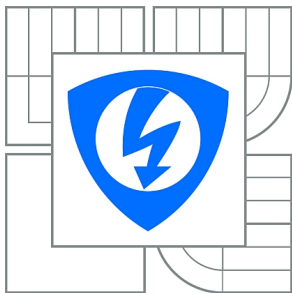


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF CONTROL AND INSTRUMENTATION

## LABORATORNÍ ÚLOHY PRO PŘÍPRAVEK S KULIČKOU NA NAKLONĚNÉ ROVINĚ

LABORATORY TASKS FOR BALL ON AN INCLINED PLANE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MARTIN ŘEZÁČ

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. PETR BLAHA, Ph.D.

BRNO 2013



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav automatizace a měřicí techniky

# Bakalářská práce

bakalářský studijní obor  
Automatizační a měřicí technika

**Student:** Martin Řezáč

**ID:** 134600

**Ročník:** 3

**Akademický rok:** 2012/2013

## NÁZEV TÉMATU:

**Laboratorní úlohy pro přípravku s kuličkou na nakloněné rovině**

## POKYNY PRO VYPRACOVÁNÍ:

- seznámte se s existujícím přípravkem kuličky na nakloněné rovině, s prostředím LabView a s jeho použitím pro řízení systémem CompactRIO,
- proveďte zprovoznění přípravku, navrhnete a realizujete vhodnou náhradu stávajícího principu snímání polohy kuličky,
- vytvořte model přípravku v prostředí Matlab Simulink (případně v LabView Control Design and Simulation),
- navrhnete několik typů regulátorů polohy kuličky na nakloněné rovině, které odzkoušíte nejprve v simulaci a následně na reálném přípravku pomocí systému CompactRIO,
- navrhnete laboratorní úlohu pro návrh regulátorů polohy kuličky a pro identifikaci parametrů modelu kuličky na nakloněné rovině. Zadání, teoretický úvod a vzorové vypracování obou úloh uveďte v příloze bakalářské práce.

## DOPORUČENÁ LITERATURA:

- [1] Soederstrom, T., Stoica, P.: System identification. Prentice Hall International, 1989.
- [2] Židek J.: Grafické programování ve vývojovém prostředí Labview, VŠB-TU Ostrava, říjen 2002.
- [3] Blaha, P., Vavřín, P.: Řízení a regulace I. Elektronické skriptum VUT.

další dle doporučení vedoucího

**Termín zadání:** 11.2.2013

**Termín odevzdání:** 27.5.2013

**Vedoucí práce:** doc. Ing. Petr Blaha, Ph.D.

**Konzultanti bakalářské práce:**

**doc. Ing. Václav Jirsík, CSc.**

*Předseda oborové rady*

## **ABSTRAKT**

Práce se zabývá problematikou identifikace a řízení dynamických systémů s využitím grafického programovacího prostředí LabVIEW. Pro řízení je využita kaskádní formu regulace s PI a PD regulátory navrženými pomocí programu Matlab. Tyto regulátory byly následně odzkoušeny na reálném přípravku. Práce obsahuje také zadání laboratorní úlohy a vzorové vypracování.

## **KLÍČOVÁ SLOVA**

CompactRIO, LabVIEW, nakloněná rovina, identifikace systému, kaskádní regulace, frekvenční charakteristiky

## **ABSTRACT**

The thesis deals with the identification and control of dynamic systems using LabVIEW graphical programming environment. Cascade form of control with PI and PD controllers has been used for regulation and designed in Matlab. These controllers were then tested on real product. Laboratory exercises with model solutions are also included.

## **KEYWORDS**

CompactRIO, LabVIEW, inclined plane, identification of system, cascade control, frequency characteristics

ŘEZÁČ, Martin *Laboratorní úlohy pro přípravek s kuličkou na nakloněné rovině*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2013. 60 s. Vedoucí práce byl Doc. Ing. Petr Blaha, Ph.D.

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Laboratorní úlohy pro přípravu s kuličkou na nakloněné rovině“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Rád bych poděkoval Ing. Ivu Veselému a Ing. Liboru Veselému za jejich ochotu a trpělivost při konzultacích, dále pak vedoucímu bakalářské práce panu doc. Ing. Petru Blahovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno .....

.....

(podpis autora)

# OBSAH

Úvod	11
<b>1 National Instrument CompactRIO</b>	<b>12</b>
1.1 CompactRIO	12
1.2 FPGA	13
1.3 Real-Time system	14
<b>2 Použité moduly C Series</b>	<b>15</b>
2.1 C series modules	15
2.2 NI 9401	15
2.3 NI 9203	16
2.4 NI 9215	16
<b>3 Napájení</b>	<b>17</b>
3.1 Diametral P230R51D	17
<b>4 Hardwarové příslušenství modelu</b>	<b>18</b>
4.1 Původní model	18
4.2 Odporový snímač náklonu	18
4.3 Servo HS-322	18
4.4 ILR 1030-8	19
4.5 Popis modelu	20
<b>5 Filtrace signálu</b>	<b>21</b>
5.1 Filtrace proudové smyčky	21
5.2 RC článek - dolní propust	22
<b>6 Regulace</b>	<b>24</b>
6.1 Kaskádní regulace	24
6.2 Regulátor	25
6.2.1 PI regulátor	25
6.2.2 PD regulátor	26
6.3 Použitý regulátor	27
<b>7 Identifikace systému</b>	<b>28</b>
7.1 Identifikace vnitřní smyčky	28
7.1.1 Získání dat pro identifikaci	28
7.1.2 Identifikace z přechodových charakteristik	29

7.1.3	System identification toolbox . . . . .	29
7.2	Získání dat pro identifikaci vnější smyčky . . . . .	30
7.3	Návrh regulátoru metodou frekvenčních charakteristik . . . . .	32
7.3.1	Posouzení stability . . . . .	32
7.3.2	Návrh regulátoru vnitřní smyčky . . . . .	32
7.4	Návrh regulátoru polohy kuličky . . . . .	34
<b>8</b>	<b>Model přípravku</b>	<b>36</b>
8.1	Tvorba modelu . . . . .	36
<b>9</b>	<b>Software</b>	<b>37</b>
9.1	LabVIEW . . . . .	37
9.2	Konfigurace CompactRIO . . . . .	37
9.3	Programování FPGA modulu . . . . .	38
9.3.1	Vytvoření pulzní šířkové modulace . . . . .	38
9.3.2	Zpracování signálů ze senzorů . . . . .	39
9.4	Komunikace FPGA a Real-Time . . . . .	40
9.5	Programování Real-Time modulu . . . . .	40
9.5.1	Zpracování dat ze senzorů . . . . .	40
9.5.2	Ruční ovládání náklonu . . . . .	42
<b>10</b>	<b>Studentská aplikace Kulicka</b>	<b>43</b>
10.1	Uživatelské prostředí aplikace . . . . .	43
10.2	Programová část . . . . .	43
<b>11</b>	<b>Závěr</b>	<b>45</b>
	<b>Literatura</b>	<b>47</b>
	<b>Seznam symbolů, veličin a zkratk</b>	<b>48</b>
	<b>Seznam příloh</b>	<b>49</b>
<b>A</b>	<b>Laboratorní úloha</b>	
	<b>Návrh řízení pro přípravek kulička na nakloněné rovině</b>	<b>50</b>
A.1	Zadání . . . . .	50
A.2	Teoretický úvod . . . . .	50
A.2.1	Přípravek kulička na nakloněné rovině . . . . .	50
A.2.2	Regulátor . . . . .	51
A.2.3	Kaskádní regulace . . . . .	51
A.2.4	System identification toolbox . . . . .	52

A.2.5	Návrh regulátoru pomocí frekvenčních charakteristik . . . . .	53
A.2.6	Aplikace Kulicka . . . . .	54
<b>B</b>	<b>Vzorové vypracování</b>	<b>55</b>
B.1	Zadání . . . . .	55
B.2	Vypracování . . . . .	56
B.3	Závěr . . . . .	59
B.4	Obsah příloženého CD . . . . .	60



# SEZNAM OBRÁZKŮ

1.1	NI Compact RIO . . . . .	12
1.2	Detail obvodu FPGA, převzato z [2] . . . . .	13
2.1	Ukázka použitých C-series modulů . . . . .	15
3.1	Napájecí zdroj Diametral P230R51D . . . . .	17
4.1	Schématické zobrazení původního modelu . . . . .	18
4.2	Závislost polohy na délce pulzu . . . . .	19
4.3	Schématické zobrazení aktuálního modelu . . . . .	20
5.1	Nefiltrovaný průběh ze snímače vzdálenosti . . . . .	21
5.2	Použitý Butterworthův fitr v LabVIEW . . . . .	22
5.3	Filtrovaný průběh ze snímače vzdálenosti . . . . .	22
5.4	Amplitudová frekvenční charakteristika . . . . .	23
6.1	Schéma regulačního obvodu . . . . .	24
6.2	Uspořádání smyček při kaskádní regulaci . . . . .	25
6.3	Frekvenční charakteristika PI regulátoru (převzato z [13]) . . . . .	26
6.4	Frekvenční charakteristiky ideálního a reálného PD regulátoru (převzato z [13]) . . . . .	27
6.5	Použitý PID regulátor . . . . .	27
7.1	Získávání dat k identifikaci vnitřní smyčky . . . . .	28
7.2	Časové průběhy odezev na jednotkový skok . . . . .	29
7.3	Návrh soustavy pomocí Ident . . . . .	30
7.4	Získávání dat pro identifikaci vnější smyčky . . . . .	30
7.5	Frekvenční charakteristika identifikovaného modelu soustavy . . . . .	33
7.6	Frekvenční charakteristika otevřeného obvodu vnitřní smyčky . . . . .	34
7.7	Frekvenční charakteristika identifikovaného modelu soustavy rozšířené o dva integrátory . . . . .	34
7.8	Frekvenční charakteristika otevřeného obvodu vnější smyčky . . . . .	35
8.1	Model přípravku vytvořený v Matlab Simulink . . . . .	36
9.1	Konfigurace . . . . .	38
9.2	Vytvoření PWM (blokové schéma) . . . . .	39
9.3	Zpracování dat pomocí FPGA . . . . .	39
9.4	Komunikace FPGA a Real-Time . . . . .	40
9.5	Ikona SubVI . . . . .	40
9.6	Zpracování dat v podprogramu vypocty.vi . . . . .	41
9.7	Čelní panel programu . . . . .	42
10.1	Uživatelské prostředí aplikace Kulicka . . . . .	43
10.2	Aplikace Kulicka programová část . . . . .	44
A.1	Schéma přípravku kulička na nakloněné rovině . . . . .	51

A.2	Schéma regulačního obvodu . . . . .	51
A.3	Uspořádání smyček při kaskádní regulaci . . . . .	52
A.4	Ukázka použití System identification toolboxu . . . . .	52
A.5	Uživatelské prostředí aplikace Kulicka . . . . .	54
B.1	Úprava signálů pro identifikaci . . . . .	56
B.2	Shoda naměřené a identifikované odezvy . . . . .	57
B.3	Frekvenční charakteristika otevřeného obvodu vnitřní smyčky . . . . .	57
B.4	Frekvenční charakteristika otevřeného obvodu vnější smyčky . . . . .	58
B.5	Odezva na skok požadované veličiny . . . . .	59

# ÚVOD

Cílem této práce je provést identifikaci a řízení dynamického systému kulička na nakloněné rovině pomocí programu LabVIEW a systému CompactRIO, a vytvořit na tyto témata laboratorní úlohy.

V první části práce jsme se seznámili se stávajícím přípravkem, provedli jeho osazení laserovým snímačem vzdálenosti. Dále jsme se seznámili s prostředím LabVIEW a jeho použitím pro řízení systému CompactRIO včetně jeho modulů. Po následném zprovoznění servopohonu a snímání všech potřebných veličin jsme započali s prací na řízení a identifikaci systému.

Při návrhu regulátoru jsme zjistili, že regulace pomocí odchylkového regulátoru je v tomto případě nedostatečná, jelikož dochází k ovlivňování náklonu roviny pozicí kuličky. Z tohoto důvodu jsme se rozhodli pro použití kaskádní regulaci. Na identifikovanou soustavu vnitřní smyčky (metodou nejmenších čtverců) jsme metodou frekvenčních charakteristik navrhli PI regulátor a následně za pomoci modelu systému vnější PD regulátor.

