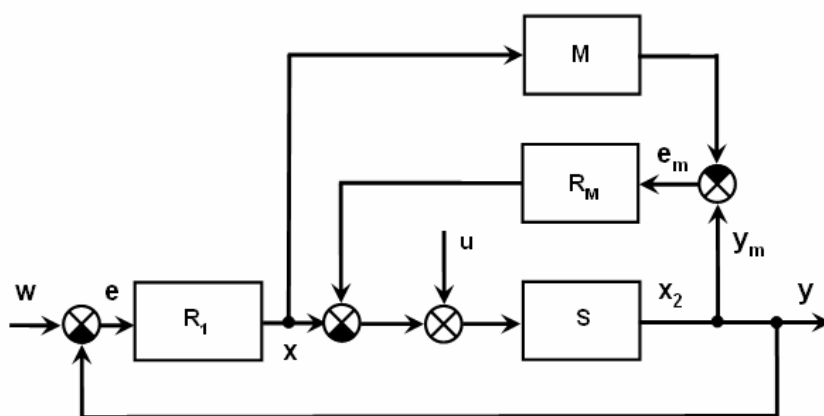
Obrázek 4.7 – Obvod s pomocnou regulovanou veličinou

V regulované soustavě z obrázku 4.7 tvořené sériovým spojením bloků S_1 a S_2 je měřena veličina x_2 , která je regulátorem R_2 řízena podle žádané hodnoty x_1 jako hlavní regulátor pracuje blok R_1



Obrázek 4.8 – Obvod s měřením poruchy

V regulačním obvodu z obrázku 4.8 porucha $u(t)$ prochází článkem s přenosem $S_u(p)$ a přičítá se k výstupu regulované soustavy. Současně tuto poruchu měříme a přes regulátor R_2 přičítáme k akční veličině $x_1(t)$. Přenos řízení zůstává touto přídatnou vazbou nezměněn. Přenos poruchy je možno regulátorem $R_2(p)$ výrazně měnit. Teoreticky lze dosáhnout úplné kompenzace poruchového signálu.



Obrázek 4.9 – Obvod s modelem regulované soustavy

Systémy s modelem regulované soustavy se používají hlavně v adaptivních obvodech, mohou však zlepšit i kvalitu regulace v jednoduchém regulačním obvodu. Blokové schéma takového systému je na obrázku 4.9 Akční veličina $x(p)$ působí jak na regulovanou soustavu $S(p)$ tak na model $M(p)$. Rozdíl výstupů $y(t) - y_m(t)$ tvoří pomocnou odchylku e_m , kterou zpracovává regulátor $R_m(p)$.

Tato práce se orientuje na obvody s modelem regulované soustavy.

5. Matematický úvod do problematiky

Pro analýzu dějů v regulovaných systémech se používají různé metody. Tato práce se zabývá popisem regulovaných systémů pomocí diferenciálních rovnic a proto je stručně naznačena problematika diferenciálních rovnic.

5.1 Metody pro výpočet diferenciálních rovnic

Diferenciální rovnicí rozumíme každou rovnici, v níž se neznámá funkce vyskytuje v derivaci. Jeli neznámá funkce funkcí jedné proměnné, jde o obyčejnou diferenciální rovnici. Jinak hovoříme o parciální diferenciální rovnici. Řád nejvyšší derivace pak udává řád této diferenciální rovnice. Řešení obsahující konstanty se nazývá obecné řešení. Graf řešení diferenciální rovnice se nazývá integrální křivka. Obyčejnou diferenciální rovnicí prvního řádu nazýváme rovnici ve tvaru:

$$F(x, y, y') = 0 \quad (5.1)$$

Explicitním vyjádření:

$$y' = f(x, y), \quad f : \Omega \rightarrow \mathbb{R}^2, \quad \Omega \in \mathbb{R}^2 \quad (5.2)$$

Diferenciální rovnice :

$$y' = f(x, y) \quad (5.3)$$

má pro každý bod jediné řešení procházející bodem, pokud jsou splněny následující podmínky:

1) funkce je spojitá na Ω 2) funkce je ohraničená na Ω 3) funkce splňuje Lipschitzovu podmínku na Ω .

5.1.1 Analytické řešení

Řešením je funkce času. Konkrétní hodnotu v určitém čase získáme dosazením tohoto času do výsledné funkce. Lze určit hodnotu v libovolném bodě, v němž je funkce definována. Analytické metody jsou obvykle složité a časově náročné, ale velmi přesné. Teorie obyčejných diferenciálních rovnic tedy vybírá určité modely jistých skupin diferenciálních rovnic, pro které je nalezeno obecné schéma řešení.

5.1.2 Numerické řešení

Řešením je posloupnost hodnot v určitých předem zvolených časových bodech. Hodnoty funkce mezi zvolenými body lze určit buď interpolací z okolních vypočtených bodů nebo opětovnou aplikací metody s menším rozestupem (krokem) časových bodů. Numerické metody jsou obvykle jednodušší a rychlejší než analytické. Při špatné volbě kroku však může dojít k velké chybě výpočtu.

5.1.2.1 Jednokrokové metody

Pro výpočet hodnoty stačí znát pouze hodnotu $f(a)$. Toto je výhodné v případech, kdy potřebujeme často měnit integrační krok.

Taylorova metoda

Základem pro tyto metody je Taylorova řada:

$$p(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \dots + \frac{f^{(n)}(a)}{n!}(x-a)^n \quad (5.4)$$

Při využívání se předpokládá, že pro výpočet řešení se bere v úvahu větší počet členů rozvoje, řádově alespoň desítky členů. Tato metoda umožňuje výpočet mnohem přesnější hodnoty řešení (bráno vzhledem k délce integračního kroku h), než běžně používané metody (Eulerova či některá varianta metody Runge-Kutta).

Princip metody

Taylorova řada je definována jako nekonečná mocninná řada

$$p(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \dots + \frac{f^{(n)}(a)}{n!}(x-a)^n \quad (5.5)$$

Pokud položíme počáteční podmínku $z = 0$ a položíme $h = x_l - z_l$, pak rovnice přejde do tvaru:

$$f(x) = f(0) + h \cdot f'(0) + \frac{h^2}{2!} f''(0) + \frac{h^3}{3!} f'''(0) + \dots \quad (5.6)$$

Nyní položíme $z_2 = x_1$ za předpokladu $h = x_2 - z_2 = x_1 - z_1$.

$$f(x_2) = f(x_1) + h \cdot f'(x_1) + \frac{h^2}{2!} f''(x_1) + \frac{h^3}{3!} f'''(x_1) + \dots \quad (5.7)$$

Hodnoty funkce $f(x)$ v bodech x_1, x_2 lze vypočítat postupně za využití Taylorovy řady. Výsledek jednoho kroku je nutný pro výpočet dalších dílčích výsledků. Parametr h je integrační krok. Integrační krok nemusí být konstantní. Pro jednotlivé kroky výpočtu se může měnit. Na velikosti integračního kroku je závislá rychlost výpočtu a také jeho přesnost. Čím je integrační krok větší, tím se také zvyšuje rychlost výpočtu. Naopak může klesat přesnost výpočtu. Před začátkem výpočtu musíme určit, s jakou přesností výsledek požadujeme. Při výpočtu pak sčítáme dílčí výsledky a pokud je rozdíl dvou po sobě jdoucích výsledků menší než požadovaná přesnost, výpočet ukončíme.

K dílčím výpočtům potřebujeme znát vyšší derivace funkce. Výpočet vyšších derivací je časově náročný a prakticky zbytečný. Vyšší derivace lze totiž odvodit z předchozích výpočtů. Toto ukážeme na následující soustavě diferenciálních rovnic.

$$y' = A \cdot y + B \cdot z \quad z' = C \cdot y + D \cdot z \quad (5.8)$$

Počáteční podmínky $y(0) = y_0, z(0) = z_0$,

Řešení klasickým způsobem:

$$y_1 = y_0 + h \cdot y'(0) + \frac{h^2}{2!} \cdot y''(0) + \frac{h^3}{3!} \cdot y'''(0) + \dots \quad (5.9)$$

$$z_1 = z_0 + h \cdot z'(0) + \frac{h^2}{2!} \cdot z''(0) + \frac{h^3}{3!} \cdot z'''(0) + \dots \quad (5.10)$$

Výhody a nevýhody použití Taylorovy řady

Nespornou výhodou této metody je její rychlost a s tím spojená výpočetní nenáročnost. Velkou výhodou je také možnost paralelního zpracování dílčích výpočtů, které se uplatňuje při výpočtech soustav diferenciálních rovnic. Pokud ovšem zvolíme špatnou velikost integračního kroku, může se metoda stát nestabilní. Je to způsobeno tím, že při chybném určení jednoho kroku se tato chyba přenáší i do dalších výpočtů a celková chyba tím může narůstat.

Eulerova metoda

Tato metoda je nejjednodušší: pro určení následující hodnoty y_{n+1} bereme v úvahu pouze první dva členy Taylorovy řady, tedy:

$$y_{n+1} = y_n + hy'_n \quad (5.11)$$

Geometrická interpretace této metody není obtížná, protože $h_i = x_{i+1} - x_i$, je vztah (4.5) rovnicí přímky se směrnicí $f(x_i, y_i)$ jdoucí bodem (x_i, y_i) , tj. na intervalu $\langle x_i, x_{i+1} \rangle$ se vždy pohybujeme po tečně k přesnému řešení úlohy $y' = f(x, y)$, $y(x_i) = y_i$ v bodě (x_i, y_i) .

Při zkracování kroku lze zpřesňovat řešení, ovšem od jisté hranice začne převládat vliv zaokrouhlovací chyby a celková chyba výpočtu při dalším zmenšování kroku poroste.

Metoda Runge-Kutta

Další jednokrokové metody, které používají pouze první derivace řešení y - výpočet $f(t, y)$ však provádějí i mezi jednotlivými uzly (t_i, y_i) - jsou zastoupeny metodami typu Runge-Kutta. Základem těchto metod je vyjádření rozdílů mezi hodnotami řešení y v bodech t_{n+1} a t_n ve tvaru:

$$y_{n+1} - y_n = \sum_{i=1}^p w_i K_i \quad (5.12)$$

kde w_i jsou konstanty a

$$K_i = hf \left(t_n + a_i h, y_n + \sum_{j=1}^{i-1} b_{ij} k_j \right), i = 1 \dots p \quad (5.13)$$

kde $t_{n+1} - t_n$ a a_i, b_{ij} jsou konstanty, přičemž $a_i = 0$

Metoda se nazývá p -hodnotová (používá p hodnot funkce $f(t, y)$). Konstanty w_i, a_i, b_i jsou vypočteny tak, aby získaná řešení souhlasila s Taylorovou řadou v bodě (t_n, y_n) až do p -té mocniny kroku h včetně. Metodu pak nazýváme metodou Runge-Kutta řádu P . Je celá řada modifikací těchto metod, liší se především v koeficientech, v principu jsou ovšem stejné.

6. Systém TKSL

Tento simulační jazyk byl vytvořen pro testování algoritmů využívající Taylorovy řady k řešení (systémů) diferenciálních rovnic a ostatních problémů.

TKSL je určen k simulaci tuhých systémů, systémů s nespojitostmi a systémů popsaných parciálními diferenciálními rovnicemi. Veškeré výpočty jsou založeny na diferenciálních rovnicích. Systém provádí numerické řešení pomocí Taylorova rozvoje. Požadovaná přesnost je udržována dynamickým nastavováním řádu Taylorova polynomu ($x > 100$). Vstupem je tedy soustava diferenciálních rovnic popisující simulovanou soustavu.

6.1 TKSL/386

Uživatelské rozhraní pracuje v grafickém režimu. Program disponuje editorem zdrojového kódu. Integrovaný překladač umožňuje ladění a překlad zdrojového kódu. Po přeložení kódu je možné provést simulaci a grafickou reprezentaci výsledku.

Zdrojový kód se programu v TKSL se skládá z těchto částí:

- deklarace proměnných
- definice konstant
- tělo programu
- rozsah os grafu

6.2 TKSL/C

TKSL/C je novější verzi TKSL. Tato verze je implementována v jazyce C, je přenositelná na různé platformy (Windows, Linux...) a umožňuje řešit velmi rozsáhlé soustavy rovnic.

TKSL/C je zatím ve fázi vývoje. V současné době jsou k dispozici pouze konzolové verze překladače a simulátoru, které se spouští s příkazového řádku a svůj výstup generují do textového souboru jako tabulku čísel. Editor ani grafický výstup zatím není implementován.

Zdrojový kód TKSL/C obsahuje pouze jednu sekci a tou je tělo programu. Formát zápisu je podobný jako v TKSL/386.

7. Simulace regulačních obvodů v TKSL

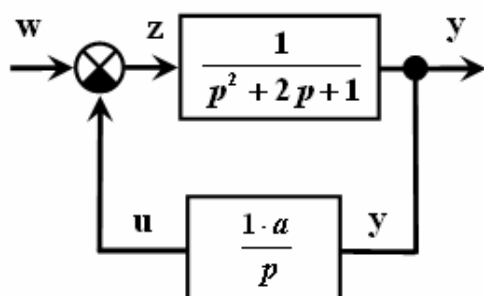
Postup simulace regulačního obvodu je následující.

- nakreslení blokového schématu
- definování operátorového přenosu
- popis soustavou diferenciálních rovnic
- převedení rovnic do programu TKSL, příp. TKSL/C
- provedení simulace
- zobrazení výsledku

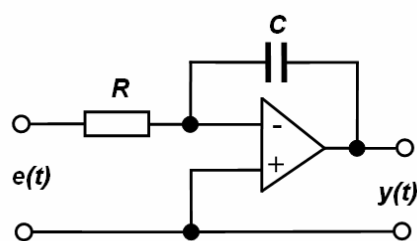
Následující příklady slouží pro první experimenty s regulovanými soustavami.

Příklad 1 – soustava s regulátorem typu I

Blokové schéma je na obrázku 7.1:



Obrázek 7.1: Blokové schéma



Obrázek 7.2: Aktivní I regulátor

Operátorový přenos jednotlivých bloků

$$1. \text{ blok (SOUSTAVA):} \quad \frac{y}{z} = \frac{1}{p^2 + 2p + 1} \quad (7.1)$$

$$\text{úpravou je:} \quad p^2 y + 2py + y = z \quad (7.2)$$

$$2. \text{ blok (REGULÁTOR):} \quad \frac{u}{y} = \frac{a}{p} \quad (7.3)$$

$$\text{úpravou je:} \quad pu = ya \quad (7.4)$$

Odpovídající soustava diferenciálních rovnic je ve tvaru:

$$y'' + 2y' + y = z \qquad u' = y \cdot a \qquad z = w - u \qquad (7.5)$$

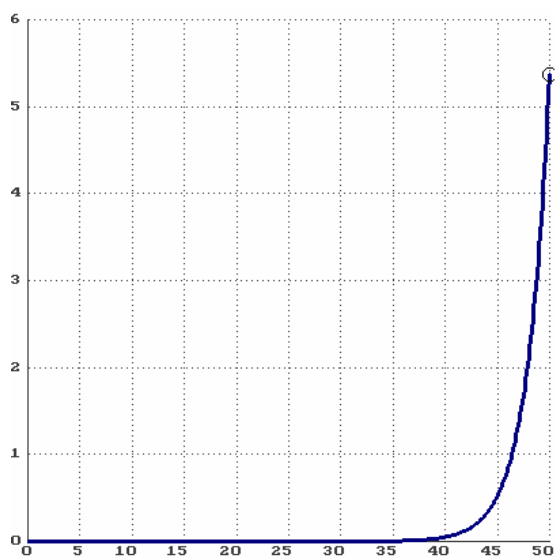
Program v TKSL

Před zápisem programu do TKSL je nutné rovnice vyšších řádů převést na soustavy rovnic 1. řádu

Po substituci $y' = q$ je $q' = (y'') = -2q - y + z$. Následuje zápis v TKSL

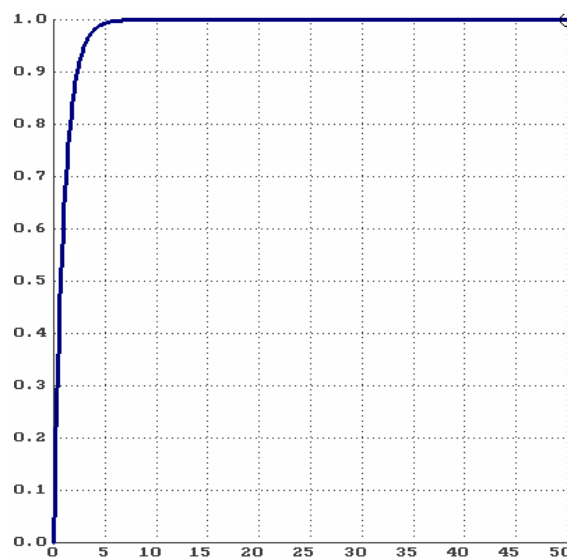
```
var y,z,u,w,q;
const a=0,dt=0.1,
eps=1e-20,tmax=20;
system
  y'=q &0;
  q'=- (2*q)-y+z &0;
  u'=y*a &0;
  z=w-u;
  w=1;
sysend.
```

Výstup simulace (časový průběh regulované veličiny y) pro různá a je v obr 7.3 až obr 7.7.



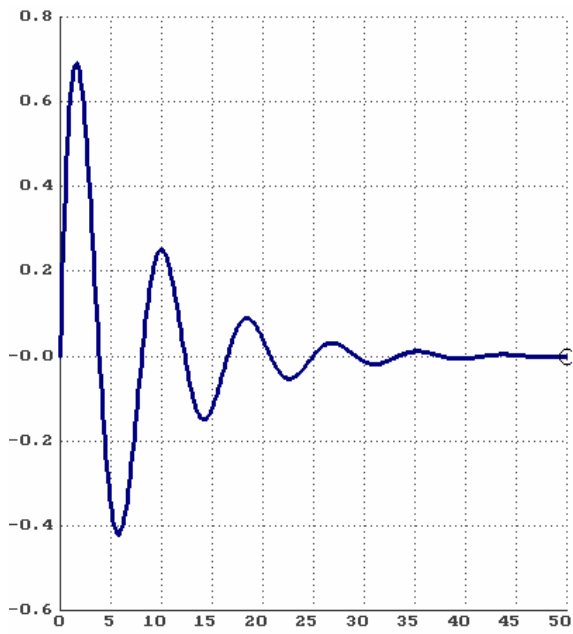
$a = -1$

Obrázek 7.3 - Soustava s I regulátorem



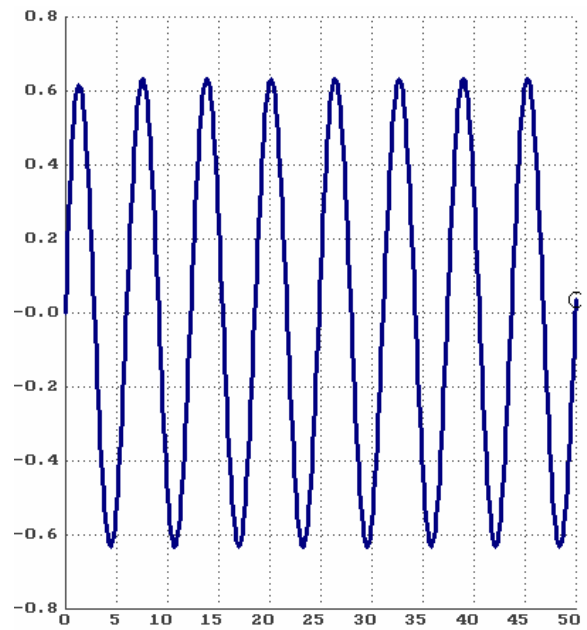
$a = 0$

Obrázek 7.4 - Soustava s I regulátorem



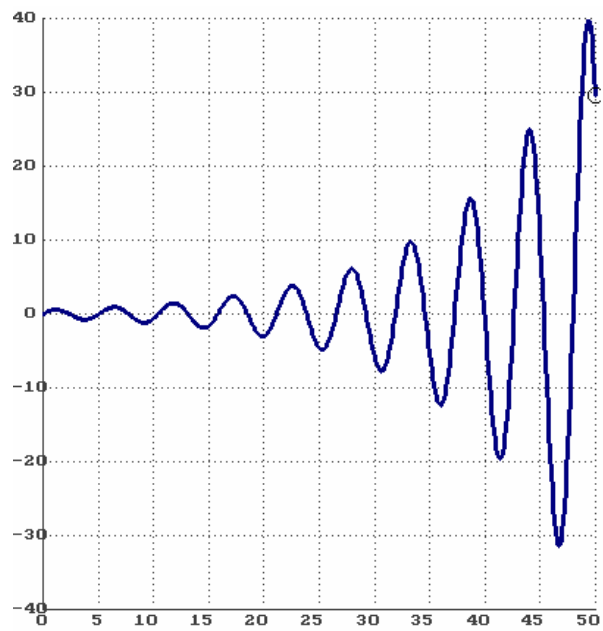
$a = 1$

Obrázek 7.5 - Soustava s I regulátorem



$a = 2$

Obrázek 7.6 - Soustava s I regulátorem

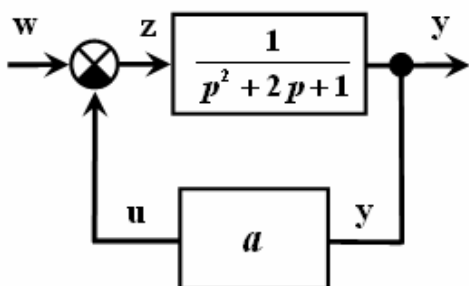


$a = 3$

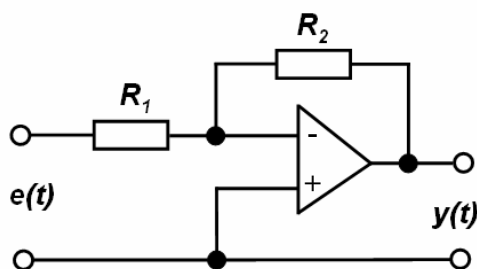
Obrázek 7.7 - Soustava s I regulátorem

Příklad 2 – soustava s regulátorem typu P

Blokové schéma je na obrázku 7.8:



Obrázek 7.8: Blokové schéma



Obrázek 7.9: Aktivní P regulátor

Operátorový přenos jednotlivých bloků:

1. blok (SOUSTAVA):
$$\frac{y}{z} = \frac{1}{p^2 + 2p + 1} \quad (7.6)$$

$$p^2 y + 2py + y = z \quad (7.7)$$

2. blok (REGULÁTOR):
$$\frac{u}{y} = a \quad (7.8)$$

$$u = a \cdot y \quad (7.9)$$

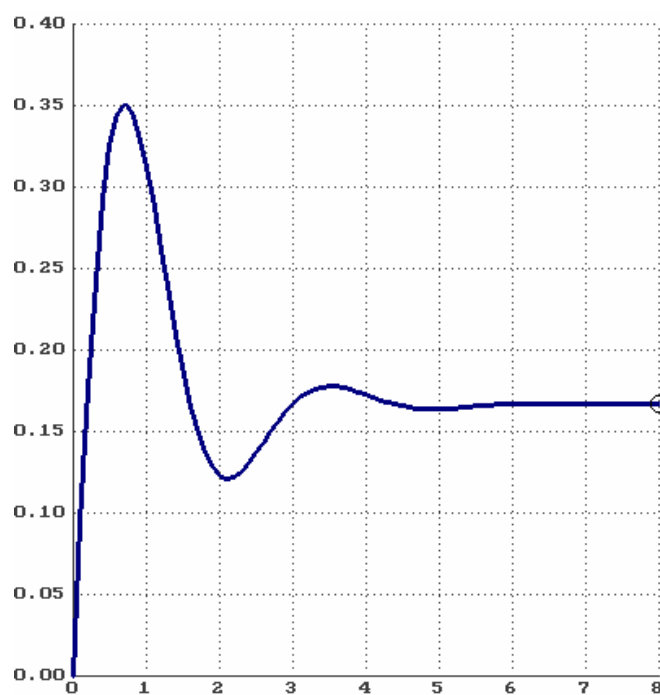
Odpovídající soustava diferenciálních rovnic je ve tvaru:

$$y'' + 2y' + y = z \qquad u = y \cdot a \qquad z = w - u \quad (7.10)$$

Program v TKSL

```
var q,y,z,u,w;  
  
const  
a=5,dt=0.1,eps=1e-20,  
t0=0,tmax=10;  
  
system  
  
y'=q &0;  
q'=- (2*q)-y+z &0;  
u=y*a;  
z=w-u;  
w=1;  
  
sysend.
```

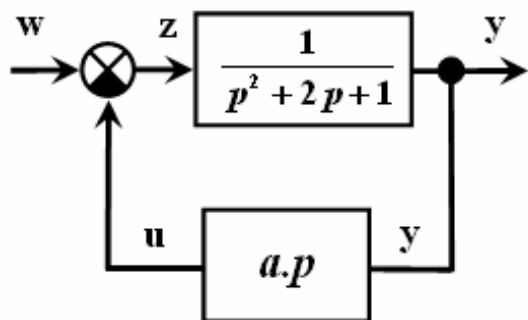
Výstup simulace (časový průběh regulované veličiny y) pro $a=5$ je v obr 7.10.