V poslední části práce jsme vytvořili laboratorní úlohu včetně teoretického úvodu a vzorového vypracování na téma Návrh řízení pro přípravek kulička na nakloněné rovině.

# 1 NATIONAL INSTRUMENT COMPACTRIO

## 1.1 CompactRIO

CompactRIO [1] je proměnný vestavěný řídicí systém vyznačující se malou spotřebou a kompaktními rozměry. Je vyráběn v několika různých verzích, které se liší jednak počtem slotů pro vstupně/výstupní karty, ale také typem procesoru (a to ať už jednodurový, či dvoudurový Intel Core i7). Šasi samotné je nejdůležitější částí celého systému CompactRIO, jelikož obsahuje dvě programovatelná jádra, první je určeno pro FPGA a druhé pro procesor s operačním systémem reálného času (a to ať už WES7 nebo LabVIEW Real-Time).

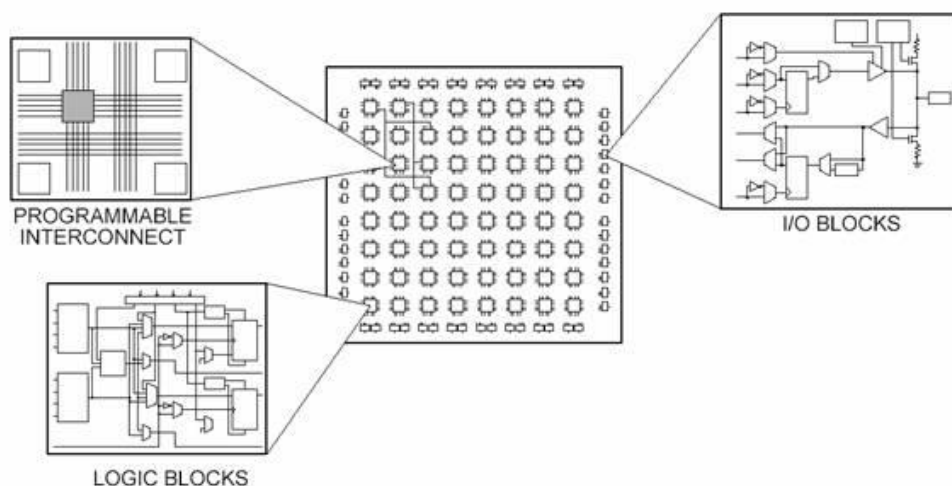
V našem případě používáme NI CompactRIO-9076, které vidíme na obrázku 1.1. Jedná se o model s jednodurovým procesorem o taktu 400 MHz a paměti DRAM 256 MB. Čip FPGA je vyroben firmou Xilinx. Do šasi je možno umístit čtyři měřicí karty, avšak v našem případě postačují tři. Na šasi je vyveden konektor RJ45 sloužící pro propojení PC a cRIO-9076, dále pak sériový port RS232 a port USB pro připojení periferních zařízení či uložení dat. Napájení je realizováno pomocí stejnosměrného napětí v rozmezí 9 až 30 V přivedeného na svorky označené V a C. Dále jsou na šasi čtyři informační led diody. Ty nás informují o napájení, připojení do sítě, práci s CRIO či práci s FPGA.



Obr. 1.1: NI Compact RIO

## 1.2 FPGA

Psáno podle [2]. První obvod tohoto typu byl vynalezen v roce 1985. V překladu se jedná o programovatelné hradlové pole, nebo přesněji přeprogramovatelné křemíkové čipy propojená maticí spojů. Přeprogramovatelný křemík má stejné vlastnosti jako mikroprocesory, ale na rozdíl od nich není limitován počtem jader, které jsou k dispozici. Současné obvody tohoto typu obsahují až 6 miliónů hradel. S velkou oblibou jsou FPGA kombinována s klasickými mikroprocesory.



Obr. 1.2: Detail obvodu FPGA, převzato z [2]

Na obrázku 1.2 jsou vidět jednotlivé části FPGA obvodu, kterými jsou I/O bloky, programovatelné spoje a logické obvody. Na následujících pár řádcích si o jednotlivých částech obvodu řekneme něco víc.

I/O bloky jsou umístěny po vnějším okraji obvodu a zajišťují komunikaci logických bloků s okolními obvody. Dále tyto bloky plní obdobné funkce jako vstupní a výstupní zesilovače, ochranné obvody apod.

Logické bloky, někdy také nazývané CLB, se skládají ze dvou základních komponentů, kterými jsou binární posuvné registry (Flip-Flops) a převodní tabulky (LUT), složené z malého množství paměti RAM.

Programovatelné spoje jsou na povrchu čipu spojeny v globální komunikační matici. Ta slouží k propojení logických obvodů po celé struktuře obvodu. Při jejím využívání však dochází k určitému zpoždění. To je možné odstranit při komunikaci sousedních bloků, které jsou mezi sebou propojeny i mimo globální matici.

Pro vytvoření nového VI v FPGA je nutné vybrat jako cíl FPGA Target(CRIO). Programování již probíhá naprosto běžně pomocí bloků. Kompilace programu probíhá v délce několika minut či desítek minut. U velmi složitých programů až několik hodin.

## 1.3 Real-Time system

Toto označení používáme pro operační systém pracujícím synchronně s reálným časem [3] a jehož základním požadavkem je včasná odpověď za jakýchkoliv okolností. Na rozdíl od běžného operačního systému, který je vyvinut pro práci s více programy najednou, je systém reálného času navrhnout tak, aby určité (kritické) operace běžely s naprostou spolehlivostí. Kritické operace vykonává v předem definovaných časových intervalech. Pokud systém zaručuje, že v žádném případě nebude překročen časový interval, nazýváme ho jako tvrdý systém reálného času (anglicky hard real-time). Pokud systém zaručí pouze maximální většinu času, nazýváme je měkký systém reálného času (soft real-time).

Pro programování v reálném čase musíme využít LabVIEW Real-Time modul. Následně v LabVIEW vytvoříme nové VI, avšak musíme jej přiřadit k real-time hardwaru, což je v našem případě šasi CompactRIO. Samotné programování již probíhá běžně pomocí bloků. Oproti FPGA je možno přiřadit určité aplikaci či části kódu vyšší prioritu vůči ostatním, to jest pokud při vykonávání kódu přijde požadavek s vyšší prioritou je vykonávaná operace přerušena a systém se k ní navrátí ihned po vykonání příkazu s vyšší prioritou. Kompilace programu je proti FPGA několikanásobně rychlejší.

## 2 POUŽITÉ MODULY C SERIES

### 2.1 C series modules

Jednotlivé moduly jsou konstruovány za předem danou úlohou. Ať už se jedná o měření napětí, proudu, teploty či modulů určených pro ovládání servopohonů či krokových motorů. Každý obsahuje kompletní obvody pro zpracování dat, včetně těch určených pro konverzi mezi analogovou a digitální částí. Moduly je možno zaměňovat s těmi určenými pro NI Compact DAQ. Dále jsou opatřeny hodnocením do prostředí s nebezpečím výbuchu. Některé mají specifikaci hodnocení vyšší než samotné CompactRIO. Na obrázku 2.1 jsou vidět příklady C-series modulů.



Obr. 2.1: Ukázka použitých C-series modulů

### 2.2 NI 9401

Vysokorychlostní osmikanálový vstupně/výstupní C-series modul využívající 5 V/TTL logiku [4]. Opakovací frekvencí je rovna 10 MHz. Při konfiguraci je možno zvolit zda se bude jednat o digitální vstupy nebo výstupy, probíhá po 4 bitech (někdy také značeno nibble). Pro připojení slouží 25 pinový D-Sub konektor. Na ten jsou

přivedeny vstupně/výstupní kanály (8 pinů), uzemnění (9 pinů) a zbylých 8 pinů není připojeno.

## 2.3 NI 9203

Jedná se o osmikanálový analogový modul pro sběr dat [5]. Jednotlivé vstupy mají programovatelné rozsahy buďto  $\pm 20$  mA nebo 0 až 20 mA. Na pouzdře je vyveden 10 pinový konektor. Na 0. až 7. pin jsou připojeny vstupní kanály, 8. je nepřipojený a 9. slouží jako zemnicí svorka. Maximální vzorkovací frekvence je 200 kS/s při 16-bitovém rozlišení. Modul obsahuje ochranu proti přechodným signálům a je odolný vůči rušení.

## 2.4 NI 9215

Čtyřkanálový modul pro zpracování analogového napětového signálu o maximálním rozsahu  $\pm 10$  V [6]. Obsahuje 16-ti bitový analogově digitální převodník s postupnou aproximací. Připojení měřených periférií je možno buďto pomocí čtyř koaxiálních kabelů nebo pomocí osmi klasických vodičů. Dále je vyveden pin určený pro uzemnění. Kalibrace modulu splňuje normy NIST.

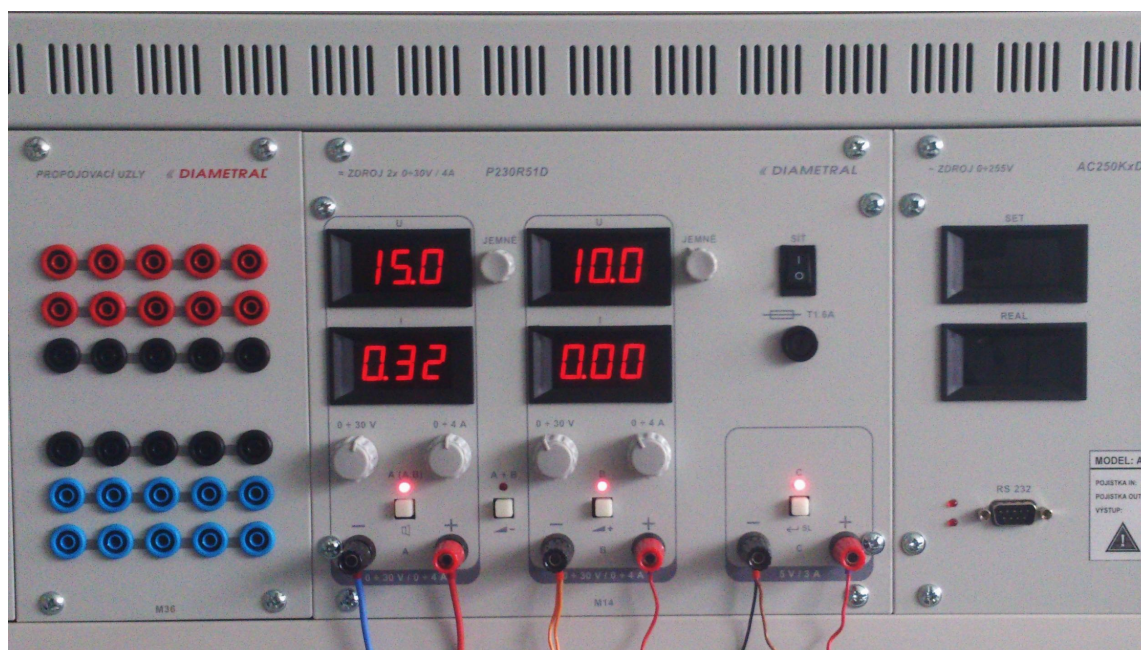


### 3 NAPÁJENÍ

Pro napájení CompactRIA a všech součástí modelu je třeba tří různých napětí. Z tohoto důvodu je vhodný laboratorní zdroj Diametral P230R51D [7].

#### 3.1 Diametral P230R51D

Dle obrázku 3.1 se zdroj skládá z jednoho pevného a dvou oddělených plynule regulovatelných zdrojů. Oba regulovatelné zdroje mají rozsah 0 až 30 V a proudové omezení od 0,1 do 4 A. Oba je možno lehce spojit do zdroje  $\pm 30$  V nebo 0 až 60 V. Pro zobrazení nastavené hodnoty proudu a napětí nám slouží LED zobrazovače. V našem případě je jeden nastaven na 15 V pro napájení CompactRIA a senzoru vzdálenosti, druhý na 10 V pro napájení snímače náklonu roviny. Pevný zdroj má výstupní napětí 5 V, které využíváme pro napájení servomotoru.

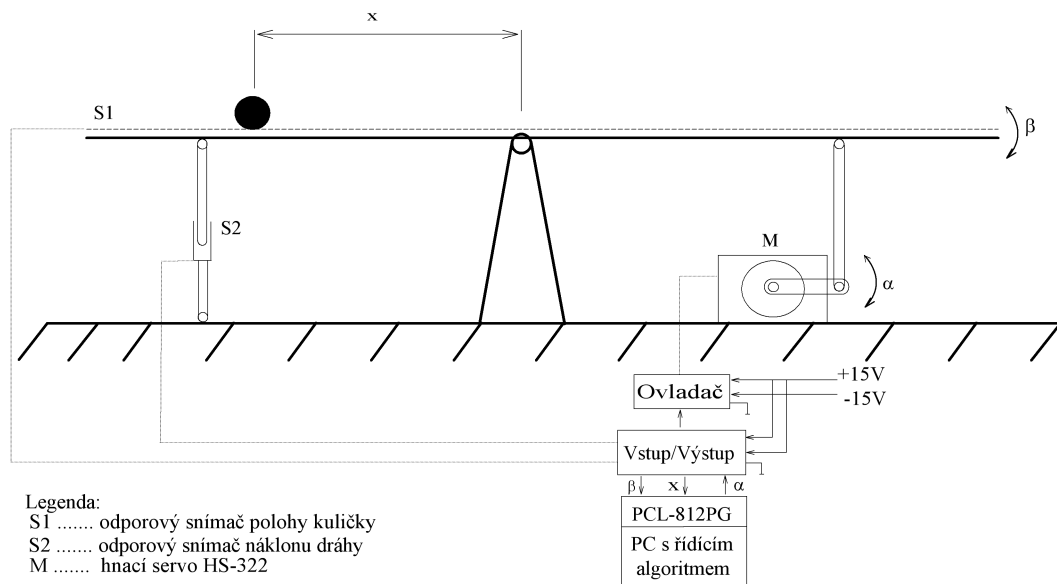


Obr. 3.1: Napájecí zdroj Diametral P230R51D

## 4 HARDWAROVÉ PŘÍSLUŠENSTVÍ MODELU

### 4.1 Původní model

Na obrázku 4.1 je schematicky znázorněn model původního přípravku. Ten byl řízen za pomoci PC s vloženou V/V analogovou kartou PCL812PG. Poloha kuličky byla snímána pomocí odporového snímače, u kterého jako jezdec sloužila samotná kulička.



Obr. 4.1: Schématické zobrazení původního modelu

### 4.2 Odporový snímač náklonu

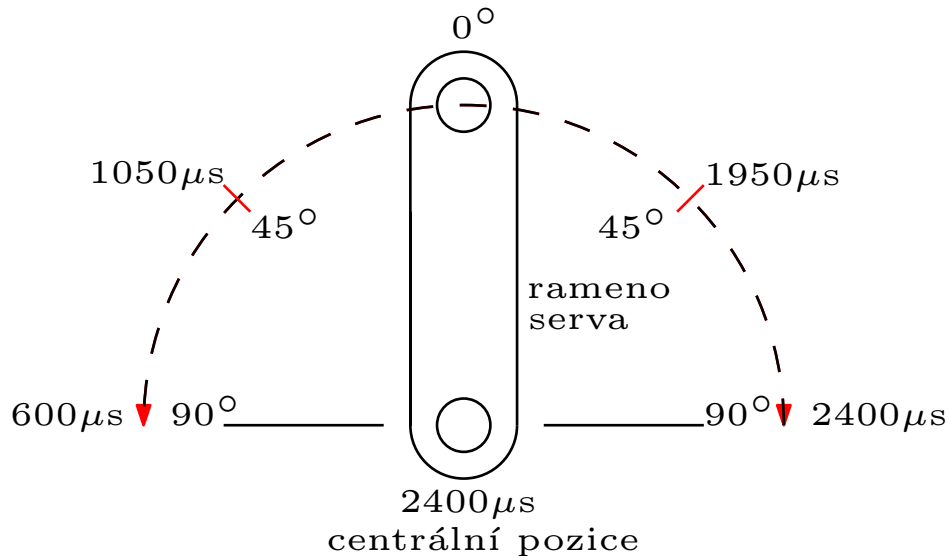
Jedná se o spojitý kovový odporový snímač polohy. Ten pracuje jako napěťový dělič, jehož poměr je určen polohou roviny. Dělič je napájen ze zdroje napětí 10 V. Závislost odporu na poloze je lineární. Mezi vodiči barvy žlutá a hnědá je hodnota odporu pro dolní polohu  $10\text{ k}\Omega$  a naopak pro horní  $6\Omega$ . Tomu odpovídají výstupní napětí měřená pomocí modulu NI 9215  $U_1 = 0\text{ V}$  resp.  $U_2 = 10\text{ V}$ . Díky tomuto snímači máme informace o poloze a můžeme ji následně řídit pomocí servomotoru.

### 4.3 Servo HS-322

HS-322 [8] je výkonné a spolehlivé servo využívající karbonitových převodů. Díky tomu je takřka 4 krát pevnější než serva využívající nylonové převody. V závislosti

na velikosti řídicího pulzu a délce jeho trvání se servo natočí do dané polohy. Je možno dosáhnout pracovního rozsahu až  $180^\circ$ .

Pro řízení serva jsou potřeba tři vodiče (hnědá, červená a žlutá). Hnědá slouží jako zemnicí vodič, červená pro přívod napětí v rozmezí 4,8 až 6 V. Pomocí žlutého vodiče přivádíme řídicí pulz s amplitudou +3 až +5 V. Tyto pulzy musí být obnovovány s frekvencí 100 Hz (10 ms). Jak je vidět z obrázku 4.2 je na délce řídicího pulzu závislý úhel natočení.



Obr. 4.2: Závislost polohy na délce pulzu

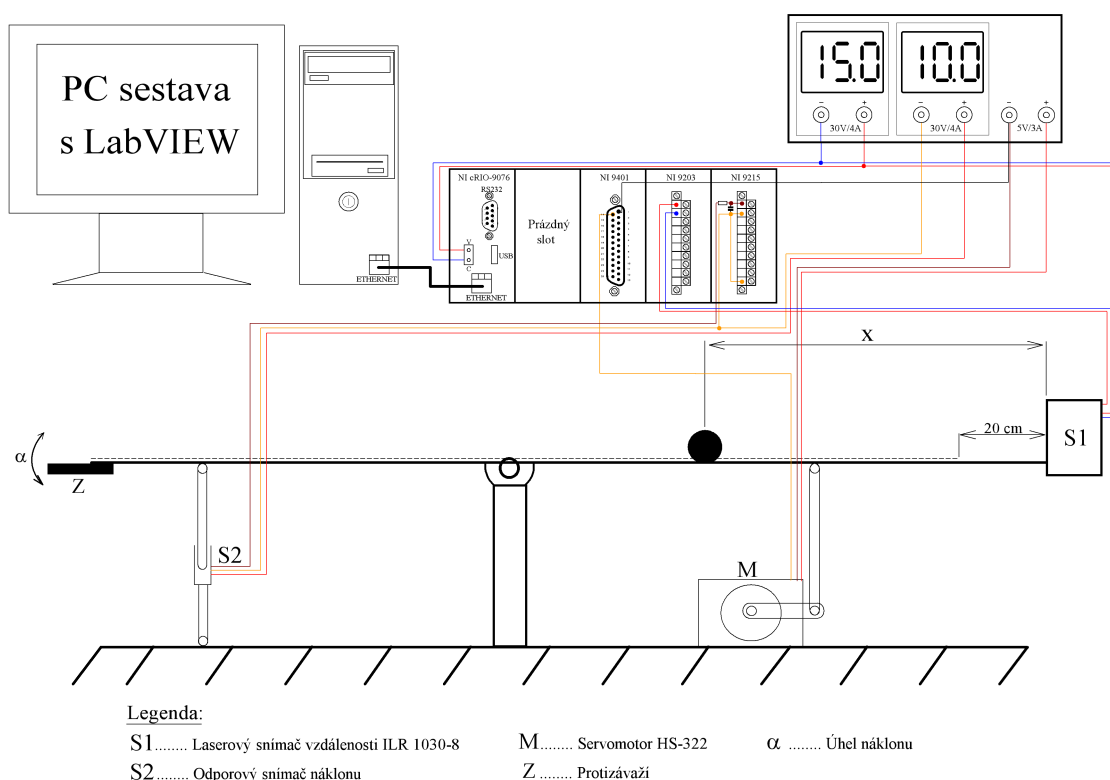
## 4.4 ILR 1030-8

Laserový senzor vzdálenosti [9], který využívá technologie měření času letu světla. Neboli měří vzdálenost z času mezi vysláním a obdržením paprsku, který se odrazil od překážky jejíž vzdálenost měříme. Výstupní veličinou je v tomto případě proud. Přístroj je schopen pracovat ve dvou režimech, a to ve spínacím nebo analogovém. V našem případě je použit druhý režim. Měřicí rozsah tohoto přístroje je 0,2 až 8 m, přičemž těmto vzdálenostem odpovídá výstupní proud 4 až 20 mA. Rozsah je závislý na barvě materiálu překážky, pro černou barvu je nejmenší, přesněji 0,2 až 2,5 m a naopak pro bílou barvu největší 0,2 až 8 m. Senzor ILR 1030-8 se vyznačuje velmi krátkou dobou odezvy. Proudová smyčka 4 až 20 mA je standart, který se využívá pro přenos naměřených hodnot v oblasti průmyslové automatizace. Je vhodný pro přenos dat na velké vzdálenosti (až stovky metrů), a také umožňuje přímé napájení připojených komponentů. Minimální hodnota 4 mA je využívána z ohledem na

možnou detekci poruchy. Další výhodou je snadná detekce přerušení smyčky, a to v případě že proud klesne na nulovou hodnotu.

## 4.5 Popis modelu

Na obrázku 4.3 je vidět celkové schéma modelu včetně všech jeho součástí. Model se skládá z nakloněné roviny, po které se pohybuje ocelová kulička, která je k dřevěnému podstavci připevněna pomocí ložiskového kloubu. Poloha kuličky je snímána pomocí laserového snímače vzdálenosti, který musí být umístěn minimálně 20 cm od okraje roviny a tudíž bylo třeba vytvořit systém uchycení a vyvážení nakloněné roviny za pomoci protizávaží. Náklon roviny je řízen za pomoci servomotoru HS-322 a její úhel je snímán pomocí odporového snímače. Pro řízení celého modelu se využívá systém CompactRIO s jeho moduly pro snímání veličin a vytváření PWM. Z obrázku jsou dále patrná jednotlivá napájecí napětí a připojení na jednotlivé moduly za pomoci barevně rozlišených vodičů.