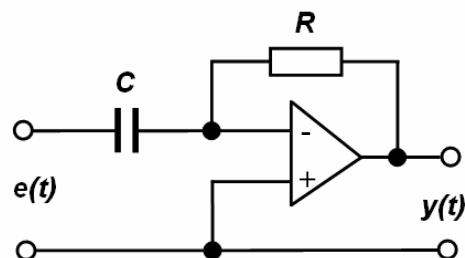
Obrázek 7.10 – Ukázka soustavy s P regulátorem

Příklad 3 – soustava s regulátorem typu D

Blokové schéma je na obrázku 7.11:



Obrázek 7.11: Blokové schéma



Obrázek 7.12: Aktivní D regulátor

Operátorový přenos jednotlivých bloků

1. blok (SOUSTAVA):
$$\frac{y}{z} = \frac{1}{p^2 + 2p + 1} \quad (7.11)$$

$$p^2 y + 2py + y = z \quad (7.12)$$

2. blok (REGULÁTOR):
$$\frac{u}{y} = a \cdot p \quad (7.13)$$

$$u = a \cdot py \quad (7.14)$$

Odpovídající soustava diferenciálních rovnic je ve tvaru:

$$y'' + 2y' + y = z \qquad a \cdot y' = u \qquad z = w - u \quad (7.15)$$

Program v TKSL

```
var q,y,z,u,w;

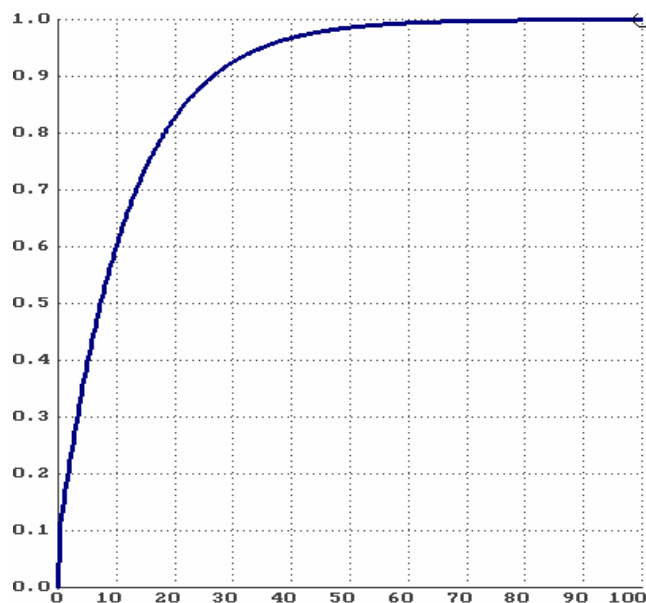
const
a=10,dt=0.1,eps=1e-20,
t0=0,tmax=200;

system

y'=q &0;
q'=- (2*q) -y+z &0;
u=a*q;
z=w-u;
w=1;

sysend.
```

Výstup simulace (časový průběh regulované veličiny y) pro $a=10$ je v obr 7.13.



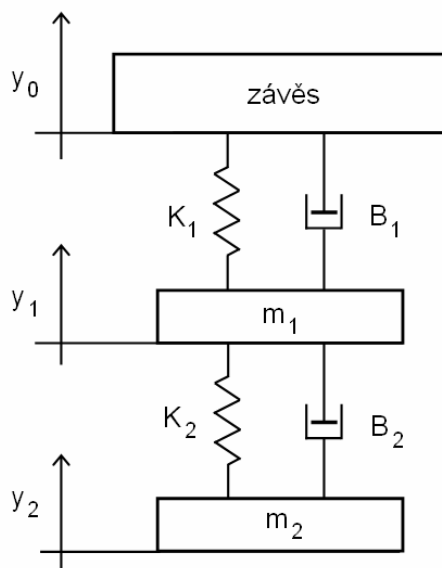
Obrázek 7.13 – Ukázka soustavy s D regulátorem

Chování jednotlivých soustav s regulátory lze samozřejmě matematicky odvodit. Odvození není jednoduché, zvláště u složitějších soustav – proto je výhodné soustavu simulovat. Zcela názorná je tato problematika u následující experimentální soustavy.

8. Experimentální regulovaná soustava

8.1 Neřízená soustava

Je zvolen model následující neřízené soustavy, která se skládá ze dvou závaží zavěšených na tlumených pružinách. Soustava je v obrázku 8.1:



Obrázek 8.1 Mechanická soustava dvou závaží

$$\begin{aligned} m_1 &= 0,4 \text{ kg} \\ m_2 &= 0,2 \text{ kg} \\ K_1 &= 9 \text{ Nm}^{-1} \\ K_2 &= 1,5 \text{ Nm}^{-1} \\ B_1 &= 0,1 \text{ Nm}^{-1} \text{ s} \\ B_2 &= 0,2 \text{ Nm}^{-1} \text{ s} \end{aligned} \quad (8.1)$$

V ustáleném stavu je:

$$\begin{aligned} y_0 &= 0 \text{ dm} \\ \Delta Y_1 &= -\frac{m_1 + m_2}{K_1} g = -6,54 \text{ dm} \\ \Delta Y_2 &= -\frac{m_2}{K_2} g = -13,08 \text{ dm} \end{aligned} \quad (8.2)$$

Vstupní veličina y_0 stanovuje polohu závěsu. Regulovat budeme druhé závaží popsané veličinou y_2 .

Tuto soustavu popisují následující diferenciální rovnice 2.řádu:

$$m_2 \ddot{y}_2 = K_2(y_1 - y_2) + B_2(\dot{y}_1 - \dot{y}_2) \quad (8.3)$$

$$m_1 \ddot{y}_1 = K_1(y_0 - y_1) + B_1(\dot{y}_0 - \dot{y}_1) + K_2(y_2 - y_1) + B_2(\dot{y}_2 - \dot{y}_1) \quad (8.4)$$

Dále upravujeme známým postupem:

$$m_2 p Y_2(p) = \frac{1}{p} K_2(Y_1(p) - Y_2(p)) + B_2(Y_1(p) - Y_2(p)) \quad (8.5)$$

$$m_1 p Y_1(p) = \frac{1}{p} K_1(Y_0(p) - Y_1(p)) + B_1(Y_0(p) - Y_1(p)) + \frac{1}{p} K_2(Y_2(p) - Y_1(p)) + B_2(Y_2(p) - Y_1(p)) \quad (8.6)$$

Dostaneme soustavu 4 rovnic integrátorů:

$$Q(p) = \frac{1}{p} K_1(Y_0(p) - Y_1(p)) \quad (8.7)$$

$$R(p) = \frac{1}{p} K_2(Y_2(p) - Y_1(p)) \quad (8.8)$$

$$Y_1(p) = \frac{1}{m_1 p} [Q(p) + R(p) + B_1(Y_0(p) + Y_1(p)) + B_2(Y_2(p) + Y_1(p))] \quad (8.9)$$

$$Y_2(p) = \frac{1}{m_2 p} [B_2(Y_1(p) + Y_2(p)) - R(p)] \quad (8.10)$$

Po další úpravě dostaneme tvar:

$$(m_1 p^2 + K_1 + B_1 p + K_2 + B_2 p) Y_1(p) = (K_1 + B_1 p) Y_0(p) + (K_2 + B_2 p) Y_2(p) \quad (8.11)$$

$$(m_2 p^2 + K_2 + B_2) Y_2(p) = (K_2 + B_2 p) Y_1(p) \quad (8.12)$$

Úpravou rovnic (8.11) a (8.12) dostaneme přenosy soustavy.

$$F_1(p) = \frac{Y_1(p)}{Y_0(p)} = \frac{(K_1 B_1 p)(m_2 p^2 + K_2 + B_2 p)}{(m_1 p^2 + K_1 + B_1 p + K_2 + B_2 p)(m_2 p^2 + K_2 + B_2 p) - (K_2 + B_2 p)^2} \quad (8.13)$$

$$F_s(p) = F_2(p) = \frac{Y_2(p)}{Y_0(p)} = \frac{(K_1 B_1 p)(K_2 + B_2 p)}{(m_1 p^2 + K_1 + B_1 p + K_2 + B_2 p)(m_2 p^2 + K_2 + B_2 p) - (K_2 + B_2 p)^2} \quad (8.14)$$

Po dosazení číselných hodnot dostaneme tyto přenosy:

$$F_1(p) = \frac{0,02p^3 + 1,82p^2 + 1,95p + 13,5}{0,08p^4 + 0,14p^3 + 2,72p^2 + 1,95p + 13,5} \quad (8.15)$$

$$F_s(p) = F_2(p) = \frac{0,02p^2 + 1,95p + 13,5}{0,08p^4 + 0,14p^3 + 2,72p^2 + 1,95p + 13,5} \quad (8.16)$$

Aby bylo možné při simulaci nastavovat jednotlivé koeficienty, je zvoleno toto označení:

$$F_s(p) = F_2(p) = \frac{ap^2 + bp + c}{dp^4 + ep^3 + fp^2 + gp + j} \quad (8.17)$$

$$F_s(p) = \frac{y}{z} \quad (8.18)$$

$$\frac{y}{z} = \frac{ap^2 + bp + c}{dp^4 + ep^3 + fp^2 + gp + j} \quad (8.19)$$

Následuje používaná metoda postupné integrace:

$$dp^4 y + ep^3 y + fp^2 y + gpy + jy = ap^2 z + bpz + cz \quad (8.20)$$

$$p^4 y = \frac{-ep^3 y}{d} + \frac{p^2(az - fy)}{d} + \frac{p(bz - gy)}{d} + \frac{cz - jy}{d} \quad (8.21)$$

Rovnici 8.21 násobíme výrazem $\frac{1}{p}$

$$p^3 y = \frac{-ep^2 y}{d} + \frac{p(az - fy)}{d} + \frac{bz - gy}{d} + \frac{1}{p} \frac{cz - jy}{d} \quad \left\{ U = \frac{1}{p} \frac{cz - jy}{d} \right\} \quad (8.22)$$

Rovnici 8.22 násobíme výrazem $\frac{1}{p}$

$$p^2 y = \frac{-epy}{d} + \frac{az - fy}{d} + \frac{1}{p} \left(\frac{bz - gy}{d} + U \right) \quad \left\{ S = \frac{1}{p} \left(\frac{bz - gy}{d} + U \right) \right\} \quad (8.23)$$

Rovnici 8.23 násobíme výrazem $\frac{1}{p}$

$$py = \frac{-ey}{d} + \frac{1}{p} \left(\frac{az - fy}{d} + S \right) \quad \left\{ R = \frac{1}{p} \left(\frac{az - fy}{d} + S \right) \right\} \quad (8.24)$$

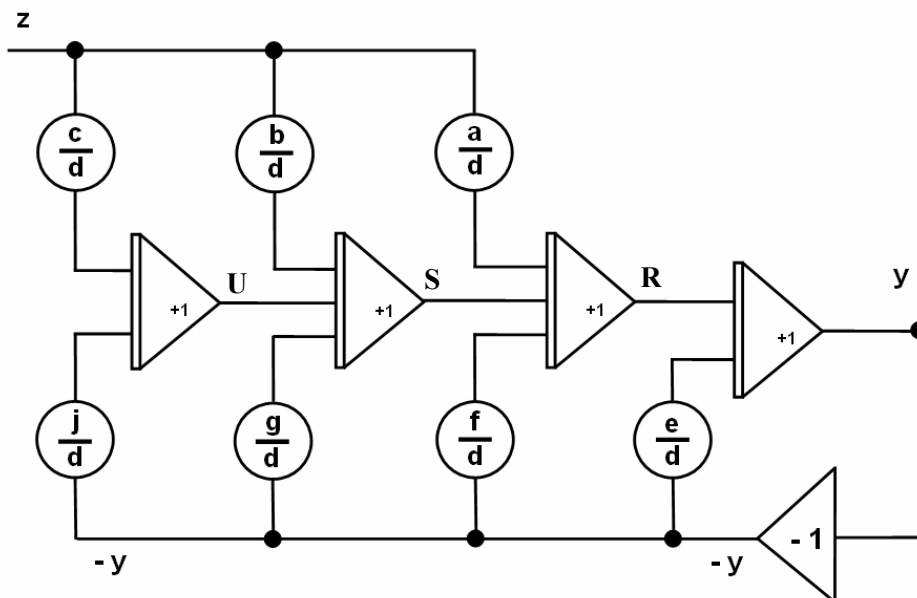
Rovnici 8.24 násobíme výrazem $\frac{1}{p}$

$$y = \frac{1}{p} \left(\frac{-ey}{d} + R \right) \quad (8.25)$$

Získají se tedy rovnice:

$$\begin{aligned} U' &= \frac{cz - jy}{d} \\ S' &= \frac{bz - gy}{d} + U \\ R' &= \frac{az - fy}{d} + S \\ y' &= \frac{-ey}{d} + R \end{aligned} \quad (8.26)$$

Odpovídající schéma je v obrázku 8.2:



Obrázek 8.2 – Experimentální soustava

Soustava se řeší Taylorovou metodou.

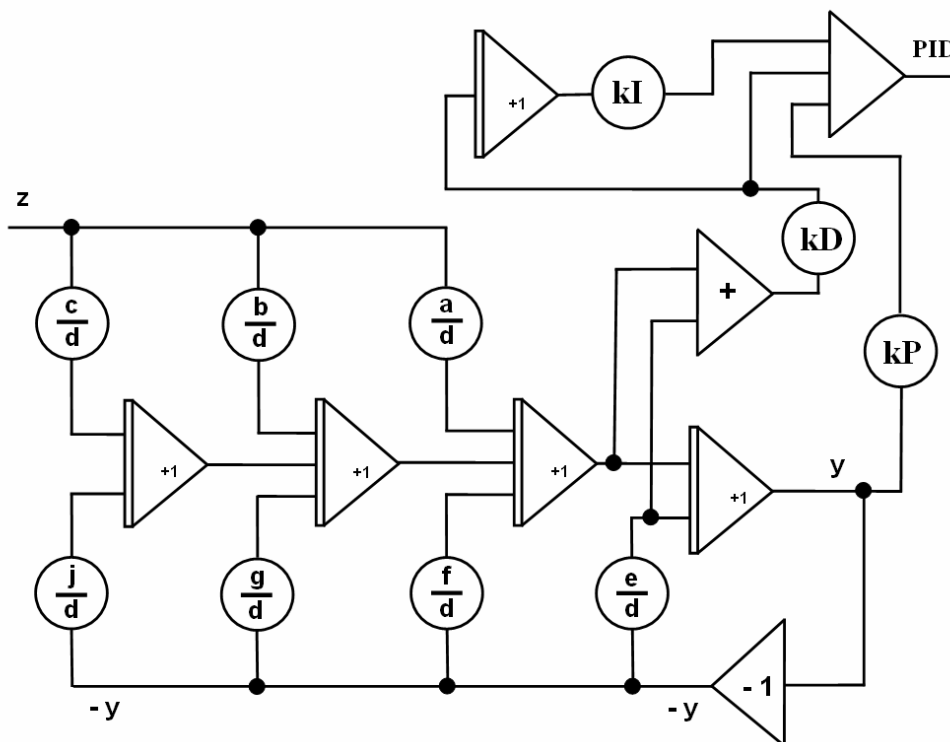
Pro zvolené konstanty je odezva soustavy z obrázku 8.1 uvedena v obrázku 8.3.



Obrázek 8.3 - Neregulovaná soustava

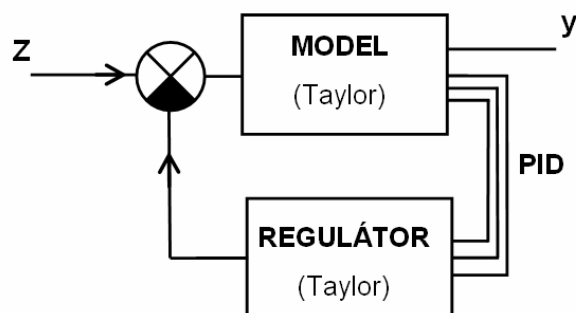
8.2 Soustava s regulátorem

Pro požadavky regulování soustavy pomocí P,I,D regulátorů doplníme schéma z obrázku 8.2 podle obrázku 8.4.



Obrázek 8.4 - Soustava s PID výstupem

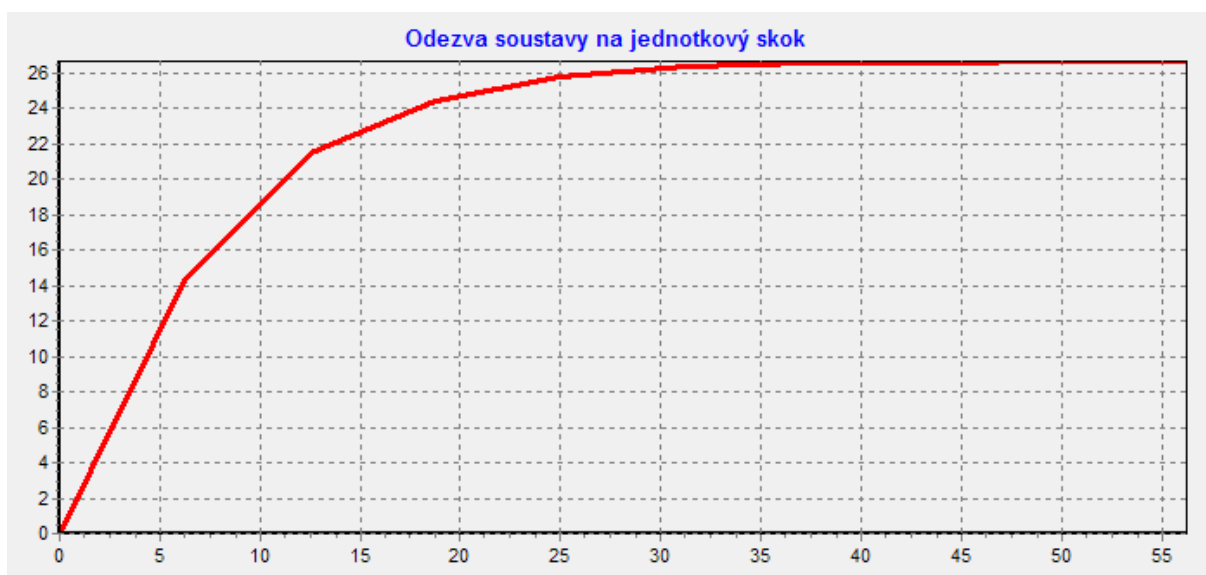
Podle obvyklých zásad teorie řízení se obrázek 8.4 překreslí na obrázek 8.5.



Obrázek 8.5 – Regulovaná soustava

V obrázku 8.5 je provedeno záměrné označení „MODEL“ (Taylor) a „REGULÁTOR“ (Taylor), protože uvedené obvody budou realizovány v počítači jako tzv. „NORMÁLOVÁ“ řešení. (Použitá numerická metoda je Taylorova řada)

Tato regulovaná soustava může být počítaná i s relativně větším integračním krokem. (V obrázku 8.6 je $h=6$)



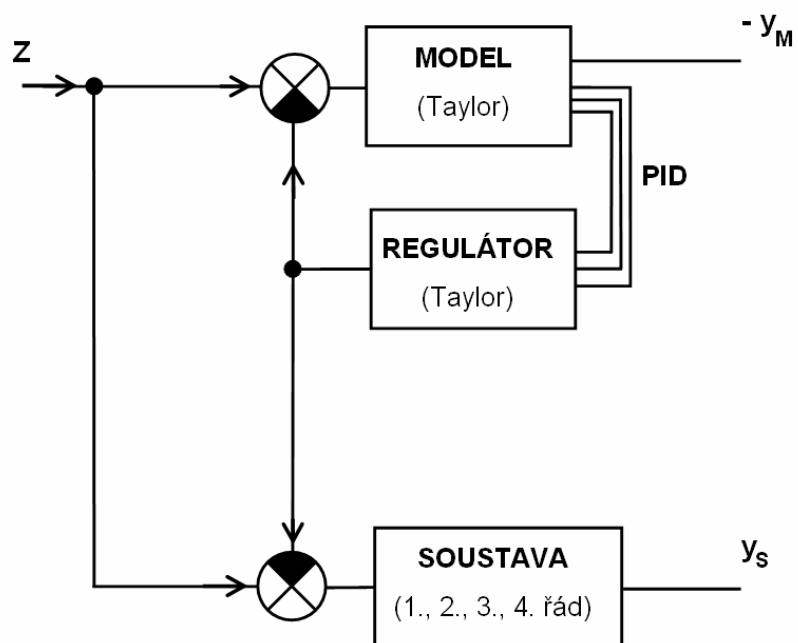
Obrázek 8.6 – Soustava počítaná s relativně velkým integračním krokem

Základní myšlenka využití spojení modelu regulované soustavy (MODEL + REGULÁTOR) a neřízené soustavy je v obrázku 8.7. Neřízená soustava je v obrázku 8.7 označena „SOUSTAVA“ (1., 2., 3., 4. řád) a může být v našich experimentech simulována numerickými metodami 1., 2., 3., 4. řádu.