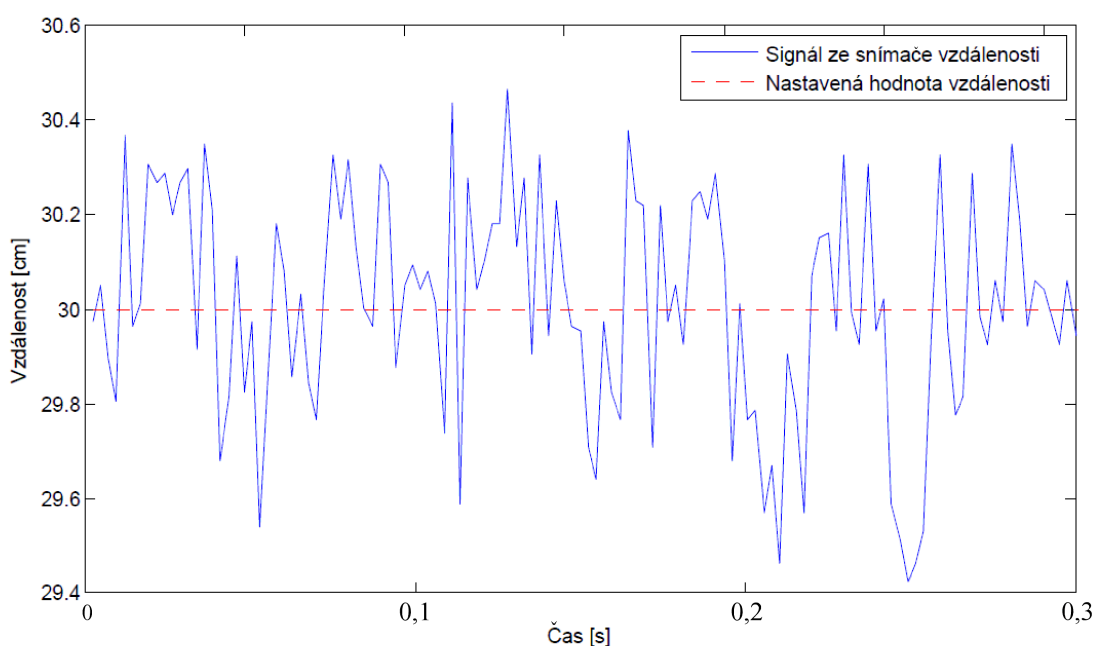


Obr. 4.3: Schématické zobrazení aktuálního modelu

## 5 FILTRACE SIGNÁLU

### 5.1 Filtrace proudové smyčky

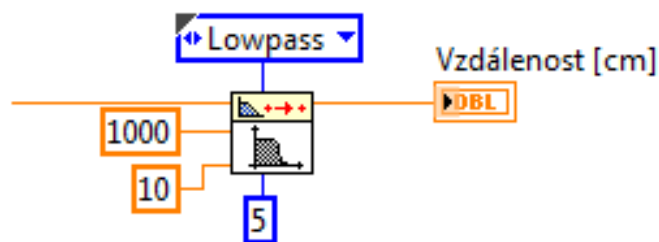
Filtrace signálu ze snímače vzdálenosti je nutná, vzhledem k velmi zašumělému signálu, který jde přímo ze snímače. Tento signál můžeme vidět na obrázku 5.1. Nastavená vzdálenost kuličky je v tomto případě 30 cm. Z průběhu můžeme vidět, že signál kmitá až v rozsahu 1 centimetru. Takový signál je vzhledem k velikosti nakloněné roviny nepoužitelný a z tohoto důvodu jsme museli vybrat vhodný filtr pro jeho filtraci.



Obr. 5.1: Nefiltrovaný průběh ze snímače vzdálenosti

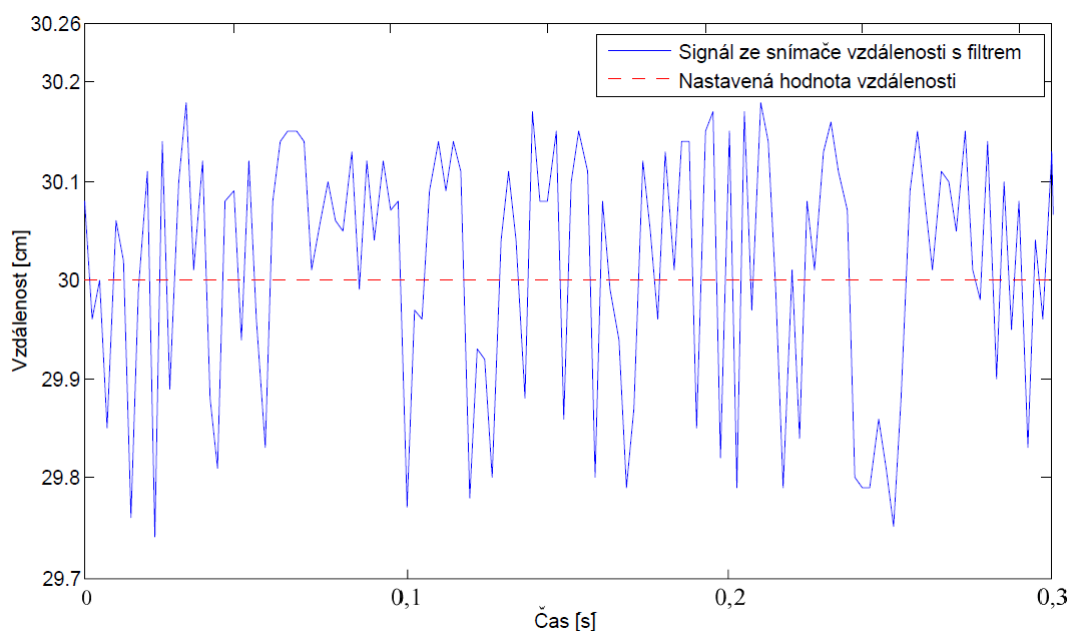
Vhodný filtr je typu dolní propust. My jsme použili Butterworth filtr a to na základě velmi ploché amplitudové charakteristiky v propustném pásmu, která začíná klesat až v blízkosti zlomového kmitočtu  $f_0$ . Další významnou vlastností je rozdíl mezi ideální a aproximovanou charakteristikou právě na kmitočtu zlomu, který je roven 3dB, a to bez vlivu použitého řádu filtru. Pro každý řád je dán normovaný polynom s komplexně sdruženými kořeny ležícími v levé polorovině. V našem případě využíváme filtr 5. řádu.

Nastavení jednotlivých konstant filtru vidíme na obrázku 5.2. Jedná se o filtr 5. řádu typu dolní propust se zlomovým kmitočtem  $f_0 = 10 \text{ Hz}$  a vzorkovací frekvencí  $f_{VZK} = 1 \text{ kHz}$ .



Obr. 5.2: Použitý Butterworthův fitlr v LabVIEW

Na obrázku 5.3 je zobrazen již filtrovaný signál snímače vzdálenosti, na kterém je vidět, že signál kmitá v rozsahu přibližně  $0,4\text{ cm}$ . To je oproti nefiltrovanému signálu, který kmital v rozsahu  $1\text{ cm}$  zlepšení o více než dvojnásobek.

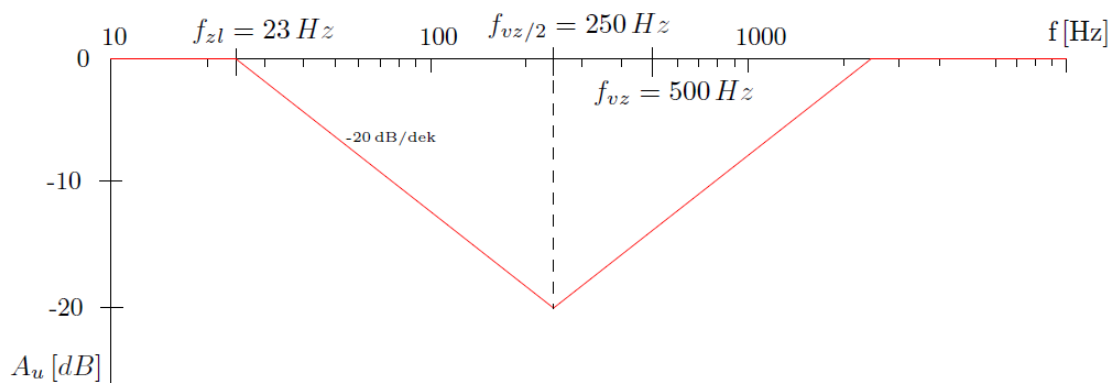


Obr. 5.3: Filtrovaný průběh ze snímače vzdálenosti

## 5.2 RC článek - dolní propust

Pro splnění Shannon-Kotelníkova vzorkovacího teorému je třeba použít RC článek. Tento teorém nám říká, že vzorkovací frekvence musí být alespoň dvakrát vyšší než je nejvyšší frekvence spektra vzorkovaného signálu. V našem případě je vzorkovací frekvence  $f_{vz} = 500\text{ Hz}$ . Dolní propust potřebujeme navrhnout tak, aby na frekvenci rovné  $1/2$  vzorkovací frekvence, to jest  $f_{vz/2} = 250\text{ Hz}$  bylo dosaženo co nejvyššího potlačení. Dle obrázku 5.4 vidíme, že zlomová frekvence je nastavena na

$f_{zl} = 23 \text{ Hz}$ , což na frekvenci  $f_{vz/2}$  znamená potlačení  $-20 \text{ dB}$  neboli přes tento filtr projde jen  $1/10$  vstupního signálu.



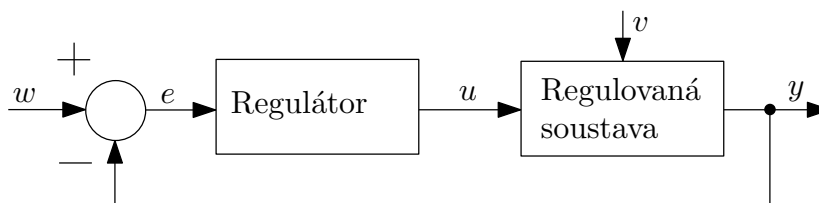
Obr. 5.4: Amplitudová frekvenční charakteristika

Zlomová frekvence  $f_{zl}$  je nastavena pomocí rezistoru  $R = 100 \text{ K}\Omega$  a kondenzátoru  $C = 70 \text{ nF}$  na frekvenci přibližně  $23 \text{ Hz}$  dle rovnice (5.1).

$$f = \frac{1}{2\pi RC} = \frac{1}{2 \cdot \pi \cdot 10 \cdot 10^4 \cdot 70 \cdot 10^{-9}} = 23 \text{ Hz} \quad (5.1)$$

## 6 REGULACE

Na obrázku 6.1 vidíme regulační obvod, který se skládá z regulátoru, regulované soustavy a záporné zpětné vazby. Na vstup regulátoru přivádíme regulační odchylku ( $e$ ), která je rozdílem žádané ( $w$ ) a regulované veličiny ( $y$ ). Výstupem regulátoru je akční veličina značená jako  $u$ . Do soustavy v některých případech může vstupovat porucha ( $v$ ), která ovlivňuje chování soustavy.



Obr. 6.1: Schéma regulačního obvodu

Regulace přípravku jedním regulátorem ztroskotala na problému ovlivňování úhlu náklonu roviny pozicí kuličky. Toto se projevuje jako porucha, kterou není regulátor schopen vyregulovat. Z tohoto důvodu jsme se rozhodli pro použití kaskádní regulace. V té použijeme vnitřní smyčku pro řízení úhlu náklonu a tím pádem k odstranění poruchy způsobené pozicí kuličky. Druhý regulátor již použijeme pro řízení polohy kuličky.

### 6.1 Kaskádní regulace

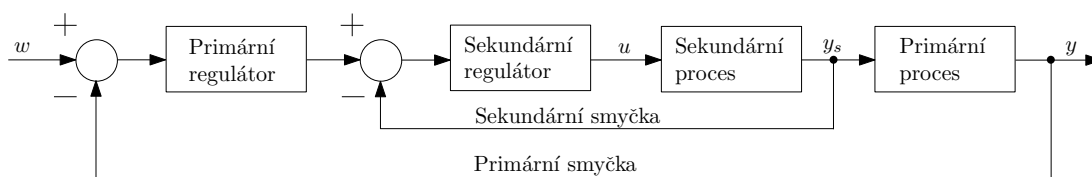
Někdy také nazývána jako regulace s pomocnou regulační veličinou [10]. Na obrázku 6.2 je blokové schéma kaskádní regulace, ze které vidíme zapojení regulátorů a procesů. Kaskádní regulace je přínosná v případech, kdy můžeme regulovaný proces rozdělit na dvě na sobě nezávislé části, což je zároveň základním předpokladem, jelikož potřebujeme měřit nezávisle oba výstupy regulovaného procesu  $y_s$  a  $y$ . Pro takový proces je možno použít i regulaci s jedním regulátorem, avšak při použití kaskádní regulace bude výsledek dosažen rychleji a bude kvalitnější potlačení vlivu poruch. Řečeno ve zkratce, při kaskádní regulaci využijeme obě měřené veličiny při jednom procesu, takže získáme lepší informaci než při regulaci pomocí odchylkového regulátoru, což vede k výraznému zkvalitnění regulace.

Tento typ regulace má tři základní výhody:

1. Porucha působící na sekundární proces neovlivní proces primární, jelikož je potlačen sekundárním regulátorem.
2. Zrychlení odezvy primárního procesu, neboli časová konstanta sekundární smyčky je menší než u jednoduché regulační smyčky.