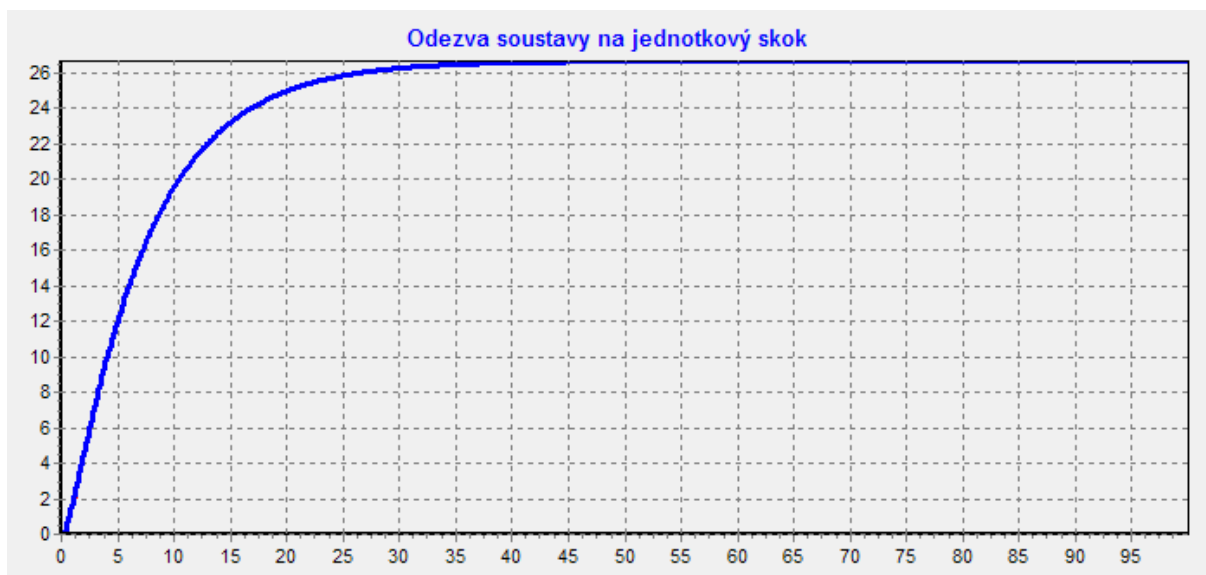
Za předpokladu, že model i soustava obrázku 8.7 mají stejné parametry je shoda y_m a y_s v obrázku 8.8.

Pro případ rozdílných koeficientů je časový průběh y_m a y_s v obrázku 8.9.

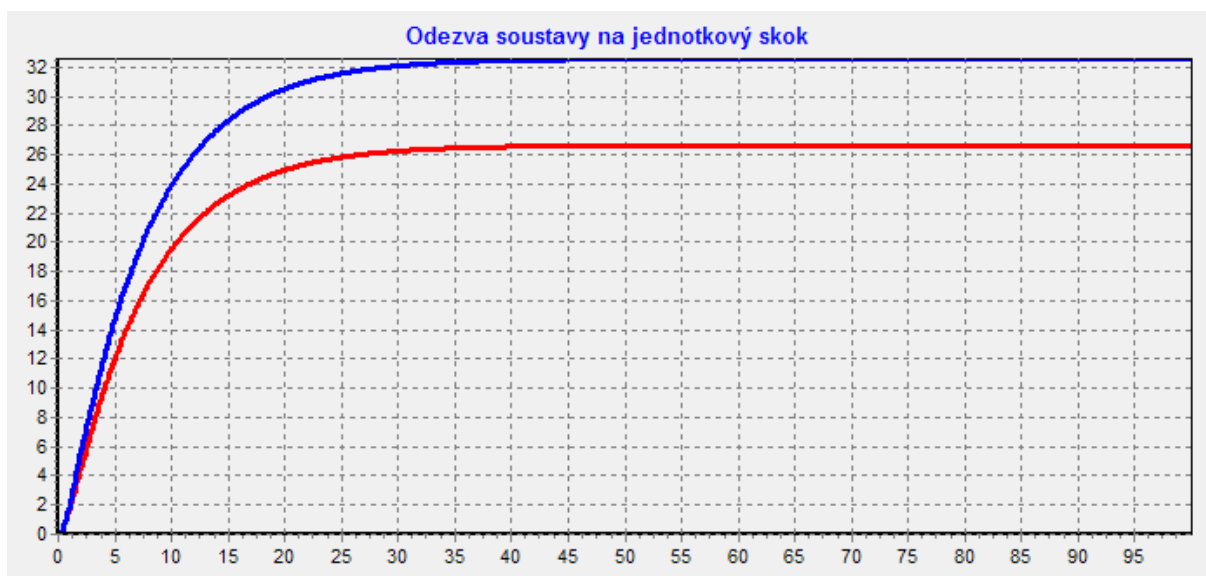
V určitých mezích je možno rozdílnost koeficientů korigovat zapojením dle obrázku 8.10. Úplné praktické zapojení modelu a neřízené soustavy je v obrázku 8.11 – mezi model a reálnou soustavu je nutno zařadit D/A a A/D převodníky.



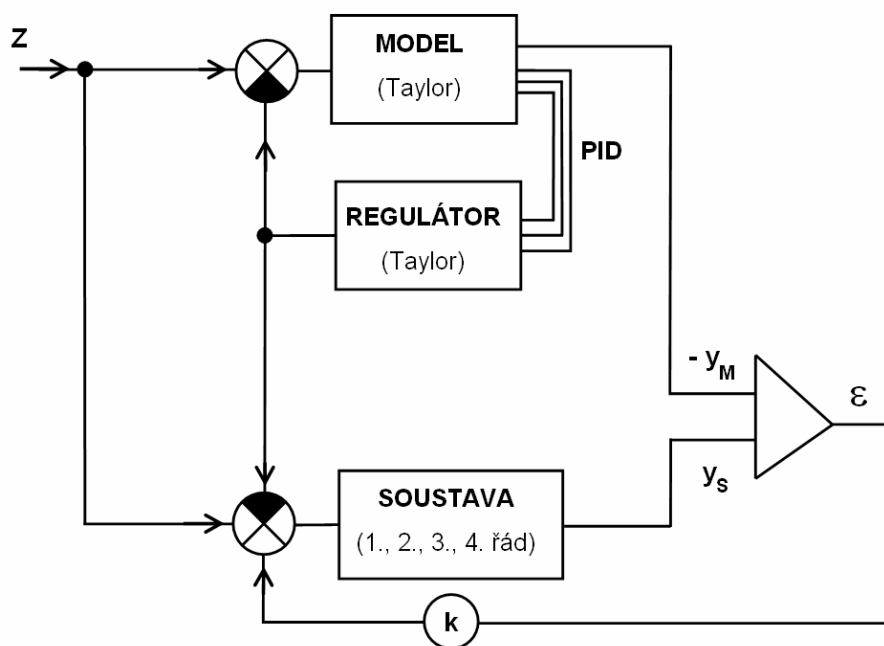
Obrázek 8.7 – Spolupráce modelu a soustavy



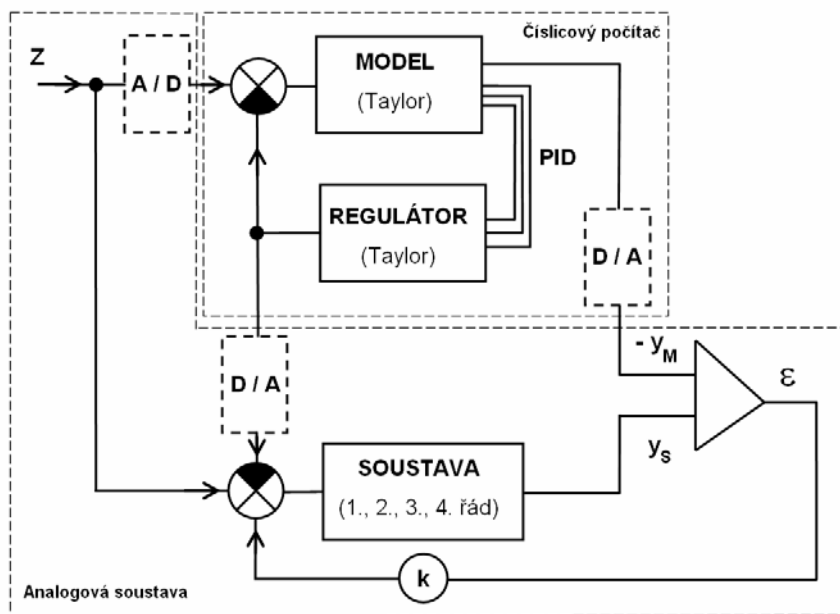
Obrázek 8.8 – Časový průběh y_m a y_s



Obrázek 8.9 – Rozdílné koeficienty



Obrázek 8.10 – Spolupráce modelu a soustavy včetně porovnání odchylky



Obrázek 8.11 – Spolupráce modelu a soustavy včetně A/D a D/A převodníků

Vlastní algoritmus odlišuje reálnou soustavu od modelu: Pro model se používá Taylorova metoda (s nekonečným řádem) a pro reálnou soustavu 1. – 4. řád Taylorovy metody. Následuje soustava všech diferenciálních rovnic:

$$PID=kP*ym+kD*rm+kI*xm$$

$$Zm=vst+PID$$

$$Ym'=((1/dm)*(-em)*ym+rm) \quad Ym(0)=0$$

$$Rm'=((1/dm)*am*zm-(1/dm)*fm*ym+sm) \quad Rm(0)=0$$

$$Sm'=((1/dm)*bm*zm-(1/dm)*gm*ym+Um) \quad Sm(0)=0$$

$$Um'=((1/dm)*cm*zm-(1/dm)*jm*ym) \quad Um(0)=0$$

$$Xm'=ym \quad Xm(0)=0$$

$$Zs=1+PID+k*(ys+ym)$$

$$Ys'=((1/ds)*(-es)*ys+rs) \quad Ys(0)=0$$

$$Rs'=((1/ds)*as*zs-(1/ds)*fs*ys+ss) \quad Rs(0)=0$$

$$Ss'=((1/ds)*bs*zs-(1/ds)*gs*ys+us) \quad Ss(0)=0$$

$$Us'=((1/ds)*cs*zs-(1/ds)*js*ys) \quad Us(0)=0 \quad (8.35)$$

Vzhledem k použití A/D a D/A převodníků, je nutno model normalizovat. Na základě simulace soustavy 9.14 v TKSL byly zjištěny následující maximální hodnoty proměnných

$$UN=150, SN=1290, RN=64, YN=37, ZN=37, PIDN=36$$

Odpovídající normalizace se provede známým postupem. Využije se analogie vztahu (8.36):

$$\left\{ \frac{U}{UN} \right\} = U_N \cdot \quad (8.36)$$

Normalizace rovnic 8.26:

$$U' = \frac{c}{d} z - \frac{j}{d} y \quad (8.37)$$

$$U = \frac{1}{p} \left(\frac{c}{d} z - \frac{j}{d} y \right) \quad (8.38)$$

$$\left\{ \frac{U}{UN} \right\}_{UN} = \frac{1}{p} \left(\frac{c}{d} \left\{ \frac{z}{ZN} \right\}_{ZN} - \frac{j}{d} \left\{ \frac{y}{YN} \right\}_{YN} \right) \quad (8.39)$$

$$U'_N = \left(\frac{c}{d} \frac{ZN}{UN} z_N - \frac{j}{d} \frac{YN}{UN} y_N \right) \quad (8.40)$$

$$S' = \frac{b}{d} z - \frac{g}{d} y \quad (8.41)$$

$$S = \frac{1}{p} \left(\frac{b}{d} z - \frac{g}{d} y \right) \quad (8.42)$$

$$\left\{ \frac{S}{SN} \right\}_{SN} = \frac{1}{p} \left(\frac{b}{d} \left\{ \frac{z}{ZN} \right\}_{ZN} - \frac{g}{d} \left\{ \frac{y}{YN} \right\}_{YN} \right) \quad (8.43)$$

$$S'_N = \left(\frac{b}{d} \frac{ZN}{SN} z_N - \frac{g}{d} \frac{YN}{SN} y_N \right) \quad (8.44)$$

$$R' = \frac{a}{d} z - \frac{f}{d} y \quad (8.45)$$

$$R = \frac{1}{p} \left(\frac{a}{d} z - \frac{f}{d} y \right) \quad (8.46)$$

$$\left\{ \frac{R}{RN} \right\}_{RN} = \frac{1}{p} \left(\frac{a}{d} \left\{ \frac{z}{ZN} \right\}_{ZN} - \frac{f}{d} \left\{ \frac{y}{YN} \right\}_{YN} \right) \quad (8.47)$$

$$R'_N = \left(\frac{a}{d} \frac{ZN}{RN} z_N - \frac{f}{d} \frac{YN}{RN} y_N \right) \quad (8.48)$$

$$Y' = -\frac{e}{d}y \quad (8.49)$$

$$Y = \frac{1}{p} \left(-\frac{e}{d}y \right) \quad (8.50)$$

$$\left\{ \frac{Y}{YN} \right\}_{YN} = \frac{1}{p} \left(-\frac{f}{d} \left\{ \frac{y}{YN} \right\}_{YN} \right) \quad (8.51)$$

$$Y'_N = \left(-\frac{f}{d} \frac{YN}{YN} y_N \right) \quad (8.52)$$

$$Z = vst + PID \quad (8.53)$$

$$\left\{ \frac{Z}{ZN} \right\}_{ZN} = \left(\left\{ \frac{vst}{1} \right\}_1 + \left\{ \frac{PID}{PIDN} \right\}_{PIDN} \right) \quad (8.54)$$

$$Zn = \left(\frac{1}{ZN} vstN + \frac{PIDN}{ZN} PIDn \right) \quad (8.55)$$

$$PID = (kPy + kDr + kIx) \quad (8.56)$$

$$\left\{ \frac{PID}{PIDN} \right\}_{PIDN} = \left(kP \left\{ \frac{y}{YN} \right\}_{YN} + kD \left\{ \frac{r}{RN} \right\}_{RN} + kI \left\{ \frac{x}{XN} \right\}_{XN} \right) \quad (8.57)$$

$$PIDn = \left(kP \frac{YN}{PIDN} y_N + kD \frac{RN}{PIDN} R_N + kI \frac{XN}{PIDN} X_N \right) \quad (8.58)$$

8.3 Implementace programu

Program je implementován v C++ Builderu, který simuluje spolupráci modelu a soustavy popsané v kapitole 8.1.

Program je zhotoven ve dvou verzích. Verze 1 neobsahuje D/A a A/D převodníky mezi modelem a reálnou soustavou a verze 2 obsahuje D/A a A/D převodníky. Vlastní algoritmus programu počítá soustavu (8.13) u verze 1 a soustavu (8.14) u verze 2. Pro výpočet se používá Taylorova metoda.

8.3.1 Implementace Taylorovy metody

- vynulování všech proměnných
- cyklus probíhající od 0 do Tmax s krokem dt
- první krok s počátečními podmínkami 0
- do celkového výsledku se přičtou výsledky dílčích výpočtů
- cyklus probíhající tak dlouho, dokud všechny přírůstky jednotlivých proměnných nejsou menší než zadaná přesnost EPS
- další kroky výpočtu s tím, že výpočet rovnic popisujících reálnou soustavu, je omezen na zvolený řád. (Neprovede se víc kroků, než je zvoleno)
- do celkového výsledku se přičtou výsledky dílčích výpočtů
- výsledek se vynese do grafu
- do proměnných pro počáteční podmínky se uloží současné výsledky a hlavní cyklus se opakuje

První krok výpočtu:

$$\begin{aligned}PID &= kP * ym0 + kD * rm0 + kI * xm0 \\DZm &= vst + PID \\DYm &= h * ((-em/dm) * ym0 + rm0) \\DRm &= h * ((am/dm) * DZm - (fm/dm) * ym0 + sm0) \\DSm &= h * ((bm/dm) * DZm - (gm/dm) * ym0 + um0) \\DUm &= h * ((cm/dm) * DZm - (jm/dm) * ym0) \\DXm &= h * ym0 \\DZs &= vst + PID + k * (ys0 + ym0) \\DYS &= h * ((-es/ds) * ys0 + rs0) \\DRs &= h * ((as/ds) * DZs - (fs/ds) * ys0 + ss0) \\DSs &= h * ((bs/ds) * DZs - (gs/ds) * ys0 + us0) \\DUs &= h * ((cs/ds) * DZs - (js/ds) * ys0)\end{aligned}$$

(8.60)

Další kroky výpočtu:

$$\begin{aligned}
 PID &= kP * DYm + kD * DRm + kI * DXm \\
 DZm &= PID \\
 DYm1 &= h1 * ((-em/dm) * DYm + DRm) \\
 DRm1 &= h1 * ((am/dm) * DZm - (fm/dm) * DYm + DSm) \\
 DSm1 &= h1 * ((bm/dm) * DZm - (gm/dm) * DYm + DUm) \\
 DUm1 &= h1 * ((cm/dm) * DZm - (jm/dm) * DYm) \\
 DXm1 &= h1 * DYm
 \end{aligned}$$

$$\begin{aligned}
 DZs &= PID + k * (DYS + DYm) \\
 DYS1 &= h1 * ((-es/ds) * DYS + DRs) \\
 DRs1 &= h1 * ((as/ds) * DZs - (fs/ds) * DYS + DSSs) \\
 DSSs1 &= h1 * ((bs/ds) * DZs - (gs/ds) * DYS + DUSs) \\
 DUSs1 &= h1 * ((cs/ds) * DZs - (js/ds) * DYS)
 \end{aligned}$$

(8.61)

Kde $h1$ je $\frac{h}{2}, \frac{h}{3}, \frac{h}{4}, \frac{h}{5} \dots \frac{h}{n}$ a n je řád metody.

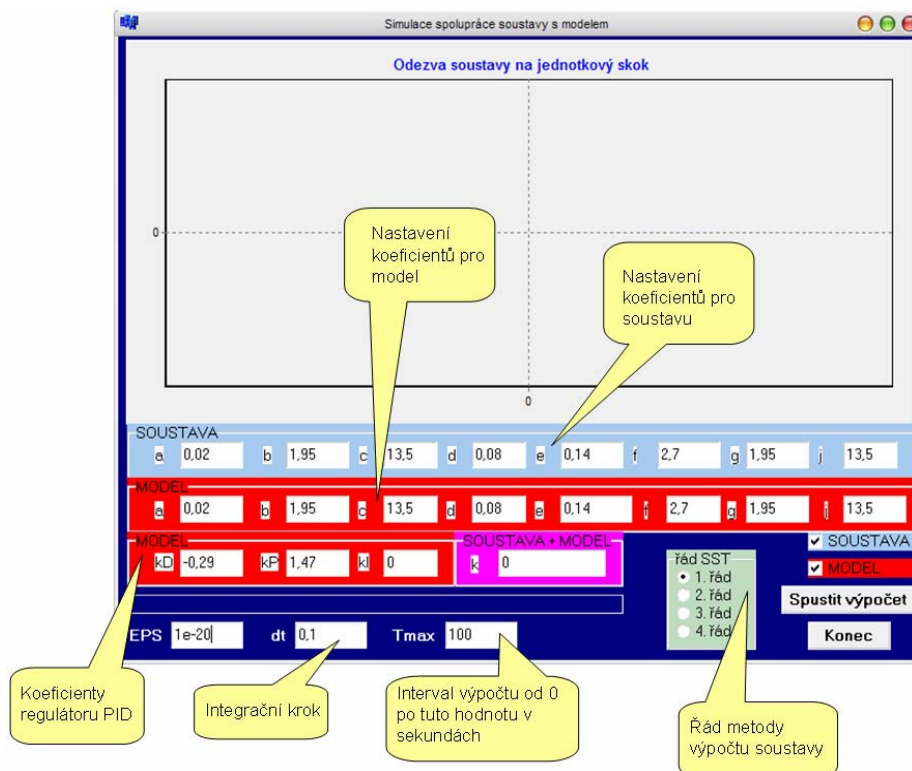
8.3.2 Implementace A/D a D/A převodníků

D/A a A/D převodníky jsou simulovány v následujících experimentech jako 4, 8 a 12-ti bitové.

8.4 Ovládání programu

Obě verze programu se spouští z prostředí Windows souborem Soustava1.exe v případě verze 1 a Soustava2.exe v případě verze 2. Po spuštění naběhne základní okno aplikace, které se skládá z grafu a části pro nastavení parametrů.

8.4.1 Program verze 1



Obrázek 8.13 - Základní okno programu verze 1

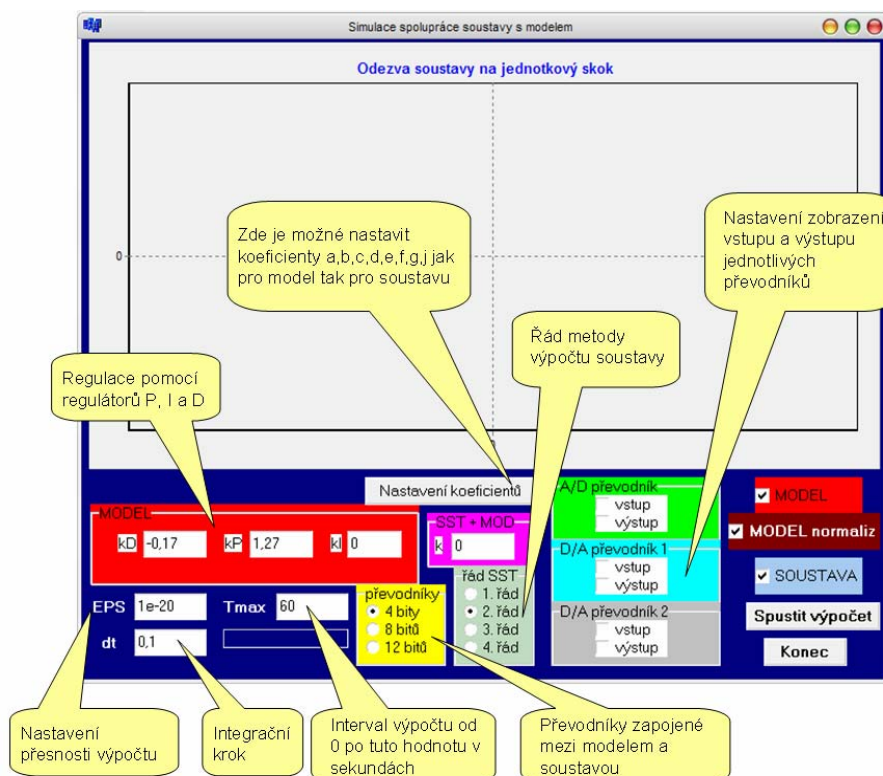
Na obrázku 8.13 je zobrazen popis nejdůležitějších nastavení simulace. Je lze možno nastavit koeficienty a až j samostatně jak pro model tak i pro soustavu a koeficienty regulátoru P, I, D. Implicitní hodnoty koeficientů jsou tyto: Model a soustava – $a=0,02$, $b=1,95$, $c=13,5$, $d=0,08$, $e=0,14$, $f=2,7$, $g=1,95$, $j=13,5$. Dále koeficienty regulátoru – $kD=-0,29$, $kP=1,47$ a $kI=0$. Koeficient k je násobek rozdílu výstupů modelu a soustavy, který se přivádí zpět na vstup soustavy a je implicitně nastaven na 0. V našem případě je možné tímto koeficientem regulovat případnou odchylku mezi soustavou a modelem. EPS značí nastavení přesnosti výpočtu, tzn. je to maximální hodnota přírůstku po kterém může být výpočet ukončen a je implicitně nastaven na $1 \cdot 10^{-20}$. Okénko dt - Integrační krok, který nese implicitní hodnotu 0,1. $Tmax$ je maximální hodnota simulačního času v našem případě přednastavena na 100. Dále je možné nastavit řád metody, kterou bude počítána soustava rovnic popisující reálnou soustavu. 1. řád představuje Eulerovu metodu, 2. – 4. metodu Runge-Kutta (Implicitně Eulerova metoda 1. řádu). Kliknutím na tlačítko spustit výpočet se zahájí proces simulace. Kliknutím na tlačítko Konec dojde k ukončení programu. Po dokončení výpočtu se zobrazí průběh výstupů soustavy a modelu. Průběh bude zobrazen,

pokud je zaškrtnut příslušný CheckBox (Implicitně Soustava i Model). Dále je možné graf zoomovat.

Zoomování probíhá takto: Levým tlačítkem myši označíme shora dolů danou oblast, kterou chceme zoomovat tak, že při označování držíme levé tlačítko. Po označení oblasti uvolníme tlačítko a daná oblast se zvětší na celý rozměr grafu. Pokud se chceme vrátit k původnímu měřítku, opakujeme tento proces s tím rozdílem, že označujeme zdola nahoru. Chceme-li graf posunovat, při stejném měřítku, pravým tlačítkem myši klikneme na kterékoliv místo v grafu a tažením myši se zmačknutým tlačítkem posunujeme libovolně graf. Po uvolnění tlačítka zůstane graf v poslední zobrazované pozici.

Pokud byl výpočet již jednou spuštěn a změní se řád soustavy, není nutné již klikat na tlačítko „spustit výpočet“, neboť se soustava automaticky přepočítá. Ve spodní části okna je ukazatel, který během simulace signalizuje v jaké fázi se výpočet nachází.

8.4.2 Program verze 2



Obrázek 8.14 - Základní okno programu verze 2



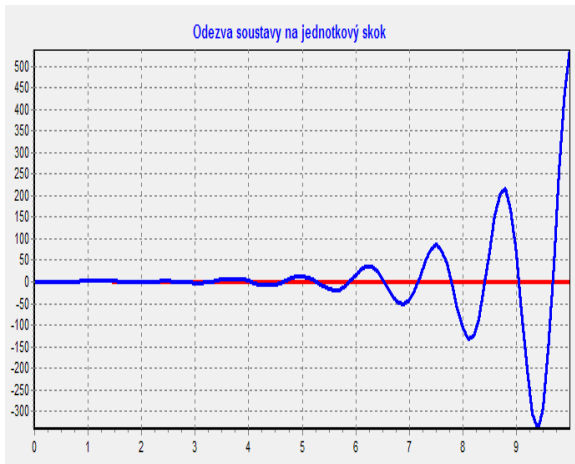
Obrázek 8.15 - Dialogové okno pro nastavení koeficientů u programu verze 2

Na obrázku 8.14 je zobrazen popis nejdůležitějších nastavení simulace. Koeficienty se narozdíl od verze 1 nastavují v samostatném dialogovém okně (obrázek 8.15), které se vyvolá kliknutím na tlačítko „nastavení koeficientů“. Opět je možné nastavit koeficienty a až j samostatně jak pro model tak i pro soustavu. Implicitní hodnoty jsou stejné jako u verze 1. V dialogovém okně se nachází tlačítka OK a STORNO. Tlačítkem OK se potvrzuje aktuální nastavení koeficientů. Tlačítkem STORNO se dostaneme do základního okna bez uložení změn v koeficientech. Význam koeficientu k je stejný jako u verze 1; implicitně nastaven opět na 0. EPS, dt, Tmax jsou nastaveny stejně a mají stejnou funkci jako ve verzi 1. řád metody se opět nijak neliší od předchozí verze. Verze 2 obsahuje navíc i jeden analogově digitální a dva digitálně analogové převodníky u kterých je možné nastavit jejich rozlišovací schopnost. Dostupné možnosti jsou: 4 bity, 8 bitů, 12 bitů (Implicitně 4 bity).

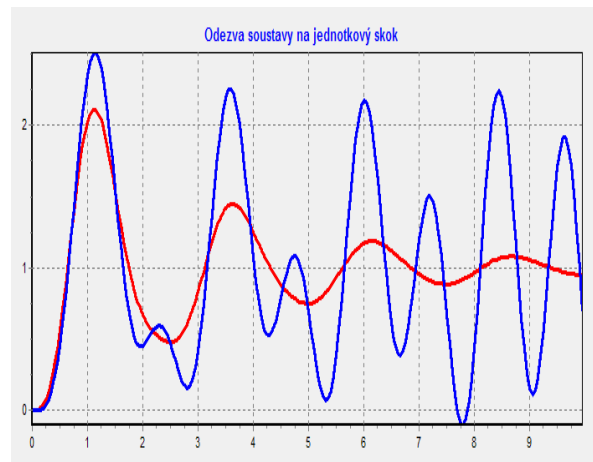
Kliknutím na tlačítko spustit výpočet se zahájí proces simulace. Kliknutím na tlačítko Konec dojde k ukončení programu. Po dokončení výpočtu se zobrazí zvolené průběhy. Kromě výstupu modelu a výstupu soustavy je v této verzi programu možné zobrazit i výstup normalizovaného modelu a vstupy a výstupy všech převodníků. Dále je možné graf zoomovat a posunovat obdobně jako u předchozí verze.

Následují důležité experimenty s modelem a se zhodnocením kvality regulace. Experimenty mají výukový charakter s očekávanými výsledky. Ve všech případech je důležité pracovat velkým řádem metody a vhodným integračním krokem. Pokud je integrační krok nevhodně zvolen, objeví se podle očekávání numerická nestabilita

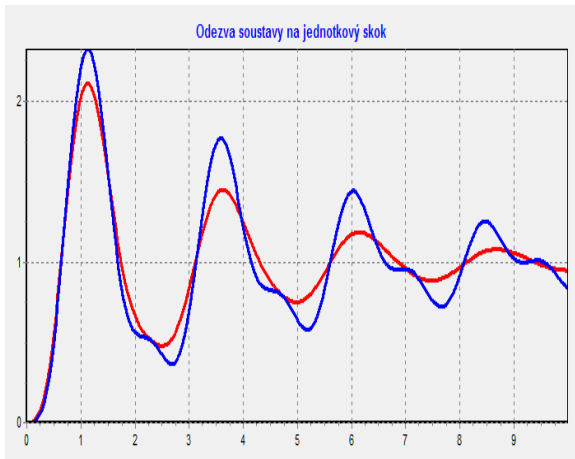
Soustava bez regulace – počítáno Eulerovou metodou (1. řád):



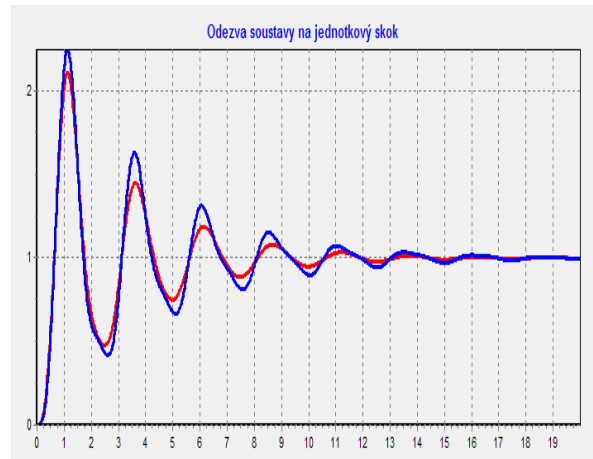
Obrázek 8.16 – $dt = 0,1s$



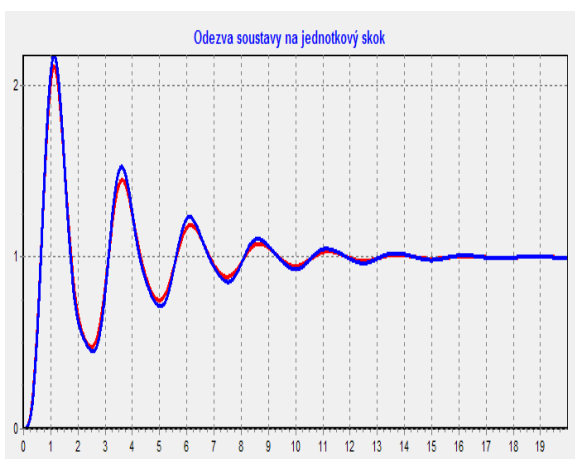
Obrázek 8.17 – $dt = 0,05s$



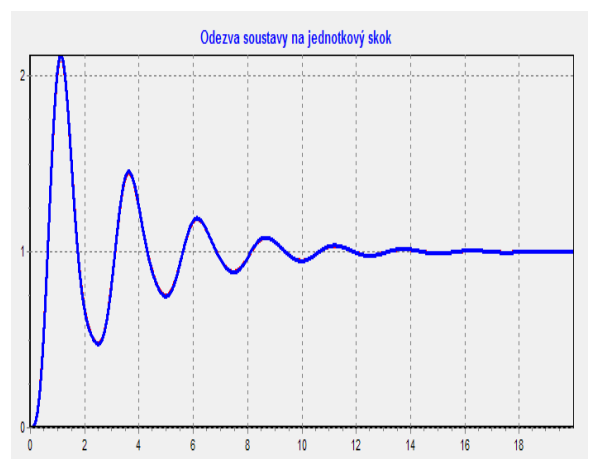
Obrázek 8.18 – $dt = 0,03s$



Obrázek 8.19 – $dt = 0,02s$

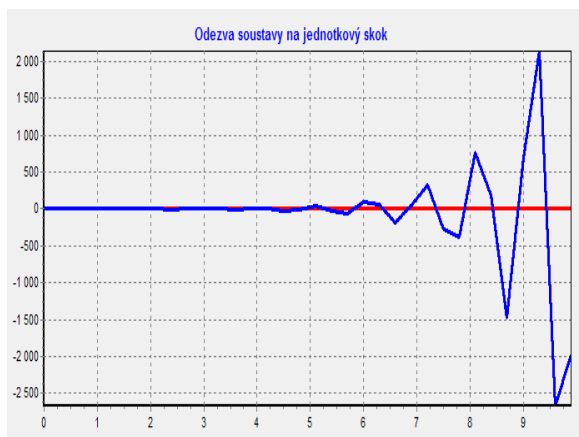


Obrázek 8.20 – $dt = 0,01s$

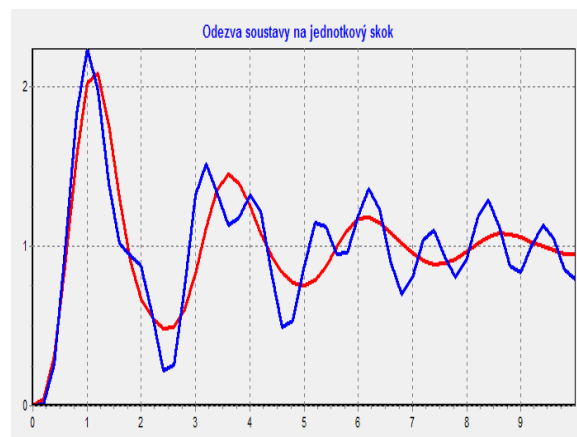


Obrázek 8.21 – $dt = 0,001s$

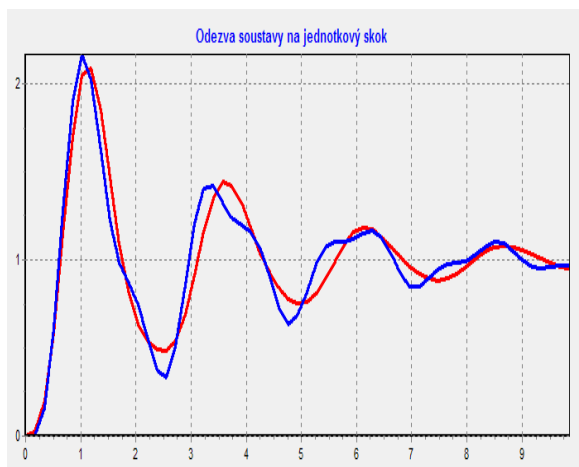
Soustava bez regulace – počítáno metodou Runge-Kutta 2. řádu:



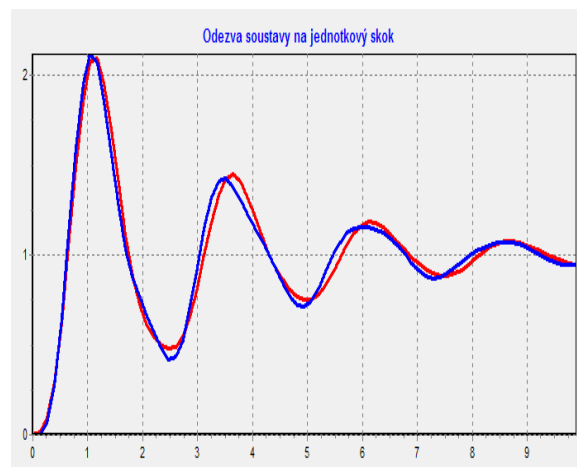
Obrázek 8.22 – $dt = 0,3s$



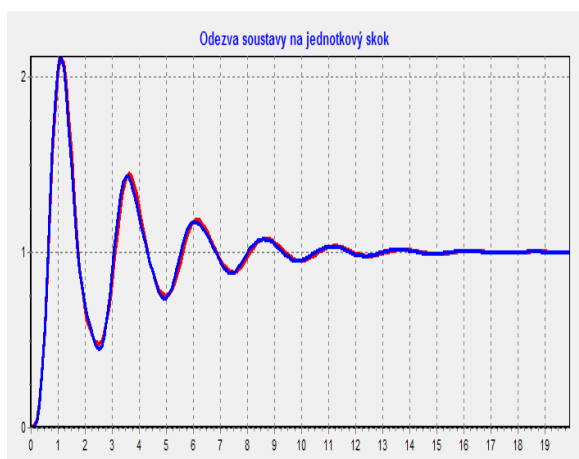
Obrázek 8.23 – $dt = 0,2s$



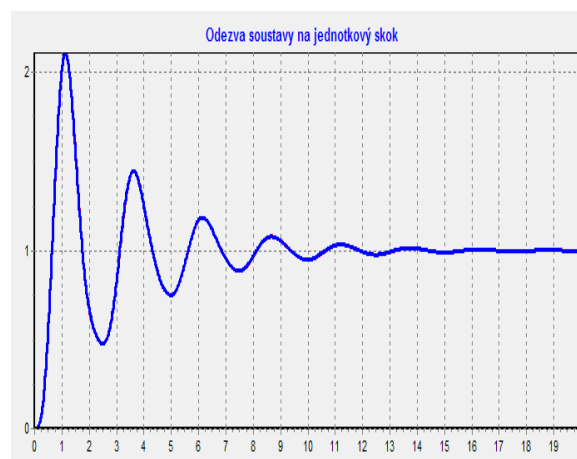
Obrázek 8.24 – $dt = 0,17s$



Obrázek 8.25 – $dt = 0,13s$

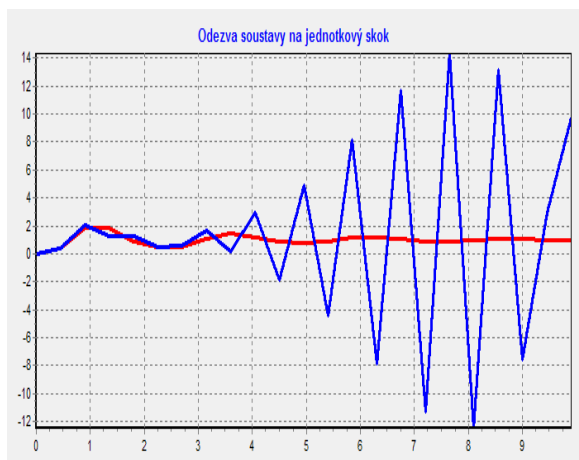


Obrázek 8.26 – $dt = 0,17s$

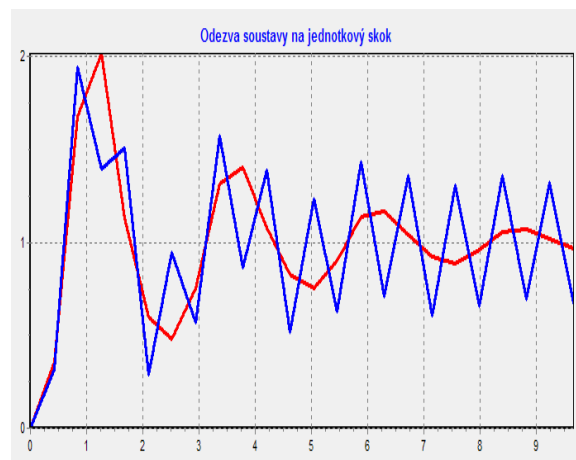


Obrázek 8.27 – $dt = 0,13s$

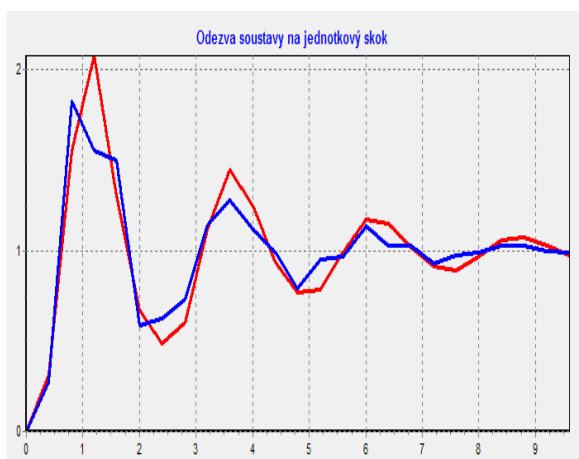
Soustava bez regulace – počítáno metodou Runge-Kutta 3. řádu:



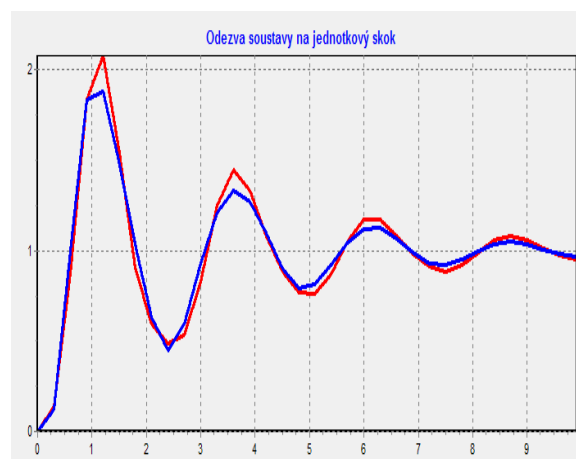
Obrázek 8.28 – $dt = 0,45s$



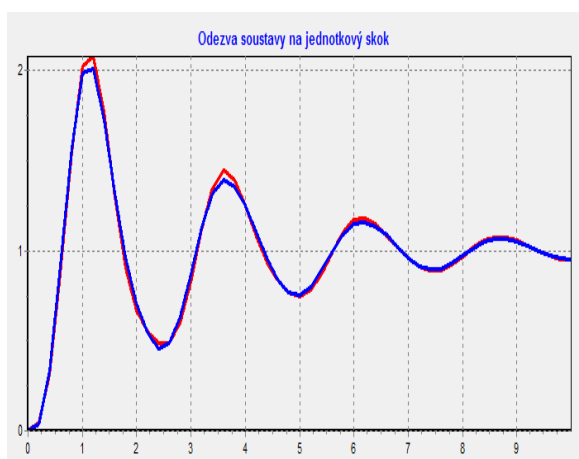
Obrázek 8.29 – $dt = 0,42s$



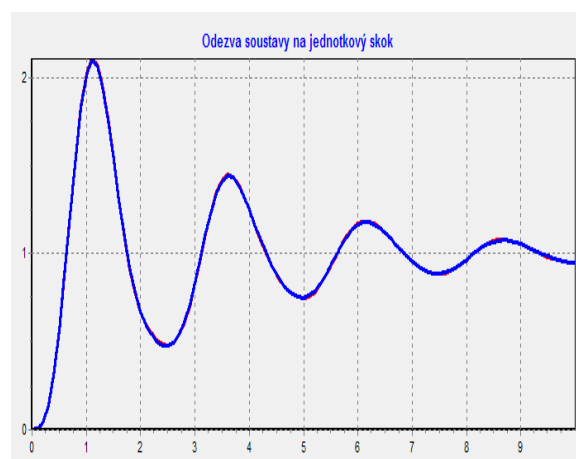
Obrázek 8.30 – $dt = 0,40s$



Obrázek 8.31 – $dt = 0,30s$

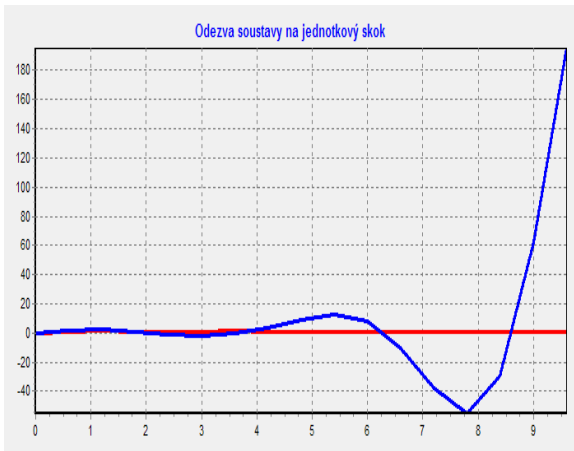


Obrázek 8.32 – $dt = 0,20s$

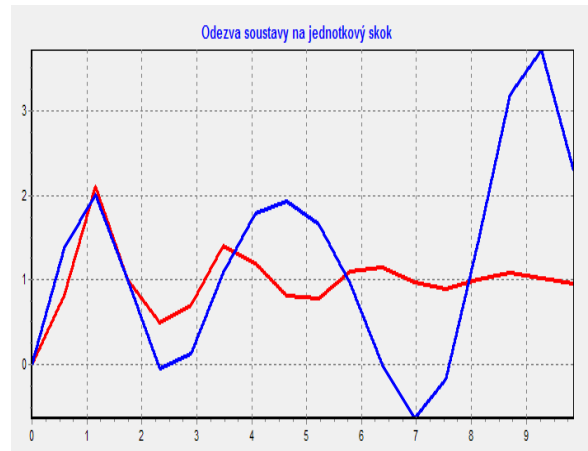


Obrázek 8.33 – $dt = 0,10s$

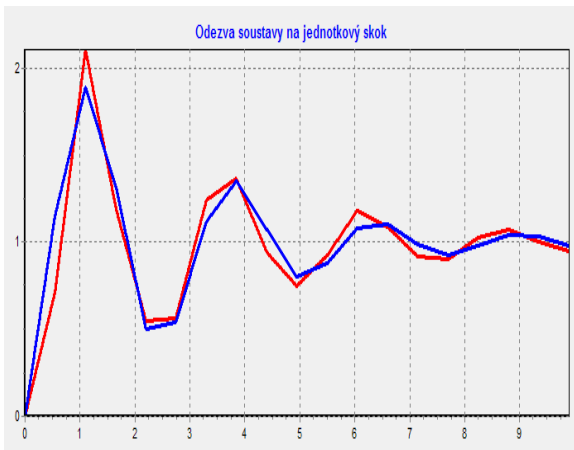
Soustava bez regulace – počítáno metodou Runge-Kutta 4. řádu:



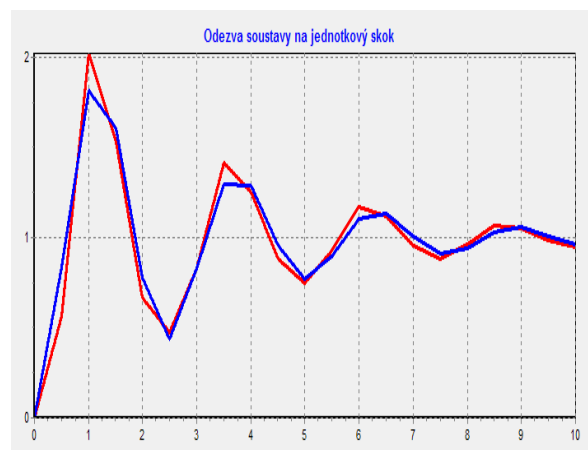
Obrázek 8.34 – $\Delta t = 0,60s$



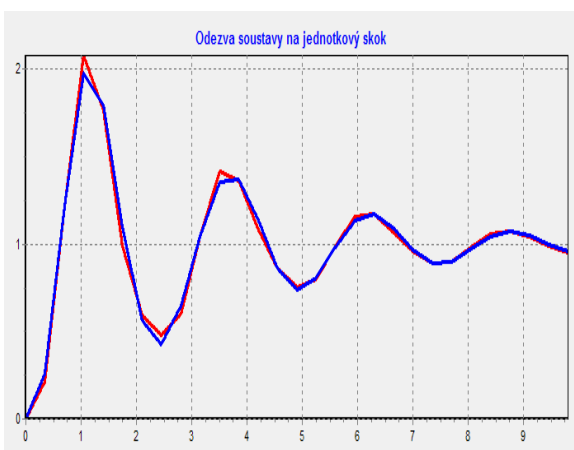
Obrázek 8.35 – $\Delta t = 0,58s$



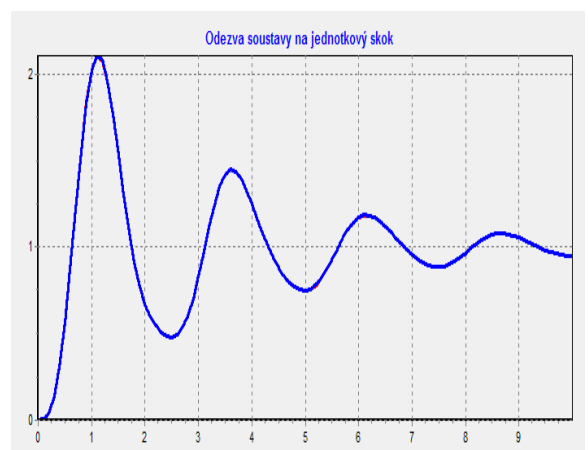
Obrázek 8.36 – $\Delta t = 0,55s$



Obrázek 8.37 – $\Delta t = 0,50s$

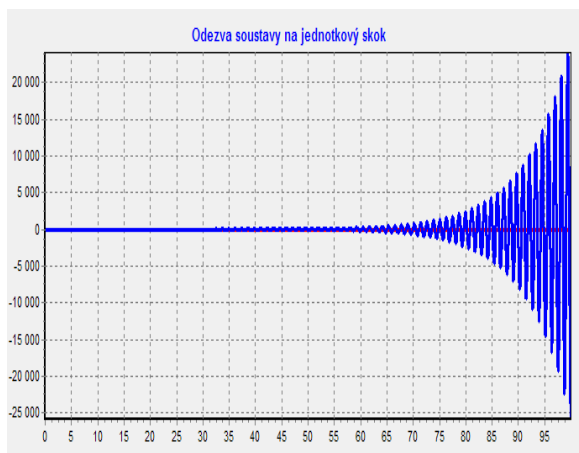


Obrázek 8.38 – $\Delta t = 0,35s$

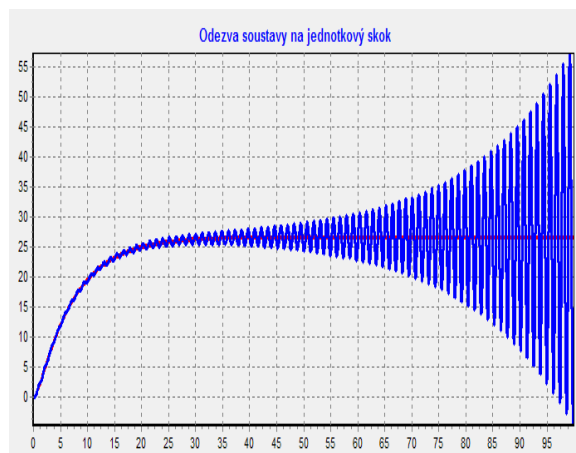


Obrázek 8.39 – $\Delta t = 0,10s$

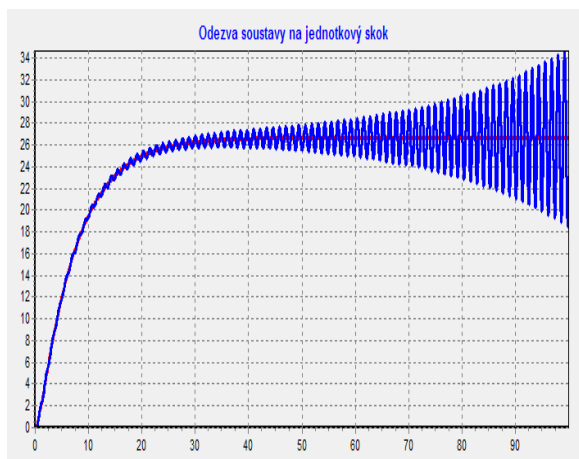
Soustava s regulací (P,D) – počítáno Eulerovou metodou (1. řád):



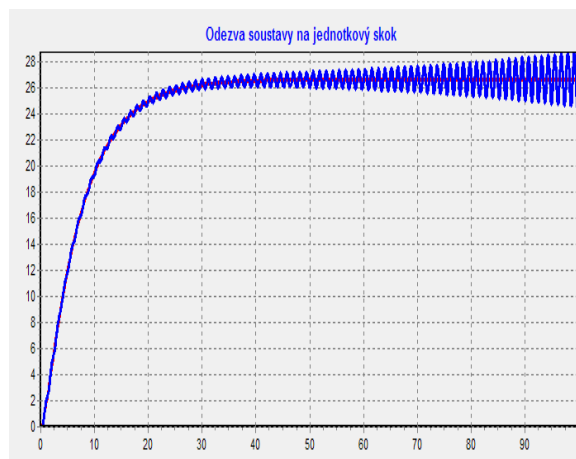
Obrázek 8.40 – $dt = 0,50s$



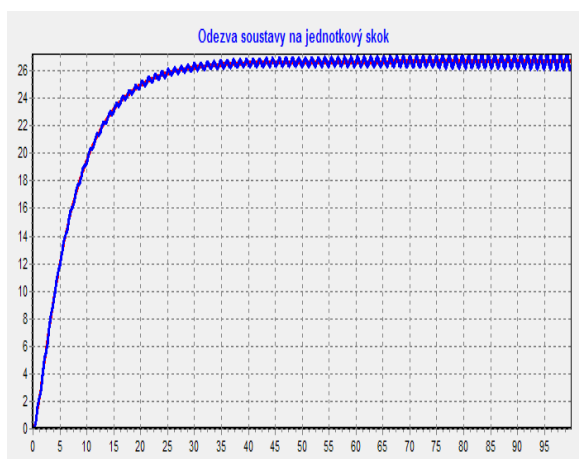
Obrázek 8.41 – $dt = 0,45s$



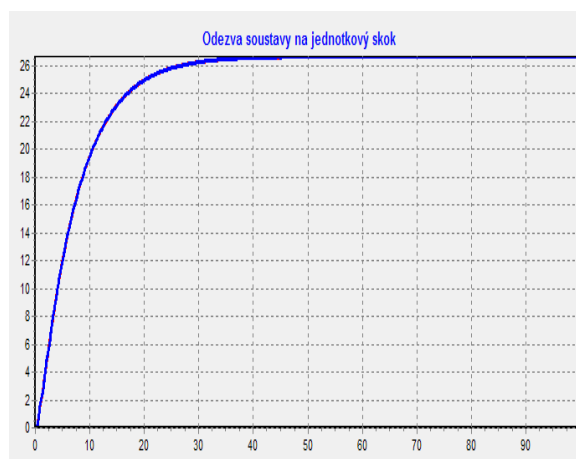
Obrázek 8.42 – $dt = 0,44s$



Obrázek 8.43 – $dt = 0,43s$

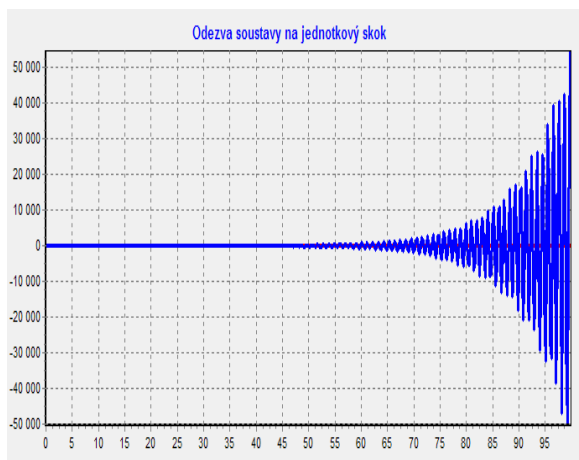


Obrázek 8.44 – $dt = 0,42s$

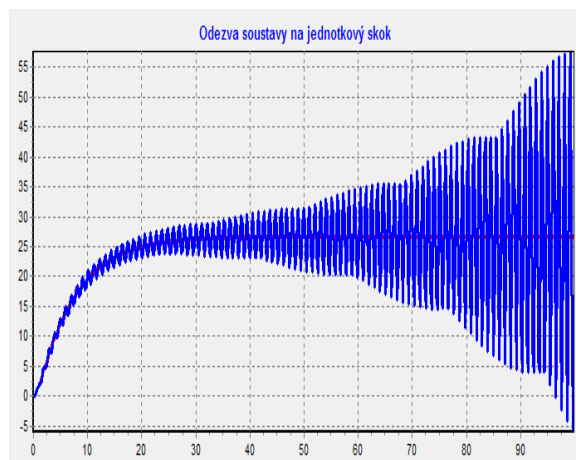


Obrázek 8.45 – $dt = 0,35s$

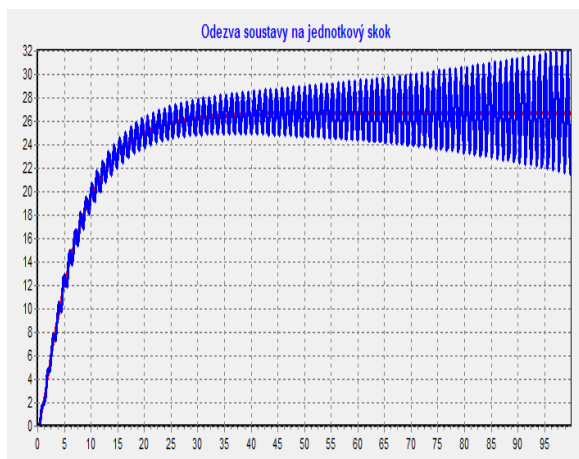
Soustava s regulací (P,D) – počítáno metodou Runge-Kutta 2. řádu:



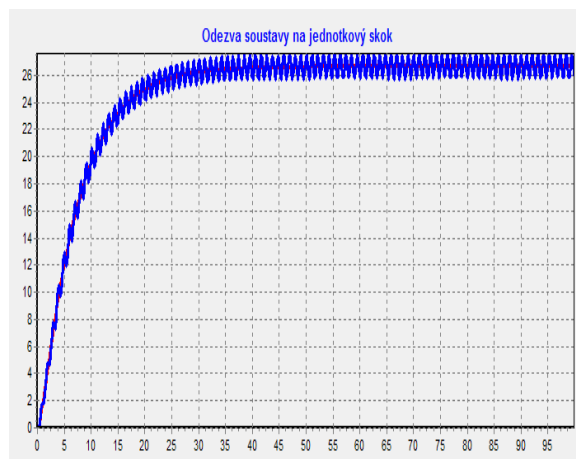
Obrázek 8.46 – $dt = 0,220s$



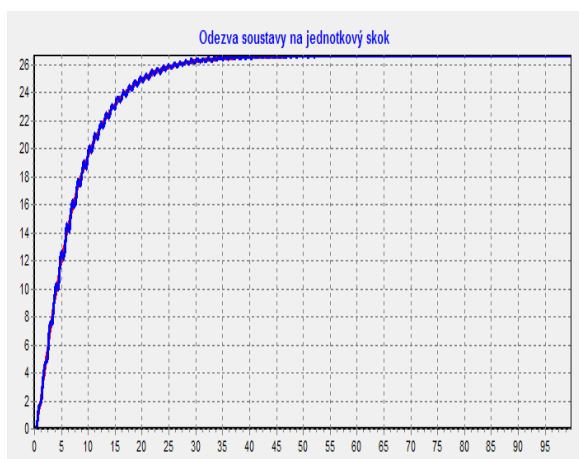
Obrázek 8.47 – $dt = 0,212s$



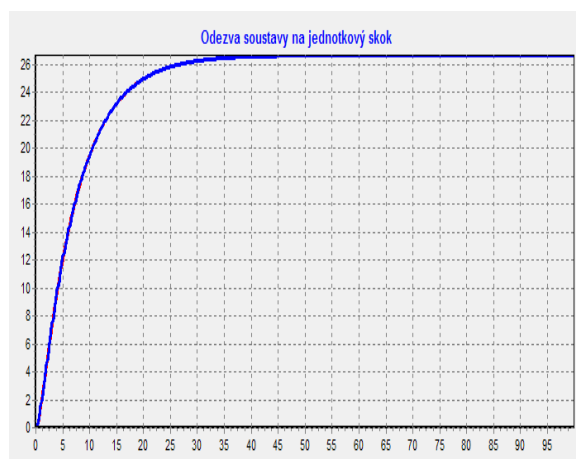
Obrázek 8.48 – $dt = 0,210s$



Obrázek 8.49 – $dt = 0,208s$

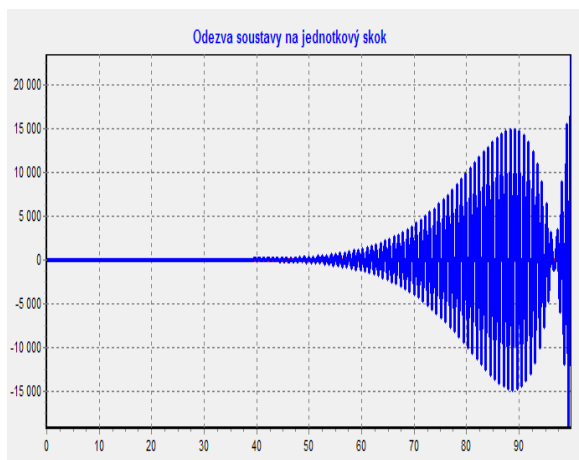


Obrázek 8.50 – $dt = 0,200s$

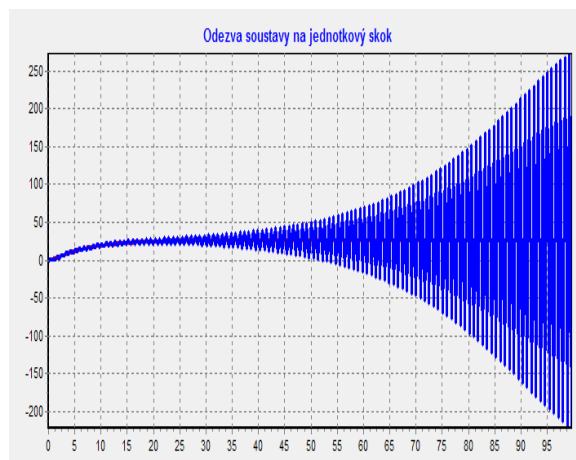


Obrázek 8.51 – $dt = 0,180s$

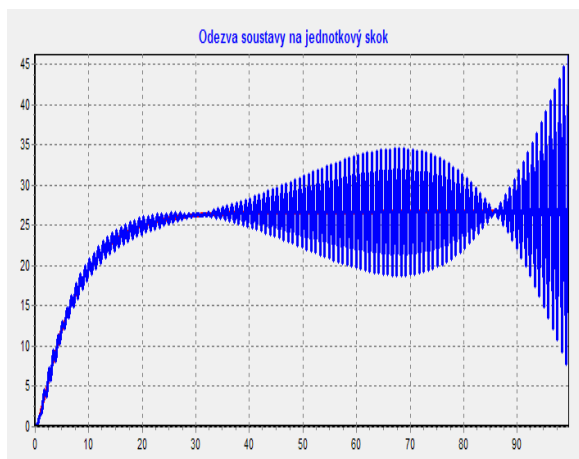
Soustava s regulací (P,D) – počítáno metodou Runge-Kutta 3. řádu:



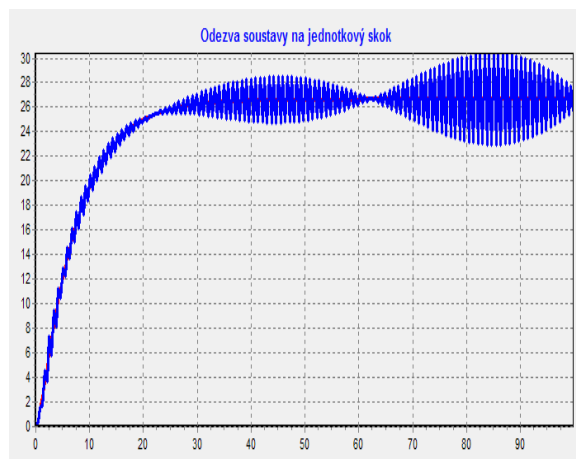
Obrázek 8.52 – $dt = 0,427s$



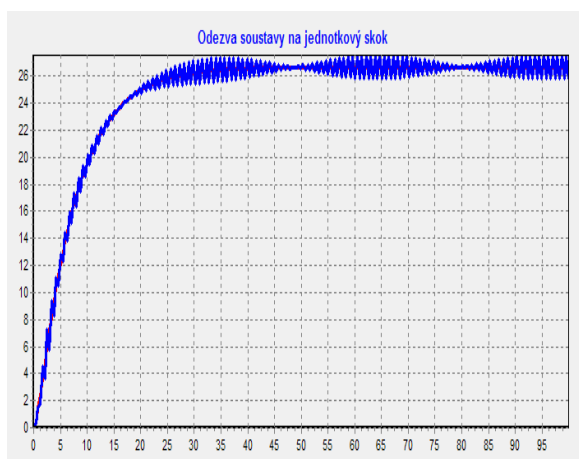
Obrázek 8.53 – $dt = 0,424s$



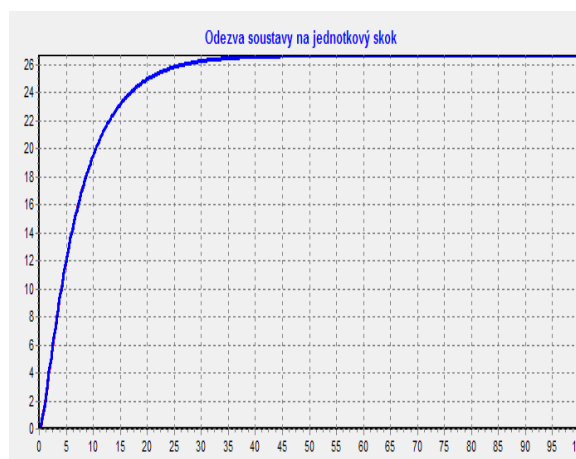
Obrázek 8.54 – $dt = 0,422s$



Obrázek 8.55 – $dt = 0,421s$

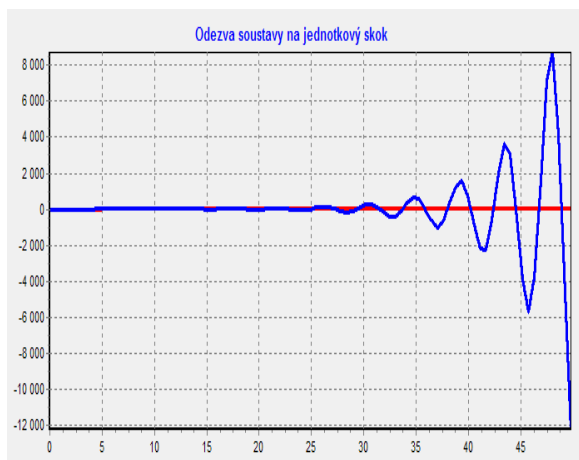


Obrázek 8.56 – $dt = 0,420s$

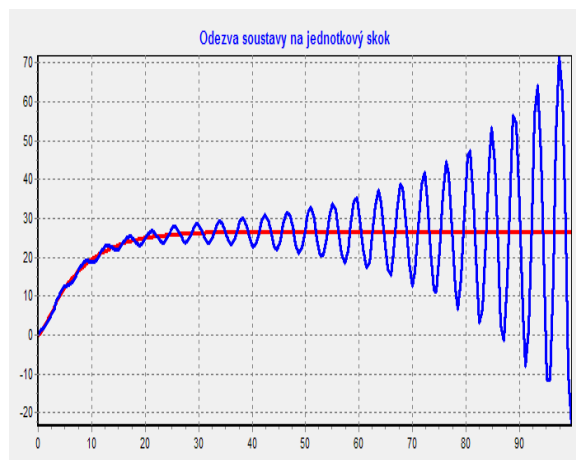


Obrázek 8.57 – $dt = 0,250s$

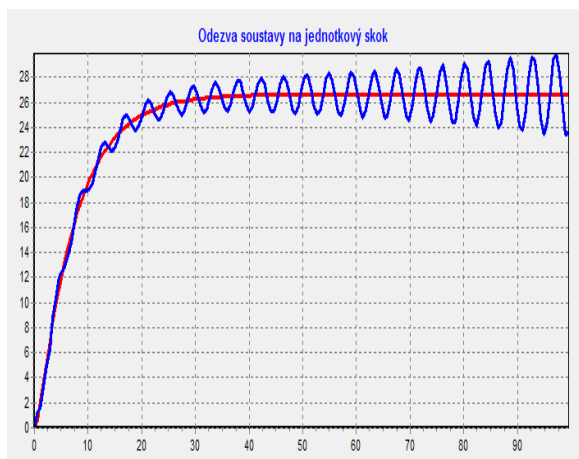
Soustava s regulací (P,D) – počítáno metodou Runge-Kutta 4. řádu:



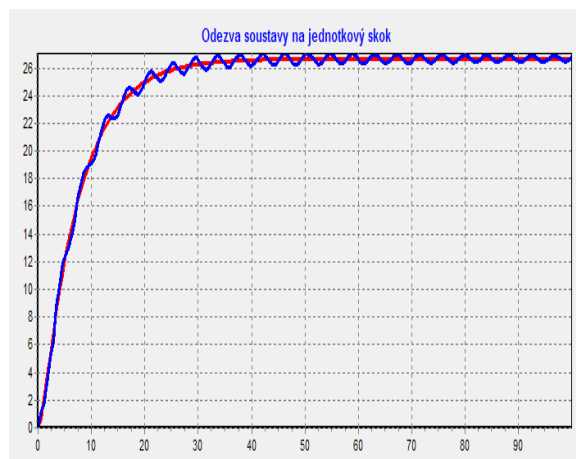
Obrázek 8.58 – $dt = 0,579s$



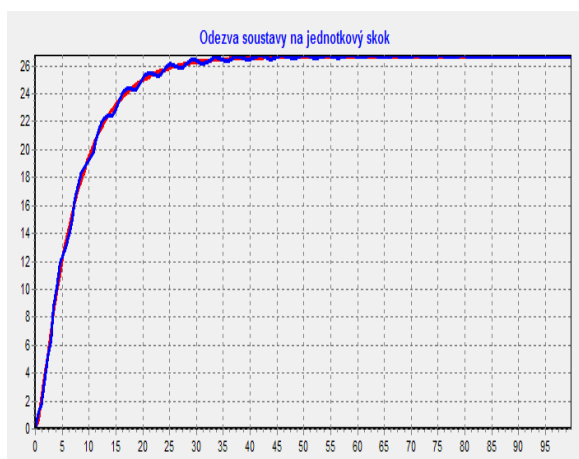
Obrázek 8.59 – $dt = 0,573s$



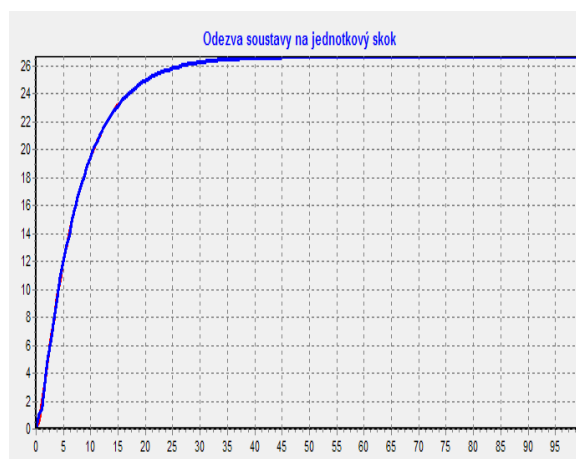
Obrázek 8.60 – $dt = 0,572s$



Obrázek 8.61 – $dt = 0,571s$



Obrázek 8.62 – $dt = 0,570s$



Obrázek 8.63 – $dt = 0,550s$

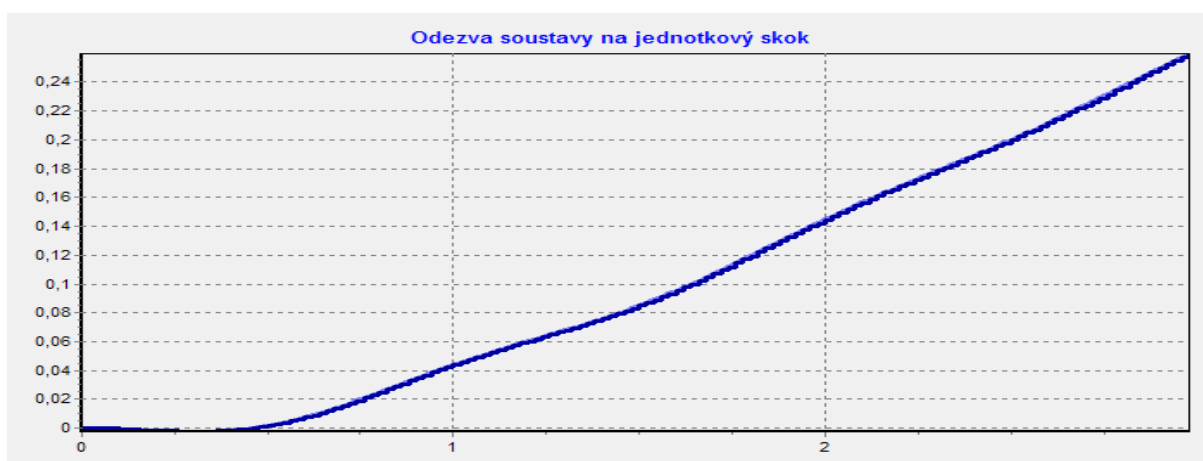
Zavedení převodníků A/D a D/A do verze s regulátorem (P,D) počítáno 4. řádem (Runge-Kutta):



Obrázek 8.64 – 4 bitový převodník

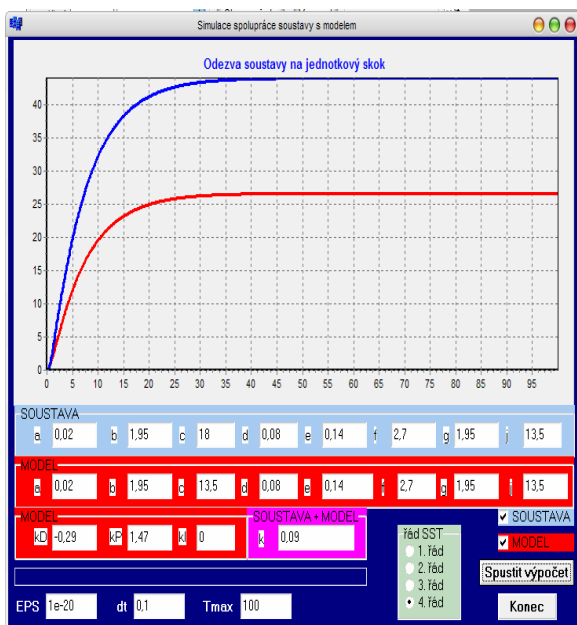


Obrázek 8.65 – 8 bitový převodník

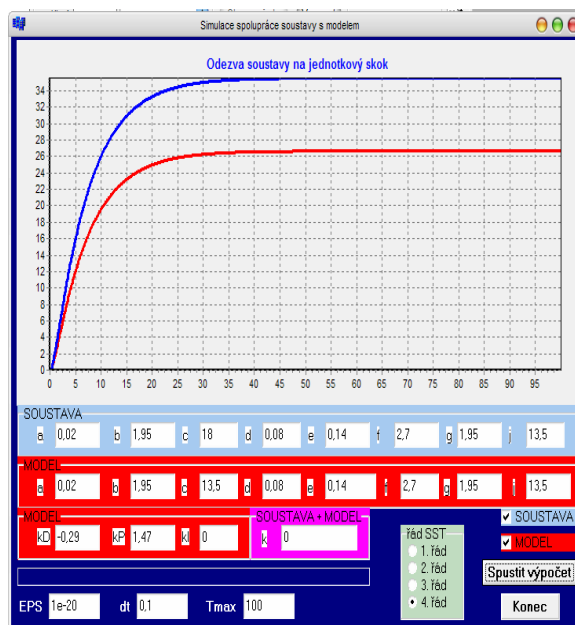


Obrázek 8.66 – 12 bitový převodník

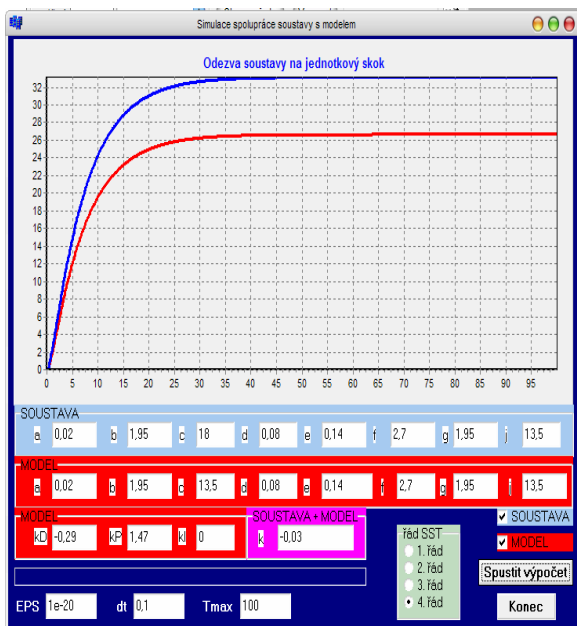
Vliv koeficientu k na model a soustavu – počítáno metodou Runge-Kutta 4. řádu:



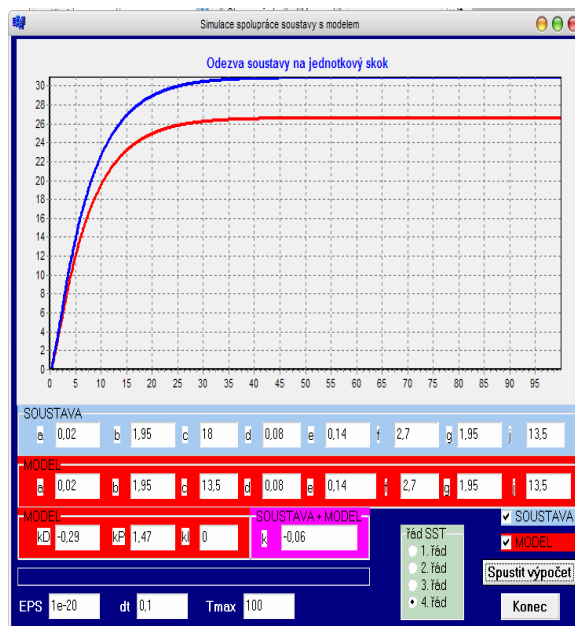
Obrázek 8.67 – $k = 0,09$



Obrázek 8.68 – $k = 0$

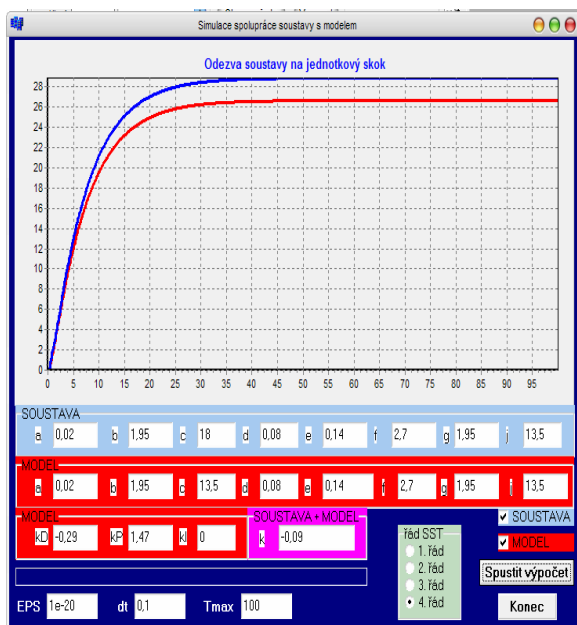


Obrázek 8.69 – $k = -0,03$

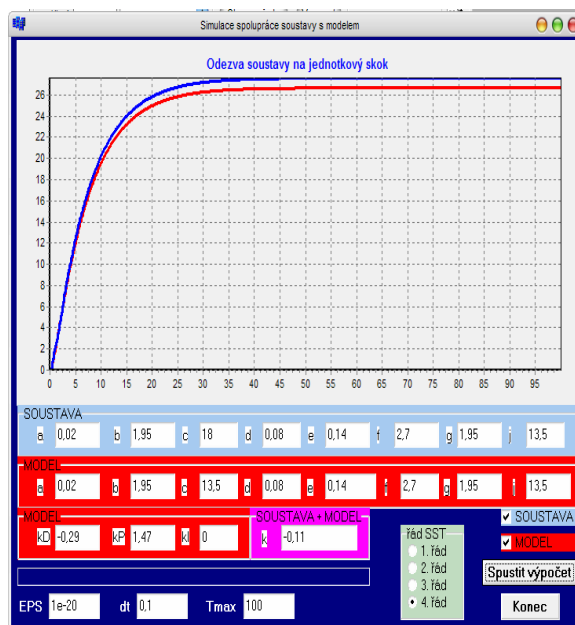


Obrázek 8.70 – $k = -0,06$

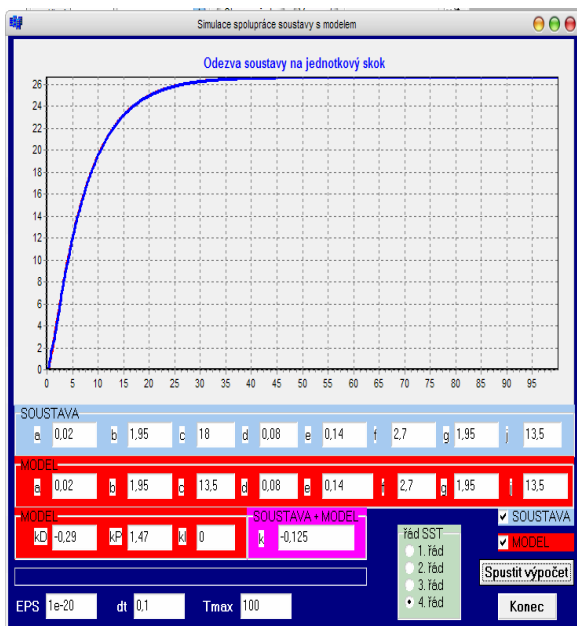
Vliv koeficientu k na model a soustavu – počítáno metodou Runge-Kutta 4. řádu:



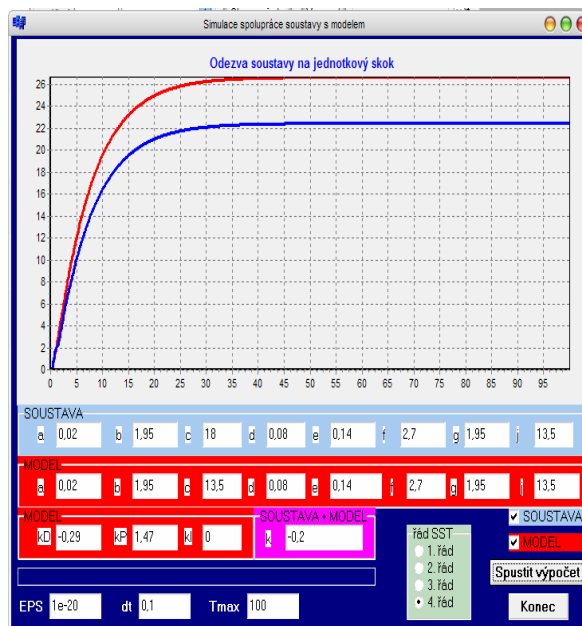
Obrázek 8.71 – $k = 0,09$



Obrázek 8.72 – $k = -0,11$

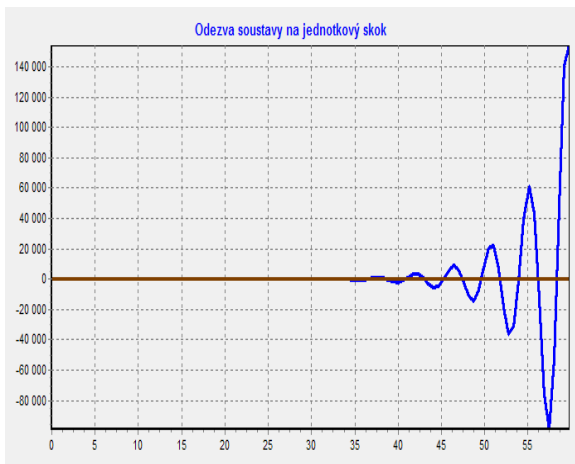


Obrázek 8.73 – $k = -0,125$

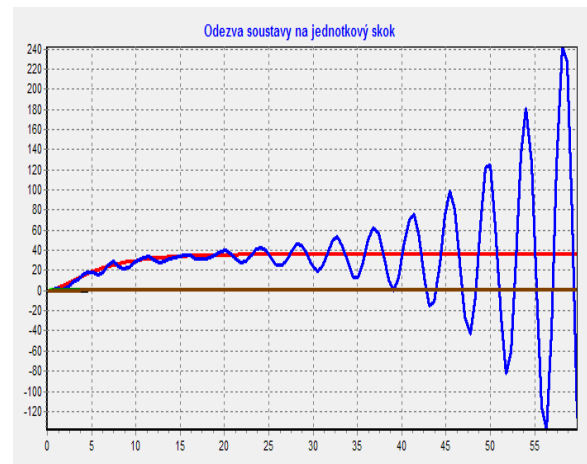


Obrázek 8.74 – $k = -0,200$

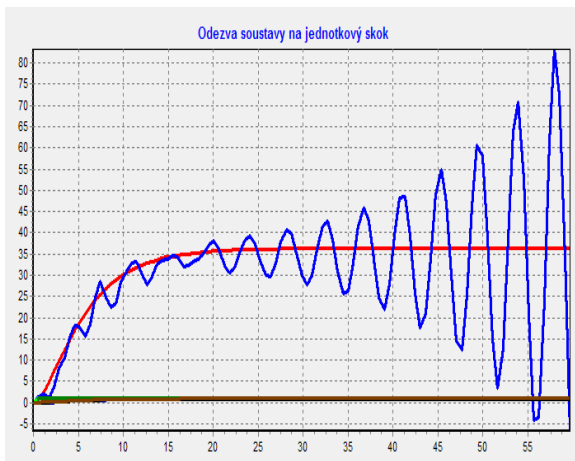
Soustava s regulací (P,D) se 4 bitovými převodníky – počítáno metodou Runge-Kutta 4. řádu:



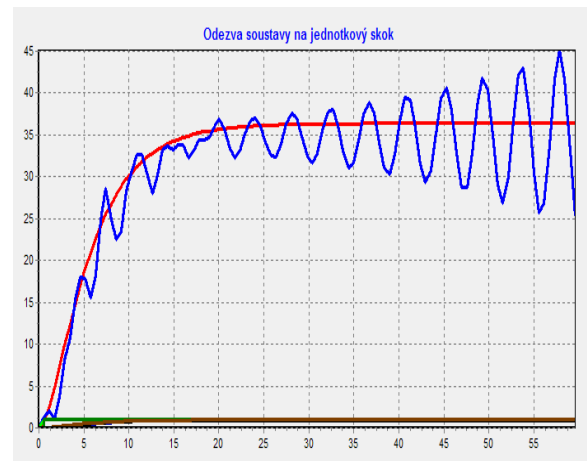
Obrázek 8.75 – $\Delta t = 0,580s$



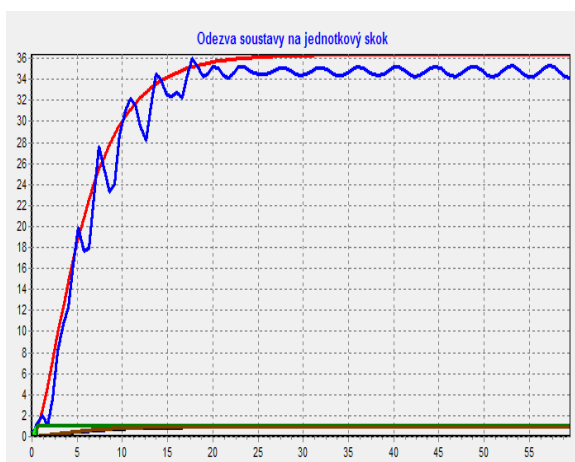
Obrázek 8.76 – $\Delta t = 0,575s$



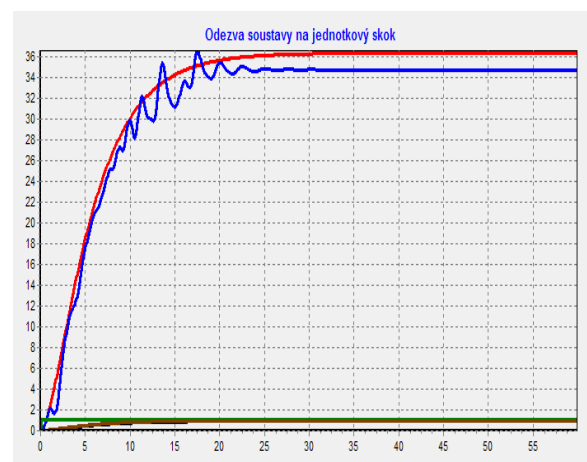
Obrázek 8.77 – $\Delta t = 0,574s$



Obrázek 8.78 – $\Delta t = 0,573s$

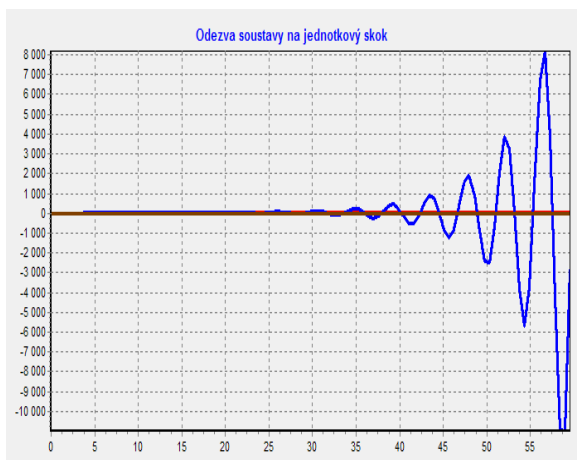


Obrázek 8.79 – $\Delta t = 0,572s$

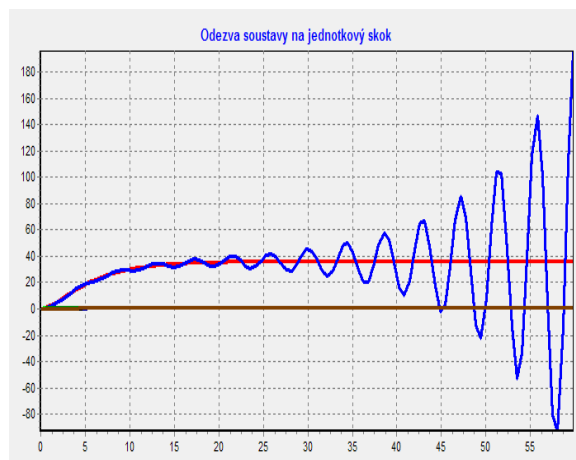


Obrázek 8.80 – $\Delta t = 0,100s$

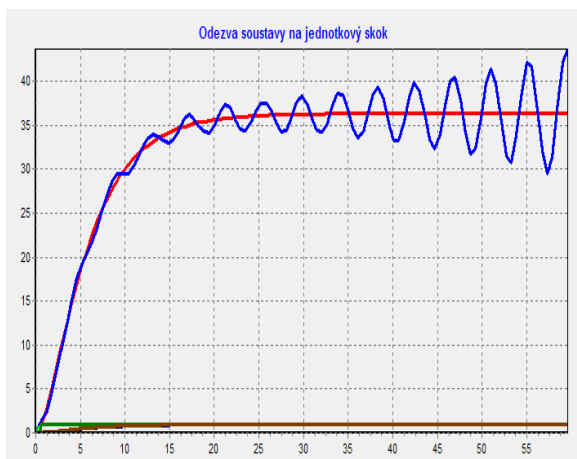
Soustava s regulací (P,D) s 8 bitovými převodníky – počítáno metodou Runge-Kutta 4. řádu:



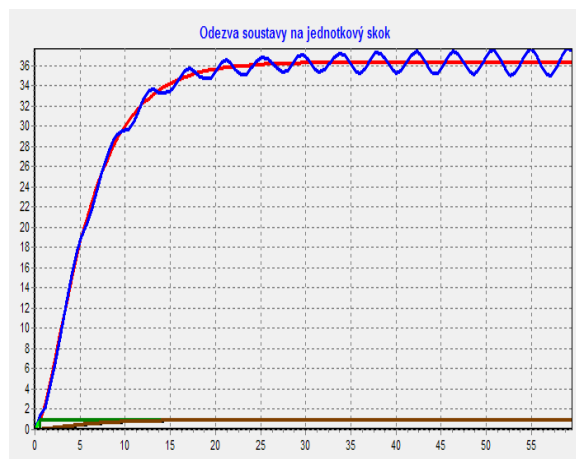
Obrázek 8.81 – $\Delta t = 0,578s$



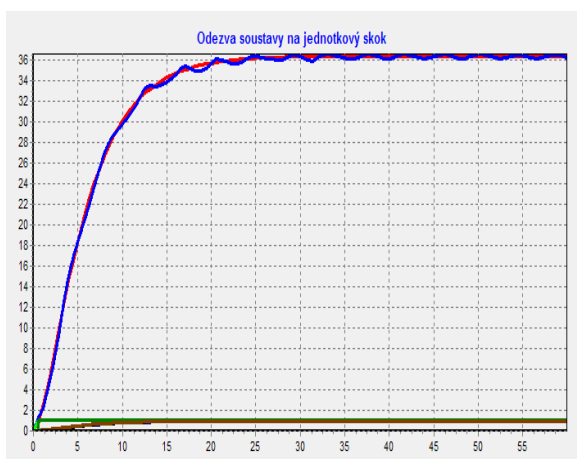
Obrázek 8.82 – $\Delta t = 0,575s$



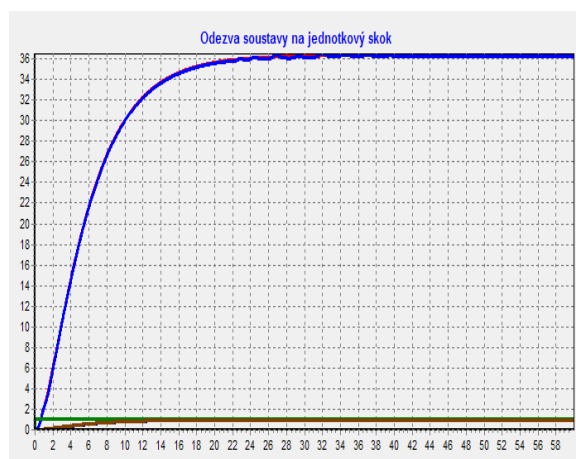
Obrázek 8.83 – $\Delta t = 0,573s$



Obrázek 8.84 – $\Delta t = 0,572s$

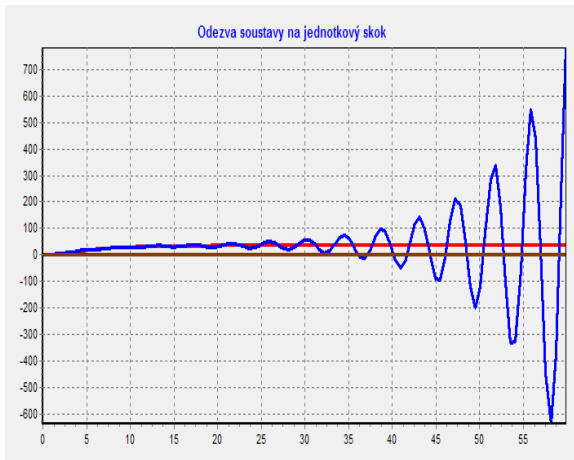


Obrázek 8.85 – $\Delta t = 0,571s$

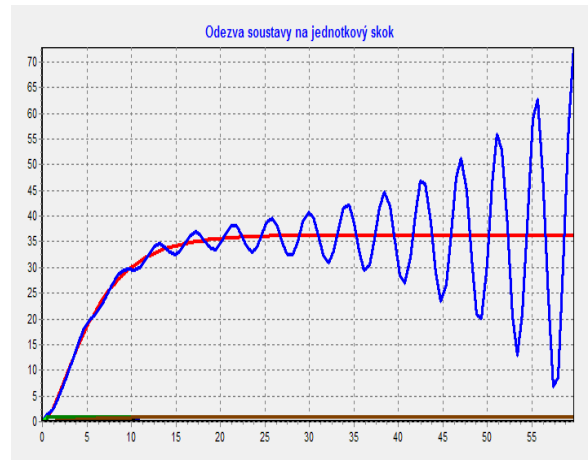


Obrázek 8.86 – $\Delta t = 0,001s$

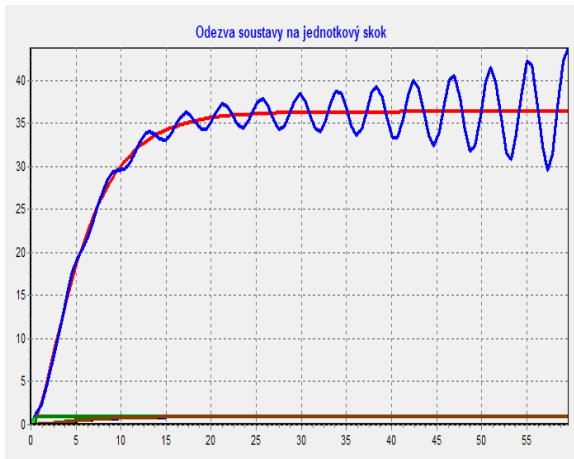
Soustava s regulací (P,D) s 12 bitovými převodníky – počítáno metodou Runge-Kutta 4. řádu:



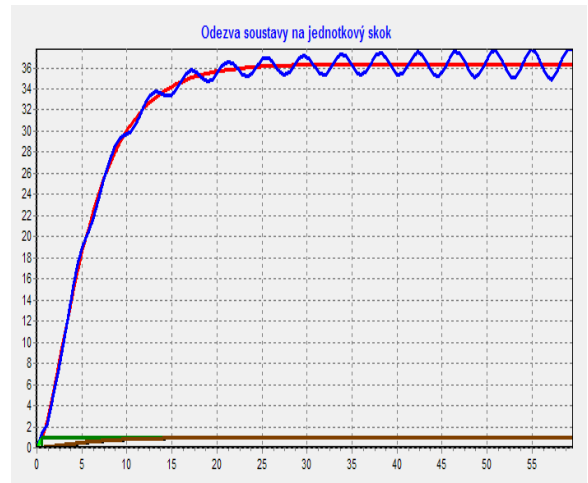
Obrázek 8.87 – $\Delta t = 0,576s$



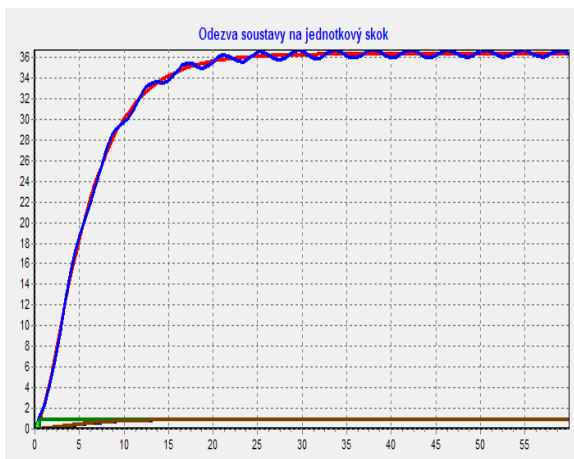
Obrázek 8.88 – $\Delta t = 0,574s$



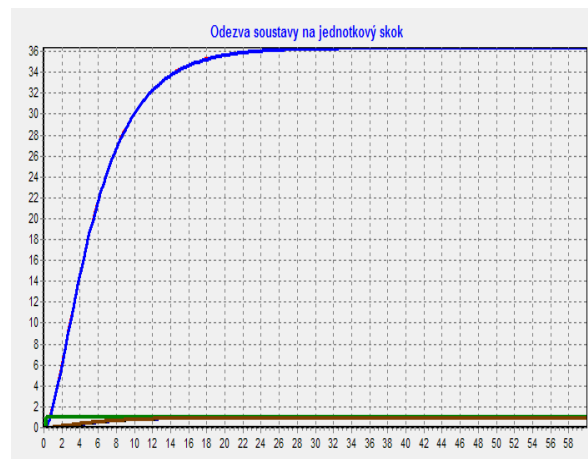
Obrázek 8.89 – $\Delta t = 0,573s$



Obrázek 8.90 – $\Delta t = 0,572s$



Obrázek 8.91 – $\Delta t = 0,571s$



Obrázek 8.92 – $\Delta t = 0,300s$

9. Závěr

V této diplomové práci se řešila teorie automatického řízení a regulace, modelování dynamických systémů a problematika modelování v reálném čase.

Jelikož je většina modelů popsána soustavou diferenciálních rovnic bylo nutné také nastudovat numerické integrační metody, zejména pak moderní variantu založenou na principu Taylorovy řady.

Další problematika probíraná v této práci se týkala regulačních obvodů, jejich reprezentace blokovými schémata a soustavami diferenciálních rovnic.

Pro výpočty a experimenty se používaly systémy TKSL a TKSL/C a vlastní program v C++ Builderu. Tyto experimenty měly za cíl ověřit vlastnosti základních regulačních prvků. Z těchto základních prvků pak byl sestaven regulační obvod s modelem regulované soustavy.

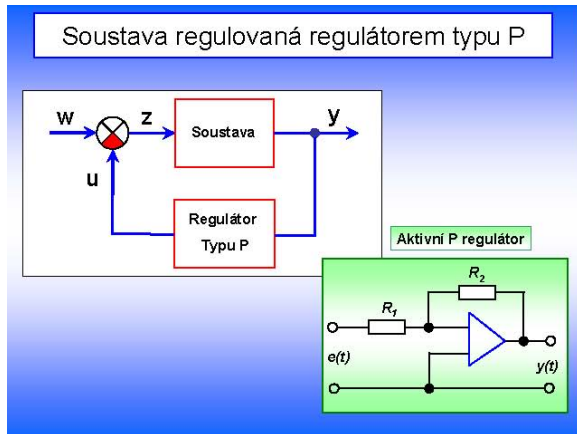
V praktické části práce jsem implementoval program pro simulaci regulované soustavy. Zjišťoval jsem vliv integračního kroku, přesnosti výpočtu, zařazení převodníků s různou rozlišovací schopností na regulační schopnosti dané soustavy.

10. Literatura

- [1] Blaha, P., Vavřín, P.: Řízení a regulace 1, skripta, VUT Brno, 1990
- [2] Murina, M.: Teorie obvodů, skripta, VUTIUM Brno, 2000.
- [3] Vavřín, P.: Teorie automatického řízení 1, skripta, VUT Brno, 1981
- [4] Vavřín, P.: Teorie řízení 1 – 1. část, skripta, VUT Brno, 1980
- [5] Vavřín, P.: Teorie řízení 1 – 2. část, skripta, VUT Brno, 1980
- [6] Spíral, L., Ovsjannikov, V.: Optimalizace průmyslových regulačních obvodů, SNTL, Praha, 1982
- [7] Kotek, Štecha: Teorie automatického řízení spojitého lineárního systému, Skriptum ČVUT, Praha, 1977
- [8] Kubík, Kotek, Šalomon: Teorie regulace 1 - lineární systémy, SNTL, Praha, 1974
- [9] Frankovič, Petráš, Skákala, Vykouk: Automatizácia a samočinné riadenie, Slov. NTL, Bratislava, 1966
- [10] Švec a kol.: Příručka automatizační a výpočetní techniky, SNTL, Praha, 1975
- [11] Markl, J.: Metody identifikace a online optimalizace procesů, Sborník ZP SVTS při USIP, Žilina, 1974

11. Přílohy – slide v PowerPointu

11.1 Regulace regulátory P, I, D



Rovnice pro regulátor typu P

1. blok $\frac{y}{z} = \frac{1}{p^2 + 2p + 1}$ $p^2 y + 2py + y = z$

2. blok $\frac{u}{y} = a$ $u = a \cdot y$

SOUSTAVA DIFERENCIÁLNÍCH ROVNIC

$y'' + 2y' + y = z$ $u = y \cdot a$

$z = w - u$

Programy

Program v TKSL

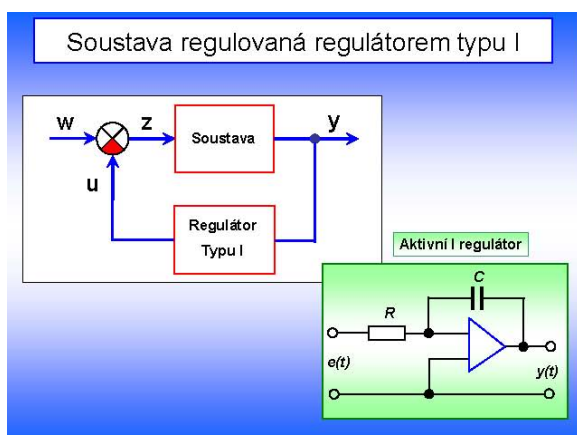
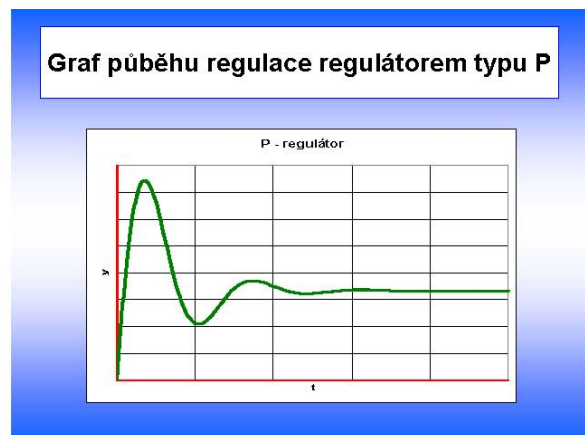
```
var x, y, z, u, w;
const a=5, dt=0.1, eps=1e-20,
t0=0, tmax=10;

system
y'=x &0;
x'=(-2)*x-y+z &1;
u=y*a;
z=w-u;
w=1;

sysend.
```

Program v TKSL/C

```
y'=x;
y(0)=0;
x'=(-2)*x-y+z;
x(0)=1;
u=y*5;
z=w-u;
w=1;
```



Rovnice pro regulátor typu I

1. blok $\frac{y}{z} = \frac{1}{p^2 + 2p + 1}$ $p^2 y + 2py + y = z$

2. blok $\frac{u}{y} = \frac{a}{p}$ $pu = a \cdot y$

SOUSTAVA DIFERENCIÁLNÍCH ROVNIC

$y'' + 2y' + y = z$ $u' = y \cdot a$

$z = w - u$

Programy

Program v TKSL

```

var x,y,z,u,w;
const a=1.5,dt=0.1,eps=1e-20,
t0=0,tmax=150;

system
y'=x &0;
x'=(-2)*x-y+z &1;
u'=y*a &0;
z=w-u;
w=1;

sysend.

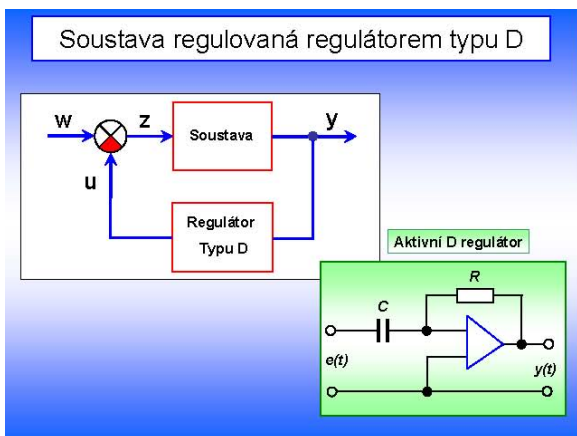
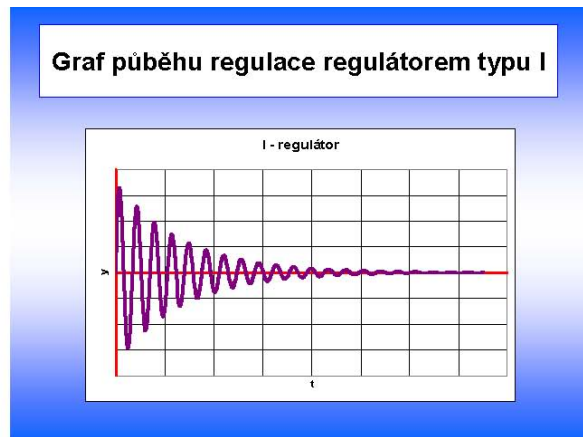
```

Program v TKSL/C

```

y'=x;
y(0)=0;
x'=(-2)*x-y+z;
x(0)=1;
u'=y*1.5;
u(0)=0;
z=w-u;
w=1;

```



Rovnice pro regulátor typu D

1. blok $\frac{y}{z} = \frac{1}{p^2 + 2p + 1}$ $p^2 y + 2py + y = z$

2. blok $\frac{u}{y} = a.p$ $u = a.p.y$

SOUSTAVA DIFERENCIÁLNÍCH ROVNIC

$y'' + 2y' + y = z$

$a.y' = u$

$z = w - u$

Programy

Program v TKSL

```

var x,y,z,u,w;
const a=10,dt=0.1,eps=1e-20,
t0=0,tmax=200;

system
y'=x &0;
x'=(-2)*x-y+z &1;
u=a*x;
z=w-u;
w=1;

sysend.

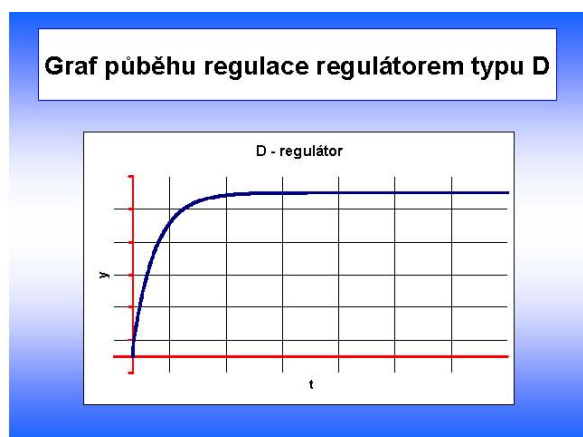
```

Program v TKSL/C

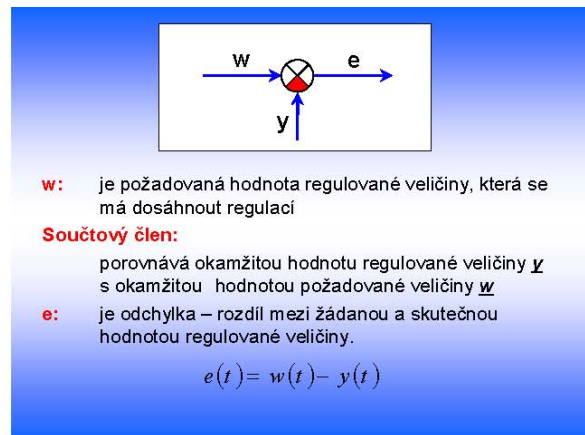
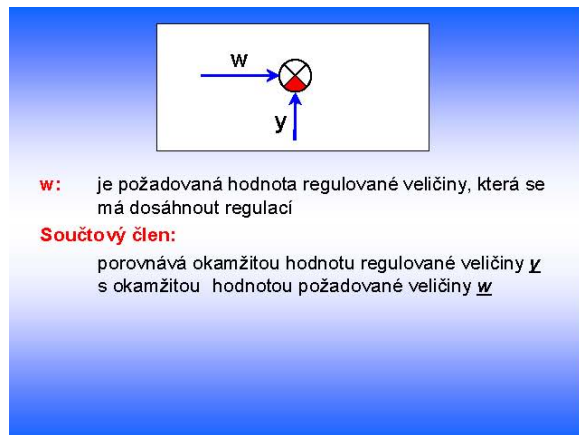
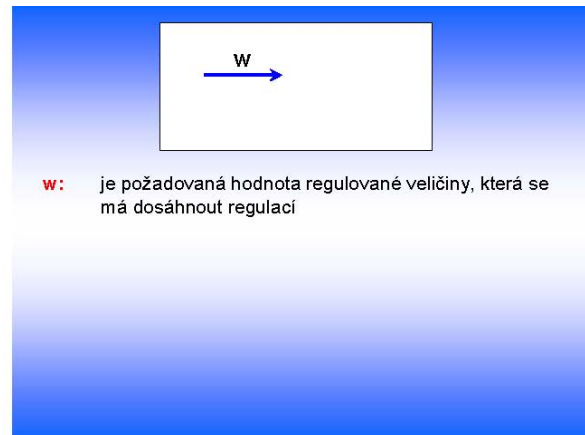
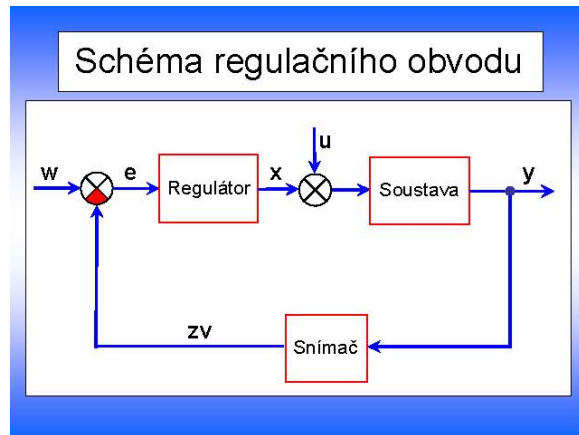
```

y'=x;
y(0)=0;
x'=(-2)*x-y+z;
x(0)=1;
u=10*x;
z=w-u;
w=1;

```

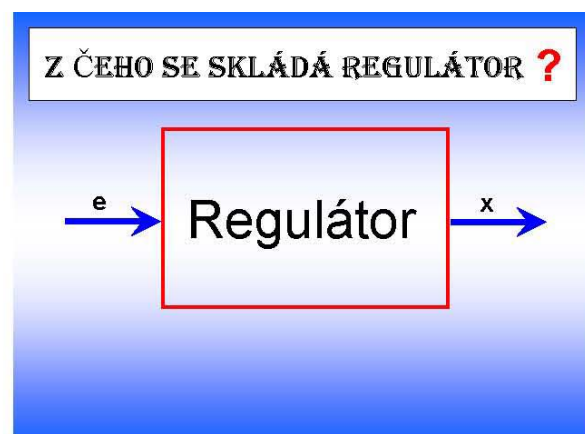


11.2 Regulační obvody

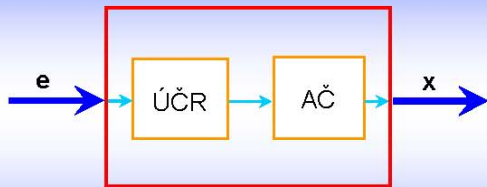


CO JE TO REGULÁTOR ?

- Je to zařízení, které v obvodech automatické regulace na základě regulační odchylky e vytváří vhodný signál
- Pomocí svého akčního členu působí na regulovanou soustavu



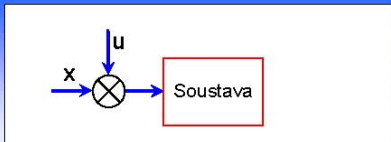
Z ČEHO SE SKLÁDÁ REGULÁTOR ?



ÚČR: Ústřední Člen Regulátoru
AČ: Akční Člen

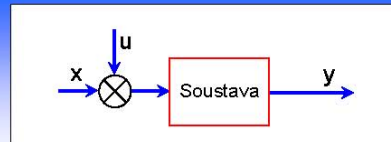


u: je porucha – rušivý signál, způsobující nežádoucí změny regulované veličiny.



u: je porucha – rušivý signál, způsobující nežádoucí změny regulované veličiny.

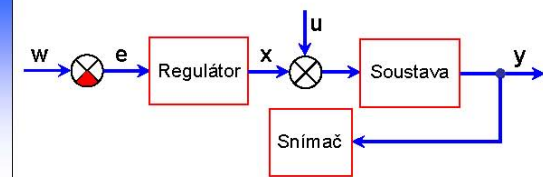
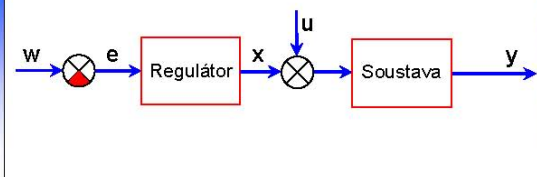
Soustava: je objekt na který prostřednictvím akční veličiny x působí regulátor (pomocí svého akčního členu) tak, aby určité fyzikální veličiny regulované soustavy měly požadované hodnoty.



u: je porucha – rušivý signál, způsobující nežádoucí změny regulované veličiny.