3. Sekundární regulátor zlepšuje robustnost regulace, útlumem vlivu změny zesílení sekundárního procesu.



Obr. 6.2: Uspořádání smyček při kaskádní regulaci

## 6.2 Regulátor

Regulátorem [13] nazýváme zařízení, které prostřednictvím akční veličiny působí na veličinu regulovanou tak, aby ji udržovala na námi vybrané hodnotě, aneb aby regulační odchylka byla co nejmenší, nejlépe nulová.

**P-složka:** Použitím P-složky zvýšíme akční veličinu, kterou působíme na regulovanou soustavu a tím dosáhneme snížení trvalé ustálené odchylky, nikoliv její odstranění.

**I-složka:** Použitím této složky již dosáhneme odstranění trvalé ustálené odchylky. Vnáší však do systému astatismus (pól v počátku), což způsobí zmenšení zásoby stability a zpomalení regulačního děje.

**D-složka:** Zavádíme ji pro zvýšení stability a zrychlení regulačního děje. Při použití příliš velké D-složky dojde k rozkmitání akční veličiny a tím i soustavy, příliš malá může způsobit nestabilitu. Při použití D-složky zavádíme realizační konstantu  $\epsilon$ , kterou umístíme do jmenovatele regulátoru. Ta způsobí menší rozkmitávání regulačního obvodu.

### 6.2.1 PI regulátor

Výstupní (akční) veličina regulátoru se v tomto případě skládá ze dvou složek, kde první je úměrná regulační odchylce a druhá její integraci. Pro časové průběhy platí dle rovnice (6.1):

$$x(t) = r_0 e(t) + r_i \int_0^t e(t) dt + x(0) \quad (6.1)$$

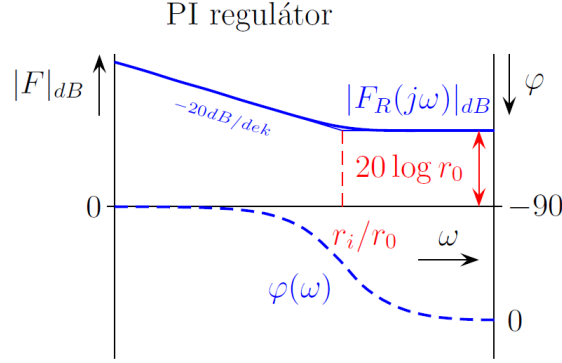
čemuž odpovídá přenos dle rovnice (6.2):

$$F(p) = \frac{X(p)}{E(p)} = r_0 + \frac{r_i}{p} = K_R \cdot \frac{T_R \cdot p + 1}{p} = \frac{T_R \cdot p + 1}{T_i \cdot p} \quad (6.2)$$

Kde poslední dva výrazy jsou jen jiným zápisem za pomoci zesílení  $K_R$  a časových konstant  $T_i$  a  $T_R$ , mezi kterými platí tyto vztahy:

$$K_R = r_i = \frac{1}{T_i} \quad ; \quad T_R = \frac{r_0}{r_i} \quad ; \quad r_0 = \frac{T_R}{T_i} \quad (6.3)$$

Na obrázku 6.3 vidíme frekvenční charakteristiku PI regulátoru.



Obr. 6.3: Frekvenční charakteristika PI regulátoru (převzato z [13])

## 6.2.2 PD regulátor

Akční veličina tohoto regulátoru se stejně jako v předchozím případě skládá ze dvou složek, tedy proporcionální a derivační. Časový průběh má tvar, který je vidět v rovnici (6.4):

$$x(t) = r_0 e(t) + r_d \cdot \frac{de(t)}{dt} \quad (6.4)$$

tomu odpovídá přenos v rovnici (6.5):

$$F_R(p) = \frac{X(p)}{E(p)} = r_0 + r_d \cdot p = K_R \cdot (T_d \cdot p + 1) \quad (6.5)$$

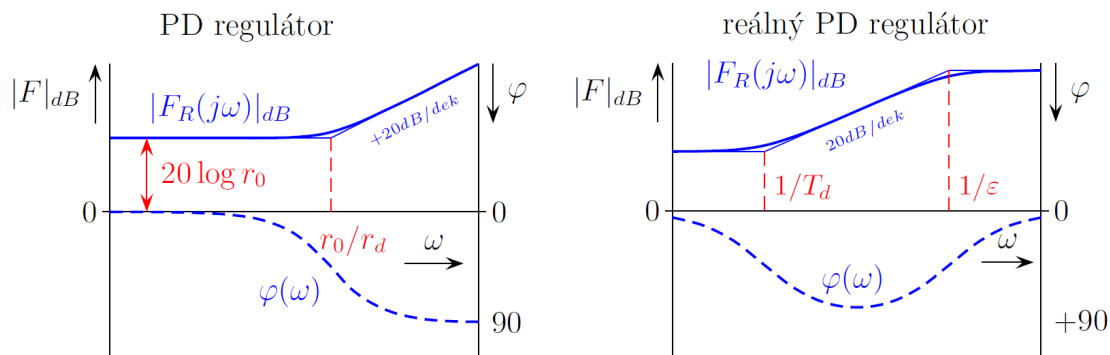
Kde poslední výraz je opět jen jiným zápisem za pomoci zesílení  $K_R$  a časové konstanty  $T_d$ . Pro jejich výpočet platí:

$$K_R = r_0 \quad ; \quad T_d = \frac{r_d}{r_0} \quad (6.6)$$

Tento zápis PD regulátoru je však fyzikálně nerealizovatelný a proto zavádíme realizační konstantu  $\epsilon$ , kterou umístíme do jmenovatele regulátoru, tak jak vidíme v rovnici (6.7). Její velikost je volena tak, aby co nejméně ovlivňovala chování ideálního regulátoru. Proto musí platit  $\epsilon \ll T_d$ .

$$F_R(p) = \frac{X(p)}{E(p)} = r_0 + r_d \cdot p = K_R \cdot \frac{(T_d \cdot p + 1)}{(\epsilon \cdot p + 1)} \quad (6.7)$$

Na obrázku 6.4 vidíme porovnání frekvenčních charakteristik ideálního (v levé části) a reálného (v pravé části) PD regulátoru.



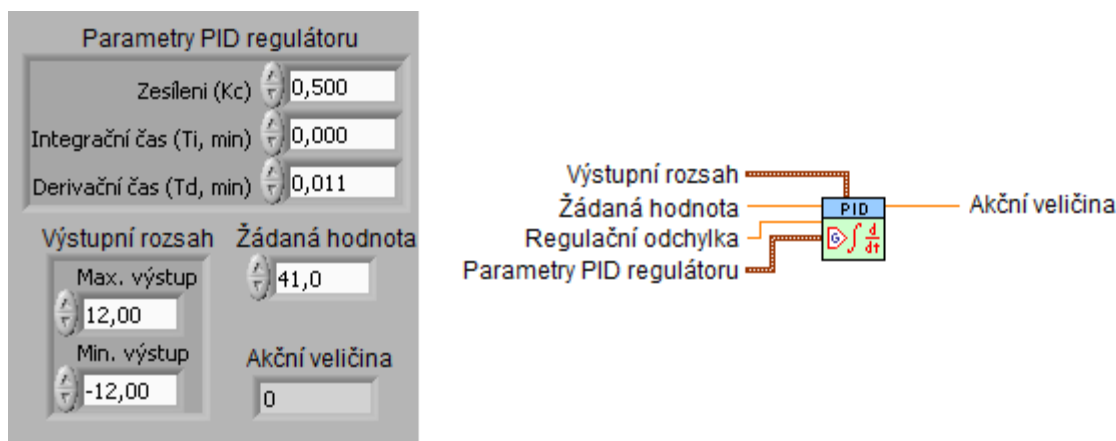
Obr. 6.4: Frekvenční charakteristiky ideálního a reálného PD regulátoru (převzato z [13])

### 6.3 Použitý regulátor

Jako regulátor vnitřní (sekundární) smyčky byl zvolen PI regulátor. Byl vybrán z důvodu odstranění vlivu poruchy.

Pro regulaci vnější smyčky je použit PD regulátor, který i bez I složky dosáhne nulové trvalé regulační odchylky, jelikož soustava na kterou tento regulátor navrhujeme již obsahuje dva integrátory, tudíž regulátor nemusí obsahovat I složku.

Na obrázku 6.5 v pravé části vidíme ikonu subVI námi používaného PID regulátoru. Ten zapojíme do schématu a dle požadavků si nastavíme výstupní rozsah a požadovanou hodnotu. V levé části obrázku vidíme část sloužící pro nastavení všech parametrů regulátoru.



Obr. 6.5: Použitý PID regulátor

V našem případě, kdy vytváříme pouze PI nebo PD regulátor provedeme vypnutí nepoužívané I nebo D složky jejich nastavením na hodnotu nula. Nikoliv nastavením I složky na nekonečno jak je často psáno v literatuře.

## 7 IDENTIFIKACE SYSTÉMU

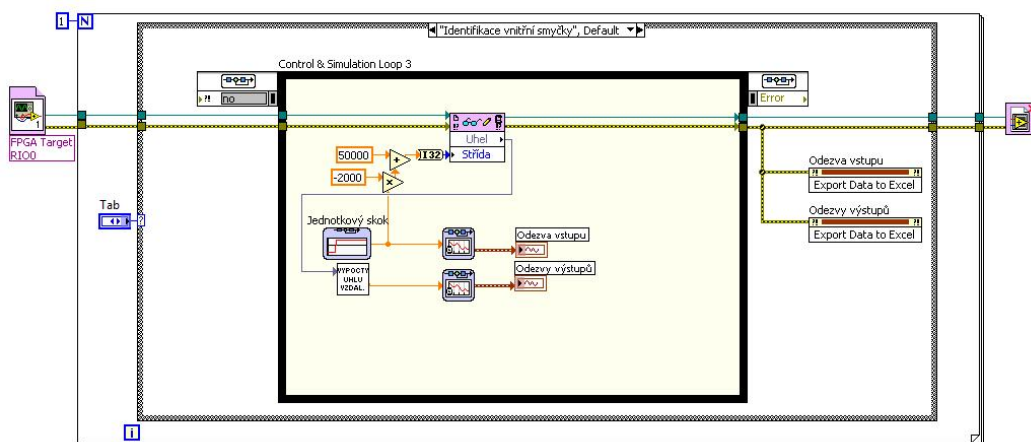
Identifikací dynamického systému se rozumí nalezení dynamického popisu experimentální cestou, to jest z dostupných signálů. Těmi jsou nejčastěji vstupní a výstupní signály. Identifikací zjistíme hodnoty parametrů, časové konstanty systému. V našem případě využíváme k získání matematického popisu systému deterministickou metodu v časové oblasti, kdy jako vstupní signál uvažujeme jednotkový skok  $h(t)$ . Identifikaci systému kuličky na nakloněné rovině jsme rozdělili na dvě části, tou první je vnitřní smyčka, neboli soustava *Požadovaný úhel náklonu –  $U_{hel}$*  (otevřená smyčka) a druhou je vnější smyčka *Požadovaný úhel náklonu –  $U_{hel}$*  (zpětnovazební obvod vnitřní smyčky s již navrženým PI regulátorem).

Postup identifikace si popíšeme na vnitřní smyčce, pro vnější smyčku postup nebudeme znovu popisovat jelikož je totožný, popíšeme si jen hlavní části.

### 7.1 Identifikace vnitřní smyčky

#### 7.1.1 Získání dat pro identifikaci

Pro získání vstupních a výstupních dat využíváme modul LabView nazvaný Control Design and Simulation. Pomocí tohoto modulu jsme si vytvořili simulační časovou smyčku, kterou můžeme vidět na obrázku 7.1 pracující na frekvenci  $1\text{ kHz}$  s periodou  $10\text{ ms}$ . Uvnitř této smyčky si vytvoříme vstupní signál, což je v tomto případě jednotkový skok požadovaného úhlu náklonu. Pokud však chceme pohybovat s náklonem roviny, musíme požadovaný úhel přepočítat na vstupní hodnoty servomotoru (Střída) v rozsahu 24000 až 74000. Pomocí funkce pro tvorbu časového průběhu si vytvoříme průběh, který si následně zobrazíme pomocí prvku waveform chart.

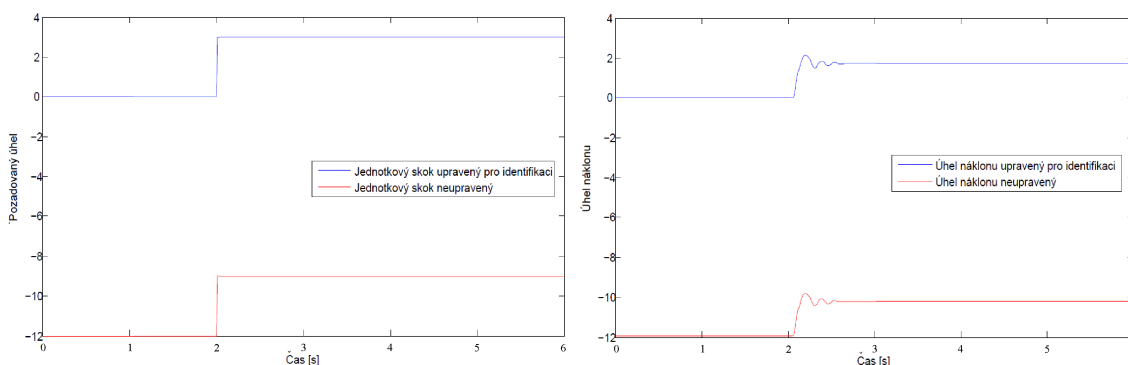


Obr. 7.1: Získávání dat k identifikaci vnitřní smyčky

Po dokončení simulační smyčky se data z průběhů pomocí funkce Invoke Node (Export Data to Excel) vyexportují do Excelu, díky čemuž si můžeme data jednoduše přesunout do Matlabu a dále zpracovávat.

### 7.1.2 Identifikace z přechodových charakteristik

Přechodovou charakteristikou nazýváme grafické znázornění přechodové funkce  $h(t)$ . Neboli odezvu na jednotkový skok při nulových počátečních podmínkách. Dříve nežli můžeme data použít pro identifikaci, musíme si průběhy upravit, viz obrázek 7.2, do normovaného tvaru, to jest s počátkem na hodnotě 0 a rostoucím charakterem s časem. Následně upravené charakteristiky použije pro identifikaci soustavy za pomoci System identification toolbox.



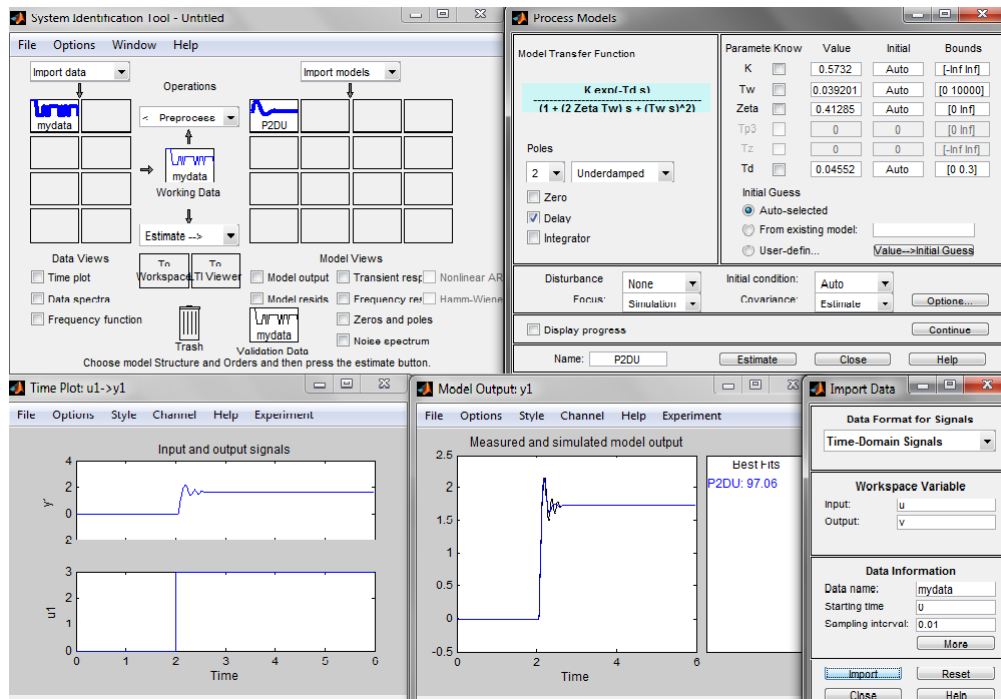
Obr. 7.2: Časové průběhy odezev na jednotkový skok

### 7.1.3 System identification toolbox

Jedná se o nástroj programu Matlab, umožňující navrhovat matematické modely soustav z naměřených vstupních a výstupních dat. V našem případě jej využíváme pro získání přenosu vnitřní soustavy, na kterou jsme následně metodou frekvenčních charakteristik navrhli regulátor. Obrázek 7.3 zobrazuje návrh modelu soustavy 2. řádu s komplexními kořeny a dopravním zpožděním. Ta se shoduje s naměřenými daty na 97,06%. Soustavu si následně přesuneme do pracovního prostoru Matlabu (*Workspace*), kde s ní budeme dále pracovat.

Přenos identifikovaného modelu soustavy vnitřní smyčky vidíme v rovnici (7.1).

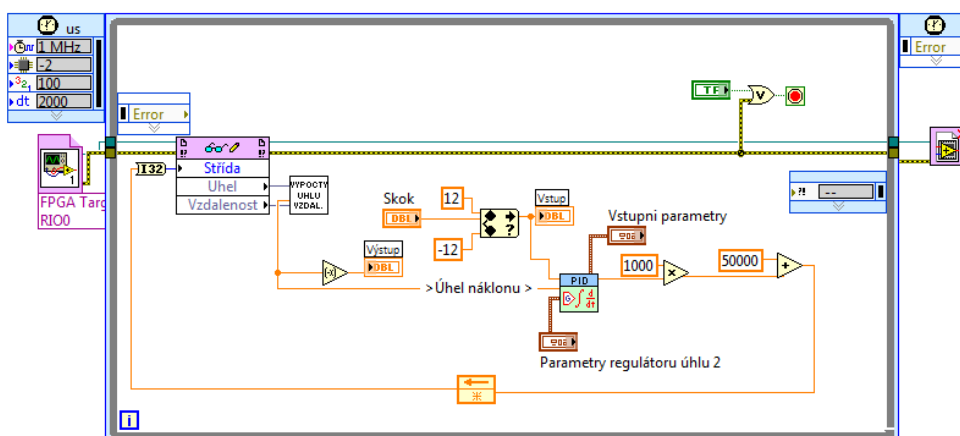
$$F_S(p) = \frac{0.5732}{0.001537p^2 + 0.03237p + 1} \cdot e^{(-0,0455p)} \quad (7.1)$$



Obr. 7.3: Návrh soustavy pomocí Ident

## 7.2 Získání dat pro identifikaci vnější smyčky

Data získáváme pomocí časové smyčky s periodou  $2\text{ ms}$ , kterou vidíme na obrázku 7.4. Ta obsahuje zpětnovazební obvod vnitřní smyčky s navrženým PI regulátorem. Na vstup přivádíme skokovou změnu požadovaného úhlu a sledujeme změnu skutečného úhlu náklonu roviny.



Obr. 7.4: Získávání dat pro identifikaci vnější smyčky

Data si dle návodu na čelním panelu aplikace vyexportujeme do Excelu a následně přesuneme do Matlabu. Zde opět provedeme identifikaci pomocí toolboxu System identification. Identifikovaný model soustavy shodující se s naměřenými daty na 95,43% je 2. řádu s reálnými kořeny. Přenos identifikovaného modelu soustavy vnější smyčky vidíme v rovnici (7.3).

$$F_S(p) = \frac{1,004}{0.007356p^2 + 0.1715p + 1} \quad (7.2)$$

Pro následný návrh regulátoru musíme k soustavě přidat dva integrátory, neboli dva póly v počátku, které nám reprezentují změnu úhlu náklonu (zrychlení kuličky) na výslednou polohu kuličky. Přenos soustavy pro návrh regulátoru tedy vidíme v rovnici (7.3).

$$F_S(p) = \frac{1,004}{0.007356p^4 + 0.1715p^3 + p^2} \quad (7.3)$$

## 7.3 Návrh regulátoru metodou frekvenčních charakteristik

Frekvenční charakteristika [13] je grafickým vyjádřením amplitudy  $|F(j\omega)|$  a fáze  $\varphi(\omega)$  frekvenčního přenosu systému v závislosti na frekvenci v logaritmických souřadnicích. Tato metoda návrhu spočívá se tvarování frekvenční charakteristiky otevřeného obvodu tak, abychom dosáhli co nejvhodnějšího tvaru. Tvarování provádíme v logaritmických souřadnicích, díky čemuž můžeme tento návrh provést ručně bez použití výpočetní techniky. Při samotném návrhu však musíme dodržet dva požadavky, kterými jsou:

1. Co nejvyšší hodnota frekvence řezu  $\omega_{\text{ř}}$ , která určuje rychlost přechodového děje.  $\omega_{\text{ř}}$  určíme jako frekvenci, při které amplitudová charakteristika nabývá hodnoty  $0 \text{ dB}$ .
2. Co nejvyšší hodnota fázové bezpečnosti, což nám zajistí malý překmit na přechodové charakteristice. Tento požadavek se dá také vyjádřit tak, že amplitudová část charakteristiky má procházet osou  $0 \text{ dB}$  pod sklonem  $-20 \text{ dB/dek}$  a držet jej v co největším okolí  $\omega_{\text{ř}}$ .

### 7.3.1 Posouzení stability

Pro posouzení jak daleko jsme při návrhu od nestability používáme pojmy fázová a amplitudová bezpečnost.

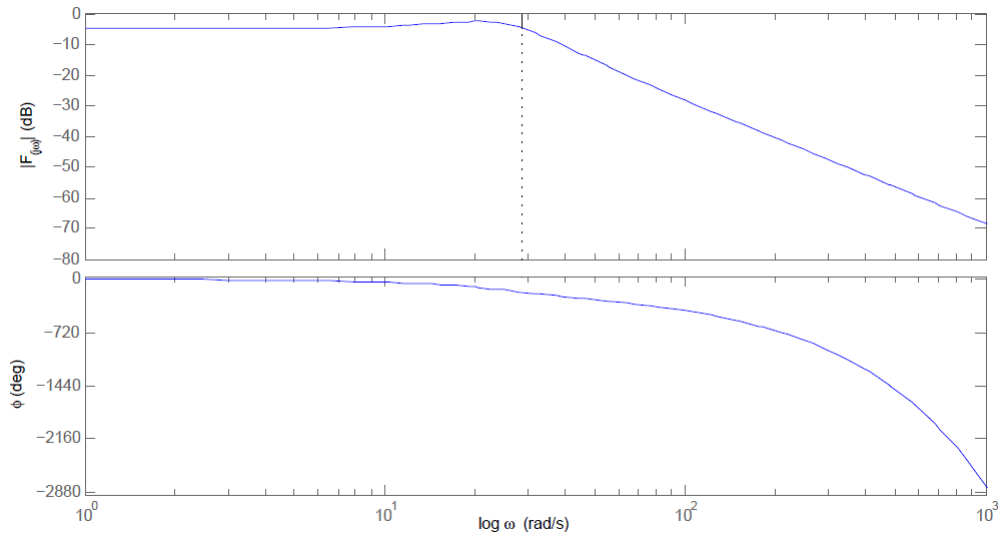
- a) Amplitudová bezpečnost nám říká, jakou hodnotou můžeme vynásobit aktuální hodnotu zesílení v otevřené smyčce, abychom uzavřenou smyčku přivedli na mez stability. Nejčastěji se uvádí v decibelech.
- b) Fázová bezpečnost je záporně vzatá změna fáze otevřeného obvodu, kterou přivedeme uzavřená obvod na mez stability.