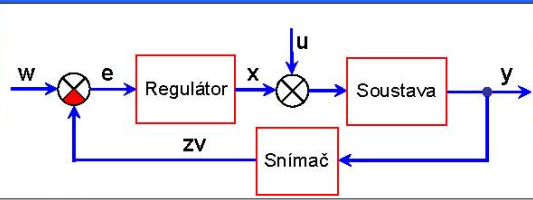
Soustava: je objekt na který prostřednictvím akční veličiny x působí regulátor (pomocí svého akčního členu) tak, aby určité fyzikální veličiny regulované soustavy měly požadované hodnoty.

y: je výstupní veličina regulované soustavy.



Snímač:

slouží ke snímání skutečné hodnoty regulované veličiny y .



Snímač:

slouží ke snímání skutečné hodnoty regulované veličiny y .

zv: je zpětná vazba – přenáší část signálu z výstupu zařízení na jeho vstup. Může být kladná (signály se sčítají) nebo záporná (signály se odečítají)

PRINCIP REGULACE

- Obvod obsahuje zpětnou vazbu
- Řídící člen dostává informaci o stavu řízeného objektu (regulované veličiny) a v případě nutnosti provádí akční zásah

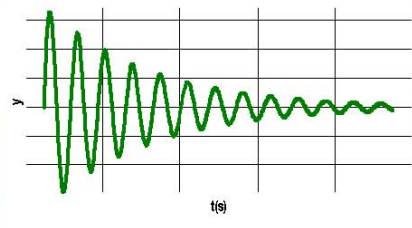
Automatická regulace – je samočinné udržování určité fyzikální veličiny na požadované hodnotě

DRUHÝ REGULACE

1. Regulace na konstantní hodnotu – v čase regulace se nemění nastavení požadované hodnoty.
2. Regulace programová – velikost požadované hodnoty se mění s časem podle předem známého programu.
3. Regulace vlečná – požadovaná hodnota regulované veličiny se mění v závislosti na nějaké jiné veličině proměnné v čase jejíž časový průběh však není znám

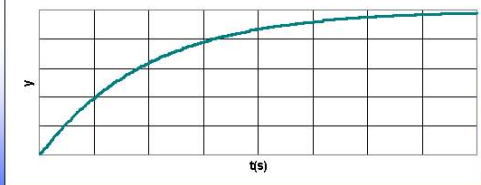
Příklad neregulované soustavy 2. řádu – 1/2

Neregulovaná kmitavá soustava



Příklad neregulované soustavy 2. řádu – 2/2

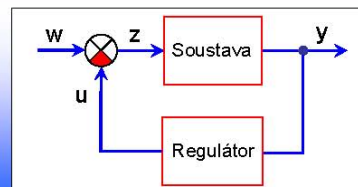
Neregulovaná velmi tlumená soustava



Soustava s regulátorem

Neregulovanou soustavu chceme regulovat takto:

- Do záporné zpětné vazby zařadíme regulátor.
- Regulátor se snaží udržet výstupní veličinu y na požadované hodnotě w .

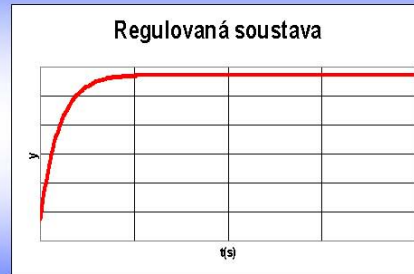


Program popisující regulační obvod

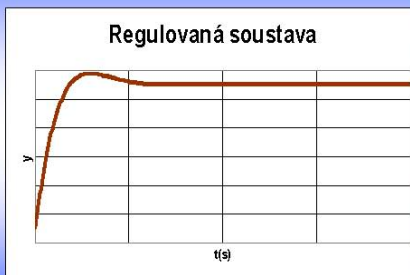
Program v TKSL

```
var x, y, z, u, w;  
  
const a=0, dt=0.1,  
      eps=1e-20, tmax=20;  
  
system  
y'=u &0;  
u'=- (2)*x-y+z &1;  
u'=y*a &0;  
z=w-u;  
w=1;  
  
sysend.
```

Výsledná regulovaná soustava 2. řádu – 1/2



Výsledná regulovaná soustava 2. řádu – 2/2



Základní typy regulátorů

Máme 3 základní typy regulátorů. P, I, D.

- **Proporcionální regulátory (P):**
 - pouze zesilují (zeslabují pokud jsou pasivní).
 - jejich přenosová funkce bývá nultého až druhého řádu.

Základní typy regulátorů

Máme 3 základní typy regulátorů. P, I, D.

- **Proporcionální regulátory (P):**
 - pouze zesilují (zeslabují pokud jsou pasivní).
 - jejich přenosová funkce bývá nultého až druhého řádu.
- **Integrační regulátory (I):**
 - jejich výstupní veličin a je integrálem regulované odchylky.
 - jejich přenosová funkce bývá prvního nebo druhého řádu

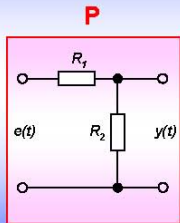
Základní typy regulátorů

Máme 3 základní typy regulátorů. P, I, D.

- **Proporcionální regulátory (P):**
 - pouze zesilují (zeslabují pokud jsou pasivní).
 - jejich přenosová funkce bývá nultého až druhého řádu.
- **Integrační regulátory (I):**
 - jejich výstupní veličin a je integrálem regulované odchylky.
 - jejich přenosová funkce bývá prvního nebo druhého řádu
- **Derivační regulátory (D):**
 - jejich výstupní signál je derivací vstupního signálu.
 - změna akční veličiny je úměrná rychlosti změny odchylky.
 - může být pouze součástí regulátoru PD a nebo PID.

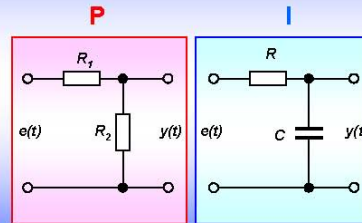
Příklady pasivních regulátorů

P



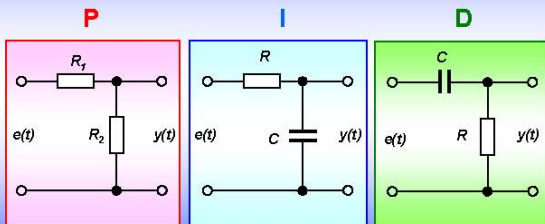
Příklady pasivních regulátorů

P, I



Příklady pasivních regulátorů

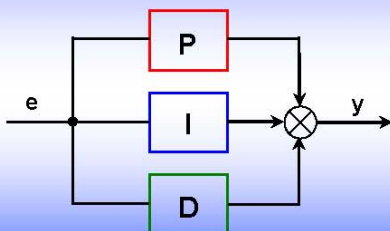
P, I, D



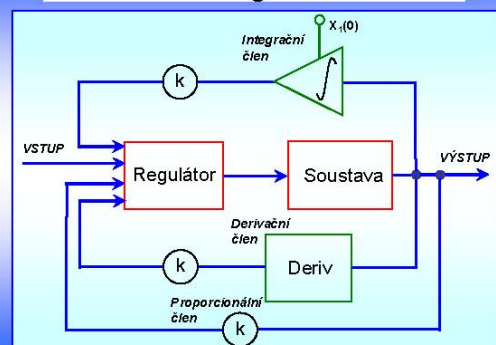
Sdružené regulátory

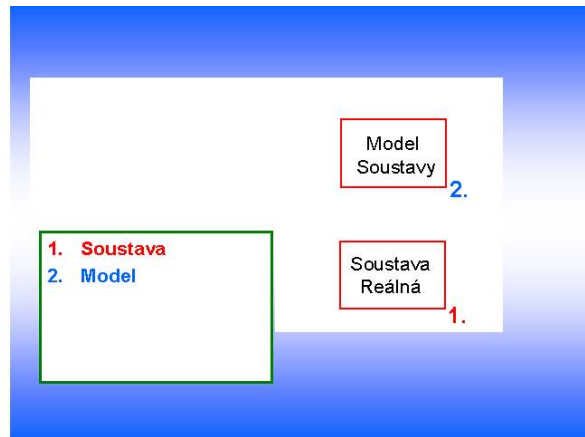
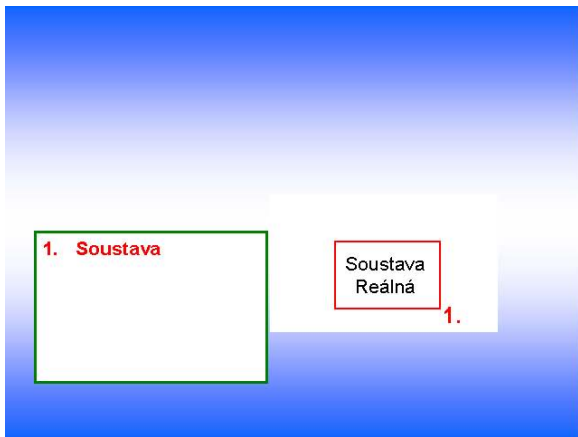
- **PD** – vznikne paralelním spojením regulátorů P a D
- **PI** – vznikne paralelním spojením regulátorů P a I
- **PID** – vznikne paralelním spojením regulátorů P, I a D.

Regulátor PID

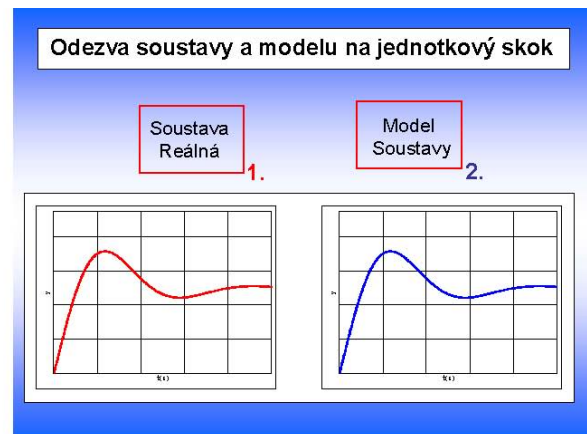
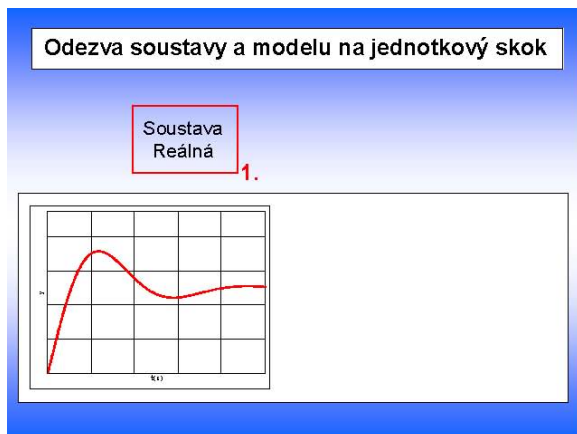
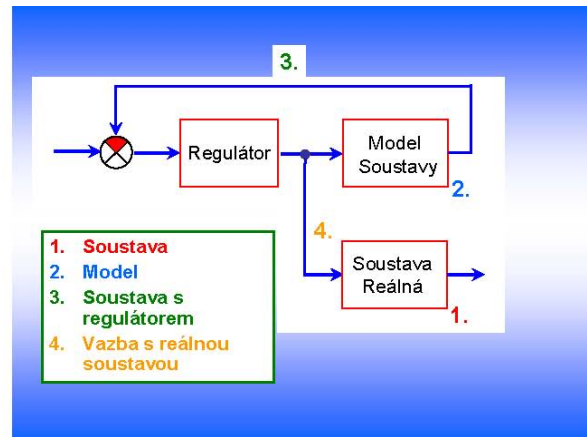
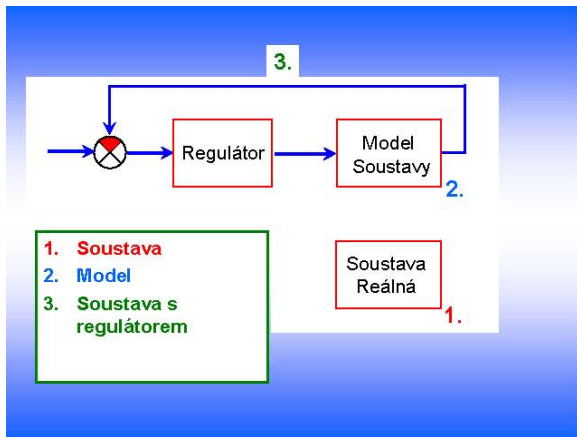


Soustava s regulátorem PID



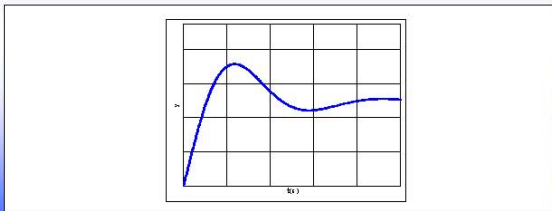


60



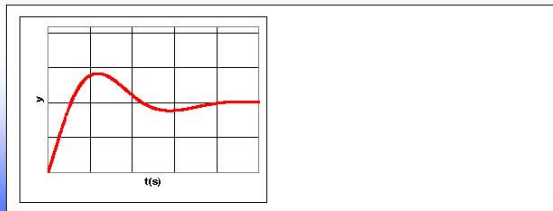
Odezva soustavy a modelu na jednotkový skok

Soustava Reálná 1. = Model Soustavy 2.



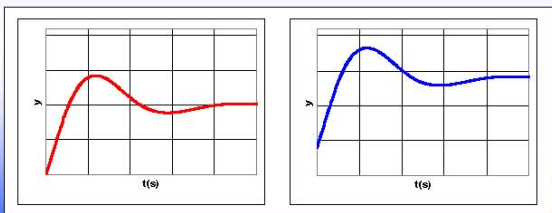
Odezva soustavy a modelu na jednotkový skok

Soustava Reálná 1.



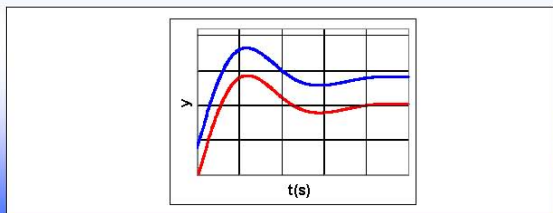
Odezva soustavy a modelu na jednotkový skok

Soustava Reálná 1. Model Soustavy 2.

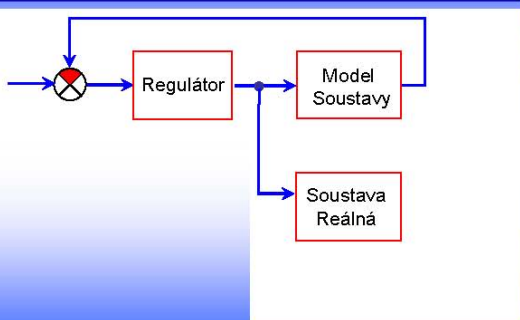


Odezva soustavy a modelu na jednotkový skok

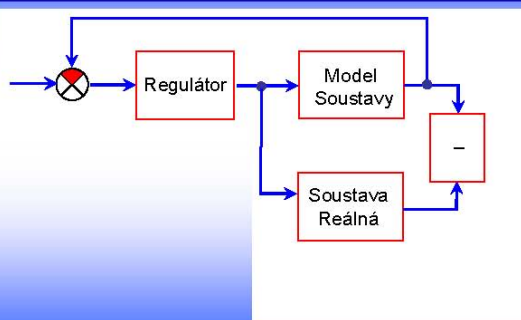
Soustava Reálná 1. ≠ Model Soustavy 2.

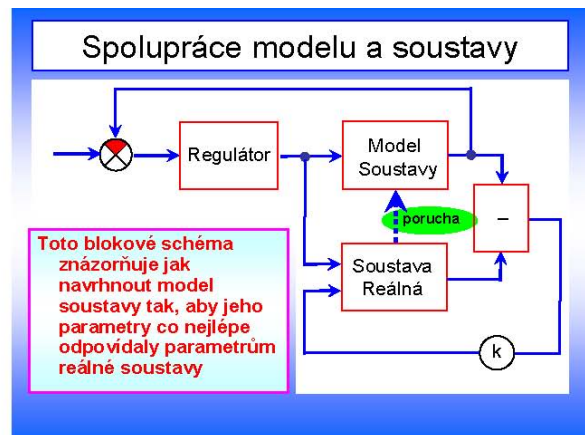
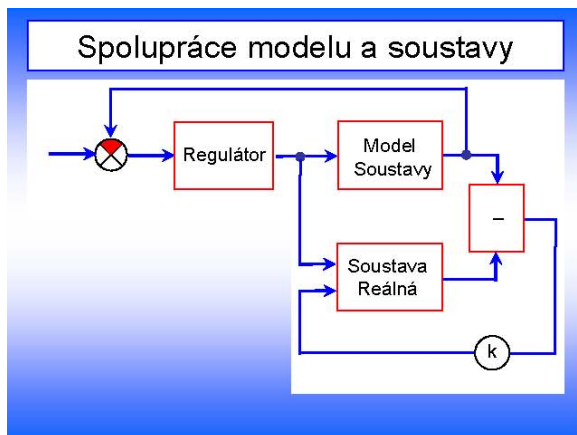


Spolupráce modelu a soustavy



Spolupráce modelu a soustavy





Toto blokové schéma znázorňuje jak navrhnout model soustavy tak, aby jeho parametry co nejlépe odpovídaly parametrům reálné soustavy

11.3 Analýza regulačního obvodu

ANALÝZA REGULAČNÍHO OBVODU

řešit regulační obvod znamená vytvořit regulační bloky a navzájem je mezi sebou propojit.

k popisu regulačních obvodů se používá přenosových funkcí

- přenosové funkce lze řešit pomocí diferenciálních rovnic

Příklad regulačního obvodu:

BLOKOVÉ SCHÉMA

The diagram shows a control loop with a summing junction. The input is 'w'. The output of the summing junction is 'z'. This signal goes through a block with transfer function $\frac{L}{p^2 + 2p + 1}$. The output is 'y'. This signal goes through a block with transfer function $\frac{Ls}{p}$. The output is 'u'. This signal goes back to the summing junction.

OPERÁTOROVÝ PŘENOS JEDNOTLIVÝCH BLOKŮ

1. blok $\frac{y}{z} = \frac{1}{p^2 + 2p + 1}$ $p^2 y + 2p y + y = z$

2. blok $\frac{x}{y} = \frac{a}{p}$ $p x = y a$

SOUSTAVA DIFERENCIÁLNÍCH ROVNIC

$y'' + 2y' + y = z$ $x' = y \cdot a$

$\Sigma = W - X$

Program v TKSL

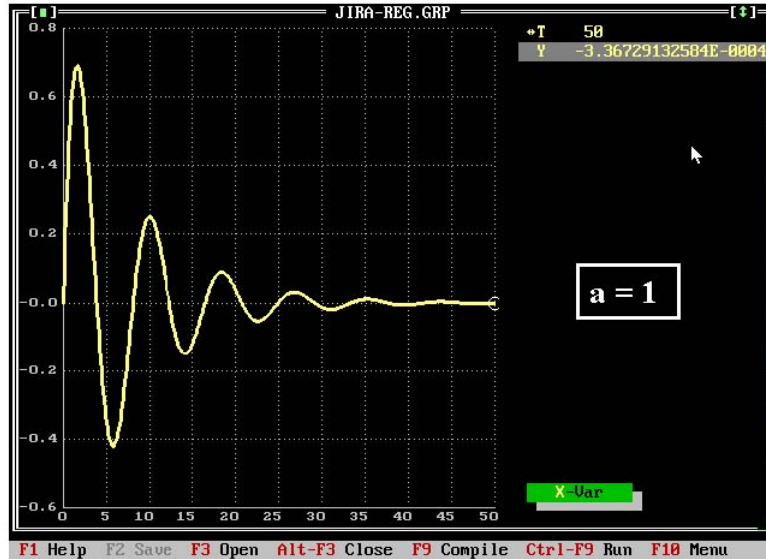
Taylor - Kinovsky Simulation Language

```

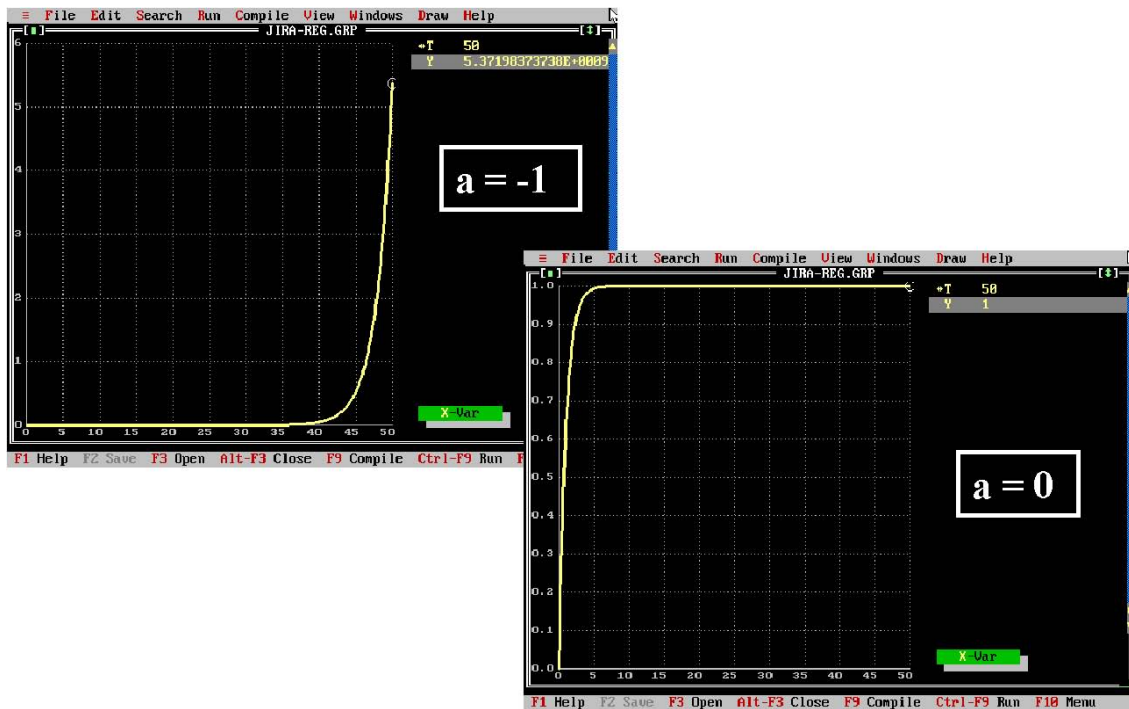
var x,y,z,u,v,
const a=1,dt=0.1,eps=1e-20,
system
y'=z-0,
x'=-12*x-y+e dt,
u'=y*a-0,
z=w-u,
w=1,
v=1,
sysend.

```


Simulace v TKSL 386 1/3



Simulace v TKSL 386 2/3



Simulace v TKSL 386 2/3