### 7.3.2 Návrh regulátoru vnitřní smyčky

Nejprve vybereme vhodný typ regulátoru, který musí obsahovat I složku pro odstranění poruchy způsobené polohou kuličky. Na základě této podmínky jsme vybrali PI regulátor, jehož návrh si následně popíšeme v několika bodech:

1. Vykreslíme si frekvenční charakteristiku soustavy v logaritmických souřadnicích (obrázek 7.5),





Obr. 7.5: Frekvenční charakteristika identifikovaného modelu soustavy

2. Napíšeme si přenos PI regulátoru:

$$F_R(p) = K_R \cdot \frac{T_i p + 1}{p}$$

3. Dle tvaru frekvenční charakteristiky regulátoru nastavíme časovou konstantu regulátoru  $T_i$  dle druhého z výše uvedených požadavků na hodnotu  $T_i = 0,034 \text{ s}$ ,
4. Požadavek na fázovou bezpečnost  $70^\circ$  splníme pokud bude hodnota zesílení regulátoru  $K_c = 11,85$ ,

Přenos takto navrženého regulátoru je uveden v rovnici (7.4).

$$F_R(p) = 11,85 \cdot \frac{0,034p + 1}{p} \quad (7.4)$$

Pro zápis navrženého regulátoru do aplikace Kulicka je třeba přepočítat jak tvar regulátoru (rovnice 7.5), tak i časovou konstantu (rovnice 7.6).

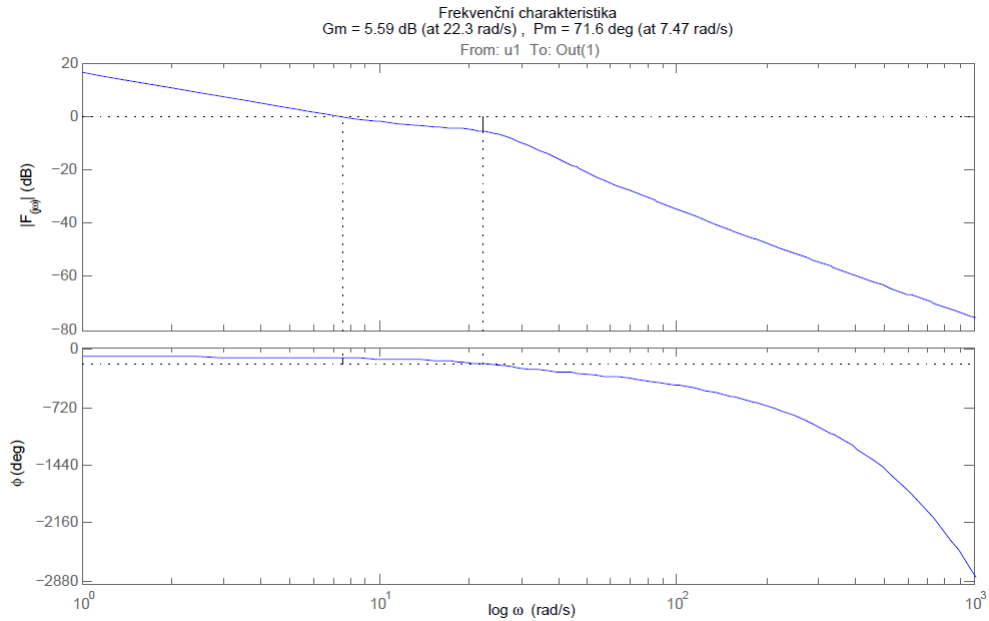
$$F_{RPI}(p) = 11,85 \cdot \frac{0,034p + 1}{p} = 11,85 \cdot 0,034 \cdot \left(1 + \frac{1}{0,034p}\right) \quad (7.5)$$

$$T_i \text{ labview} = \frac{T_i}{60} = \frac{0,034}{60} = 0,00057 \quad (7.6)$$

Výsledný tvar regulátoru je uveden v rovnici (7.7).

$$T_{RPILB}(p) = 0,4029 \cdot \left(1 + \frac{1}{0,00057p}\right) \quad (7.7)$$

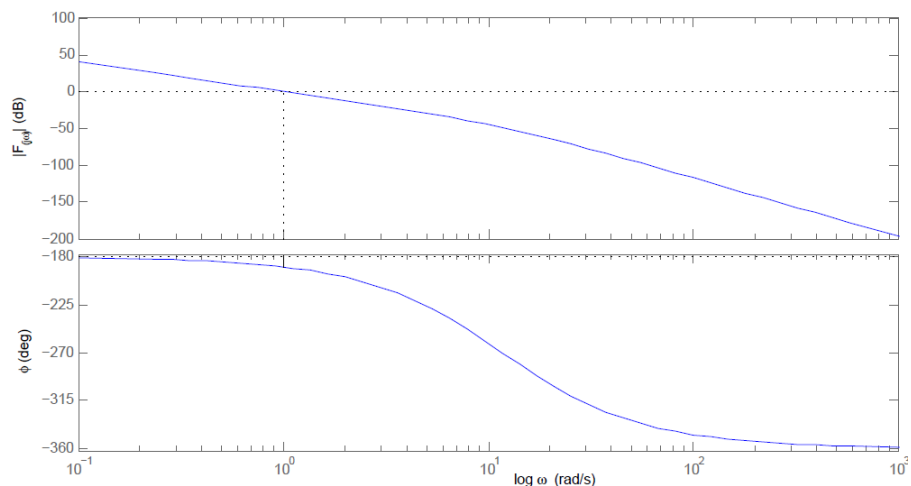
Na obrázku 7.6 si můžeme prohlédnout frekvenční charakteristiku otevřeného obvodu vnitřní smyčky.



Obr. 7.6: Frekvenční charakteristika otevřeného obvodu vnitřní smyčky

## 7.4 Návrh regulátoru polohy kuličky

Nejprve jsme opět museli vybrat vhodný typ regulátoru. Jelikož soustava, pro kterou tento regulátor navrhujeme obsahuje dva integrátory, tak bude nejlépe vyhovující PD regulátor. Na obrázku (7.7) vidíme frekvenční charakteristiku soustavy v logaritmických souřadnicích.



Obr. 7.7: Frekvenční charakteristika identifikovaného modelu soustavy rozšířené o dva integrátory

Postup je obdobný jako v případě návrhu PI regulátoru, to jest nejprve zvolíme časovou konstantu PD regulátoru  $T_D = 0,9 \text{ s}$  a pro splnění fázové bezpečnosti  $30^\circ$

nám vyšlo zesílení  $K = 6,7$ . Přenos regulátoru je uveden v rovnici (7.8).

$$F_{RPD}(p) = 6,7 \cdot (0,9p + 1) \quad (7.8)$$

Zesílení regulátoru  $K$  je však nutné přepočítat dle rovnice (7.9).

$$K_c = \frac{K}{A} = \frac{6,7}{23} = 0,29 \quad (7.9)$$

Konstanta  $A = 23 \text{ cm/s}^2$  reprezentuje zrychlení kuličky na jeden stupeň náklonu roviny, zjistili jsme ji experimentálně změřením doby pohybu kuličky ( $t = 0,39 \text{ s}$ ) přes délku roviny ( $x = 42 \text{ cm}$ ) v náklonu  $\alpha = 12^\circ$  a vypočetli dle vztahu 7.10.

$$a = \frac{x}{t^2} = \frac{42}{0,39^2} = 276 \text{ cm/s}^2$$

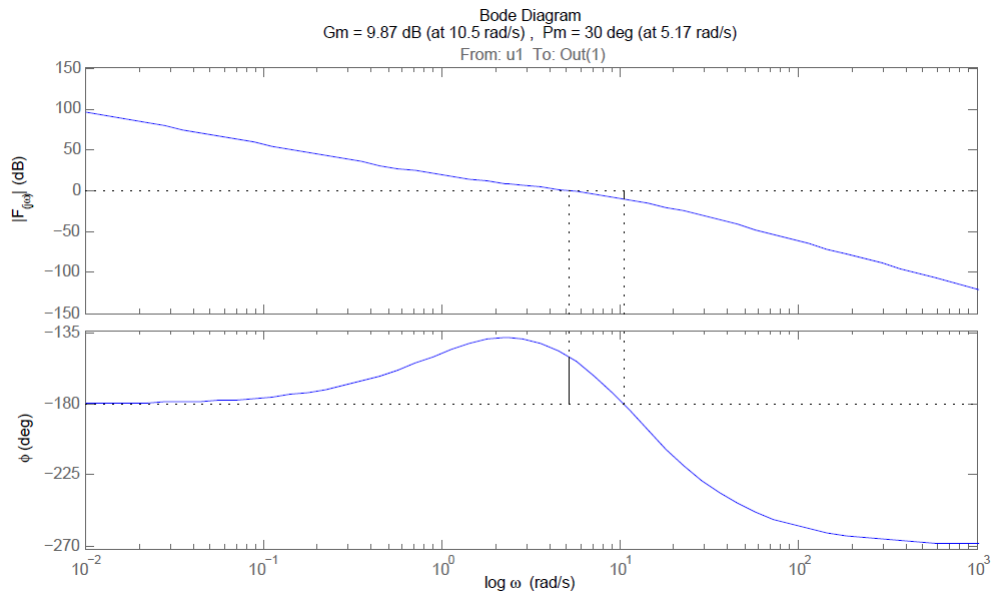
$$A = \frac{a}{\alpha} = \frac{276}{12} = 23 \text{ cm/s}^2/^\circ \quad (7.10)$$

$$(7.11)$$

Pro zápis regulátoru do aplikace je stejně jako v případě PI regulátoru nutné přepočítat časovou konstantu dle rovnice 7.6. Tvar regulátoru upravený pro aplikaci Kulicka vidíme v rovnici 7.12.

$$F_{RPDLB}(p) = 0,291 \cdot (0,015 \cdot p + 1) \quad (7.12)$$

Na obrázku 7.8 vidíme frekvenční charakteristiku otevřeného obvodu vnější smyčky.



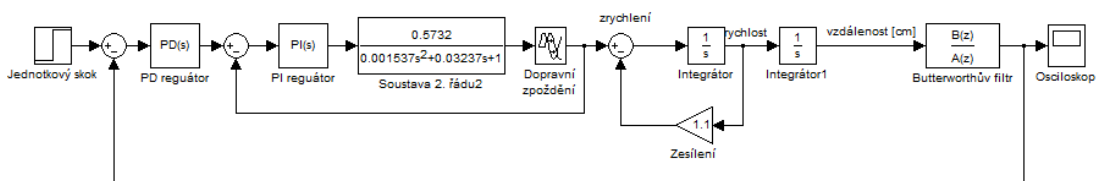
Obr. 7.8: Frekvenční charakteristika otevřeného obvodu vnější smyčky

## 8 MODEL PŘÍPRAVKU

Model jsme vytvořili v prostředí MATLAB - Simulink, slouží nám pro návrh regulátoru vnější smyčky a k jeho odzkoušení před použitím k řízení přípravku.

### 8.1 Tvorba modelu

Nejdříve jsme se pokoušeli vytvořit model dle fyzikálního popisu systému kuličky na nakloněné rovině, což se později ukázalo jako hodně složité. Z toho důvodu jsme se rozhodli pro vytvoření modelu, který vidíme na obrázku 8.1, skládajícího se ze soustavy 2. řádu vytvořené identifikací vnitřní smyčky přípravku řízené PI regulátorem, ke které jsou přidány dva integrátory které nám reprezentují změnu úhlu náklonu (zrychlení kuličky) na výslednou polohu kuličky. Pro zajištění co nejvyšší shody s reálným přípravkem jsme model doladili pomocí členu *Zesílení*, umístěného ve zpětné vazbě, jehož hodnotu jsme zjistili experimentálně. Vnější smyčkou nebo také primárním regulátorem systém je PD regulátor.



Obr. 8.1: Model přípravku vytvořený v Matlab Simulink

Soustava zapsaná v bloku *Soustava 2.řádu* je ve tvaru, který je zobrazen v rovnici (8.1) a do bloku *Dopravní zpoždění* zapíšeme hodnotu 0,0455.

$$F_S(p) = \frac{0.5732}{0.001537p^2 + 0.03237p + 1} \quad (8.1)$$

Na modelu je možné odzkoušet navržené regulátory. Pro funkčnost modelu je nutné regulátory zapsat ve správném tvaru, který je pro PI regulátor uveden v rovnici (8.2) a pro PD regulátor v rovnici (8.3).

$$F_{RPI}(p) = 0,4029 \left( 1 + \frac{1}{0,034p} \right) \quad (8.2)$$

$$F_{RPD}(p) = 6,7 \cdot (0,9p + 1) \quad (8.3)$$

## 9 SOFTWARE

Programová část semestrálního projektu je realizována v prostředí LabVIEW vyvíjeného americkou společností National Instrument. Jedná se o zkratku anglického Laboratory Virtual Instruments Engineering Workbench což v českém překladu znamená Laboratorní pracoviště virtuálních přístrojů.

### 9.1 LabVIEW

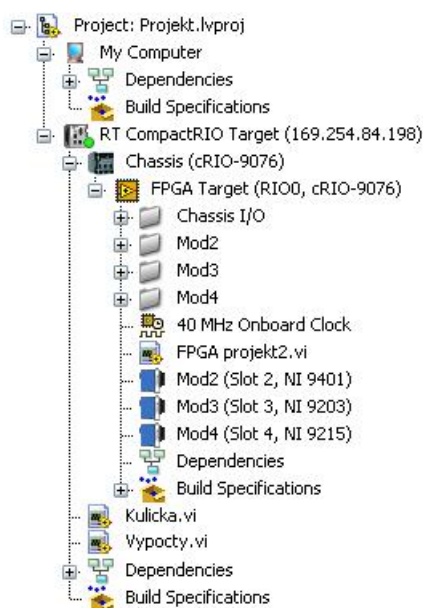
LabVIEW [11] je grafický programovací jazyk, někdy též nazývaný G-jazyk, který místo psaní textu pro vytvoření programu využívá programování pomocí bloků. Program v LabVIEW má příponu VI. Skládá se ze dvou částí, a to z čelního panelu a blokového schéma. Čelní panel slouží k zobrazení ovládacích a zobrazovacích prvků, které je možno také vkládat, mazat či jinak upravovat. Využívá se také pro řízení programu za běhu. Blokové schéma využíváme pro samotné programování programu. Pomocí panelu Function je možno vkládat různé logické nebo matematické operace, smyčky a to ať už podmínkové či časové nebo přidělovat a využívat jednotlivé vstupy a výstupy z karet připojených ke CompactRIO. Samotné programování již probíhá tzv. metodou drag-and-drop, neboli přetahováním jednotlivých bloků z nabídky do blokového schéma a následného propojení vodiči.

Princip vykonávání programu se nazývá Data flow [12]. Jedná se o princip při kterém se najednou vykonávají uzly, které mají na všech vstupech platná data. Další výhodou je paralelismus, neboli schopnost vykonávat například více smyček najednou na různých procesorech. U spojovacích vodičů rozlišujeme dle typu proměnné barvu (integer - modrá, boolean - zelená, apod) a tloušťku (2D pole, 3D pole atd).

Po úspěšné kompilaci a odzkoušení programu je následně možno vytvořit .EXE aplikaci, kterou je možné spustit bez nutnosti předchozí instalace LabVIEW na PC.

### 9.2 Konfigurace CompactRIO

Při založení nového projektu je nutné provést konfiguraci CompactRIO a programu LabVIEW. Jako první zkontrolujeme, zda máme CompactRIO připojeno na zdroj napětí a také zda jej máme propojeno s PC pomocí ethernetového kabelu. Následně přidáme šasi do nově vytvořeného projektu a v něm vytvoříme FPGA rozhraní. Do tohoto rozhraní již můžeme přidat moduly C-series, které máme vložené v šasi. Na obrázku 9.1 vidíme námi vytvořenou konfiguraci pro práci na projektu kuličky na nakloněné rovině.



Obr. 9.1: Konfigurace

## 9.3 Programování FPGA modulu

### 9.3.1 Vytvoření pulzní šířkové modulace

Tvorba PWM pro řízení servomotoru se skládá z časované smyčky s kmitočtem 40 MHz, to znamená, že perioda jednoho cyklu má délku 25 ns. Smyčka se opět spouští nebo vypíná ovladačem STOP. Konstanta  $T$  je nulovaná po každém vypnutí smyčky nebo při rovnosti s maximální hodnotou  $T_{MAX}$  a naopak inkrementována při každém proběhnutí smyčky. Výpočet konstanty  $T_{MAX}$  je vidět na rovnici 9.3, ze které vidíme, že nabývá maximální hodnoty 400 000. *Střída* je ovládací prvek, pomocí kterého zadáváme šířku pulzu, dle výpočtu z rovnice 9.4. Za  $x$  dosadíme čas v rozsahu od 600 do 2400  $\mu s$ . Tento rozsah je dán dle obrázku 4.2, kde je vidět úhel natočení odpovídající délce pulzu. Mod2/DIO0 je nultý kanál modulu NI9401, který využíváme pro vytváření pulzu o délce *Střída* s amplitudou 5 V. U všech proměnných využíváme 32 bitový integer, jelikož hodnota 400 000 zabírá 19 bitů.

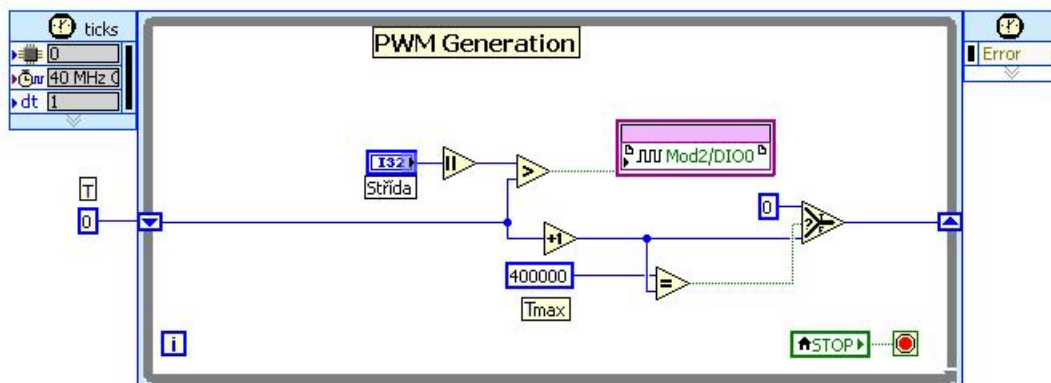
$$T_S = \frac{1}{40 \cdot 10^6} = 25 \cdot 10^{-9} s \quad (9.1)$$

$$T_M = \frac{1}{100} = 0,01 s = 10\,000 \mu s \quad (9.2)$$

$$T_{MAX} = \frac{T_M}{T_S} = \frac{0,01}{25 \cdot 10^{-9}} = 400\,000 \quad (9.3)$$

$$Střída = \frac{x}{T_M} \cdot 400\,000 = \frac{x}{10\,000} \cdot 400\,000 \quad (9.4)$$

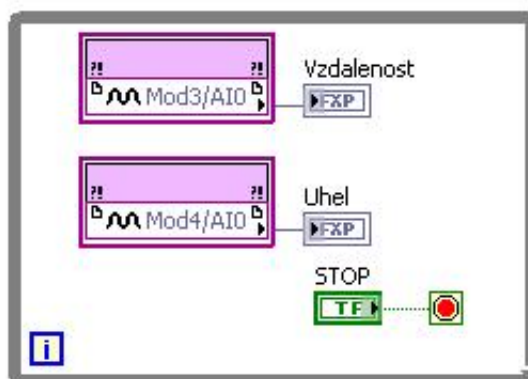
kde:  $T_S$  ... vzorkovací perioda modulu NI9401  
 $T_M$  ... vzorkovací perioda servomotoru  
 $T_{MAX}$  ... maximální perioda PWM pulzu



Obr. 9.2: Vytvoření PWM (blokové schéma)

### 9.3.2 Zpracování signálů ze senzorů

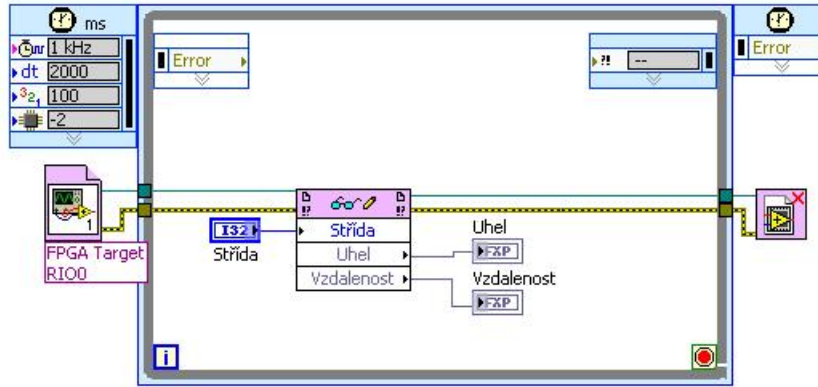
Zpracování signálů v FPGA není nijak složité, což je patrné z obrázku 9.3. Jedná se pouze o while smyčku, kterou zastavujeme ovládacím prvkem *STOP*. Ve smyčce máme nultý kanál modulu NI9203 (Mod3/AI0) a taktéž nultý kanál modulu NI9215 (Mod4/AI0), ze kterých přivádíme signál na zobrazovací prvek typu Fixed-Point pojmenovaný *Vzdalenost*, respektive *Uhel*. Zobrazovače nám neslouží pouze k zobrazení hodnot v FPGA, nýbrž také k předávání hodnot do Real-Time.



Obr. 9.3: Zpracování dat pomocí FPGA

## 9.4 Komunikace FPGA a Real-Time

Na obrázku 9.4 vidíme ukázkou komunikace. Přesněji se jedná o zápis do proměnné *Střída* a čtení dvou proměnných (*Uhel* a *Vzdalenost*), které se nachází v FPGA. Přečtené hodnoty následně zobrazujeme za pomoci indikátorů.

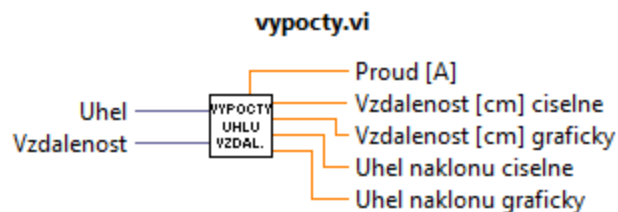


Obr. 9.4: Komunikace FPGA a Real-Time

## 9.5 Programování Real-Time modulu

### 9.5.1 Zpracování dat ze senzorů

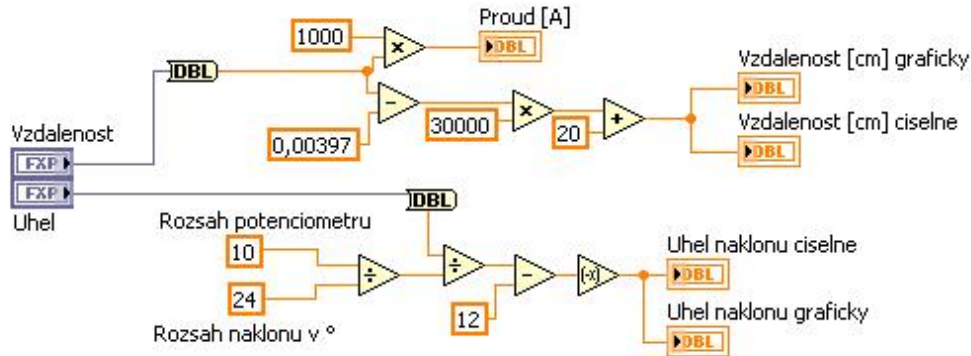
Data ze snímačů předána do Real-time zpracováváme v SubVI pojmenovaném *vypocty*, jehož ikona je zobrazena na obrázku 9.5. Na vstupy přivedeme signály ve formátu fixed-point získané pomocí FPGA, které musíme mít předem zkompilevané. Na výstupu potom dostáváme proud v ampérech, vzdálenost jako číselný údaj v centimetrech a v grafu a dále úhel ve stupních a taktéž v grafu.



Obr. 9.5: Ikona SubVI



V podprogramu již probíhají úpravy signálu, kterými jsou přepočty údajů ze snímačů na skutečnou vzdálenost kuličky v centimetrech a úhel náklonu roviny ve stupních. Blokové schéma výpočtů je vidět na obrázku 9.6.



Obr. 9.6: Zpracování dat v podprogramu vypocty.vi

Před výpočtem samotné vzdálenosti jsme museli experimentálně zjistit konstantu  $k$ , kterou jsme získali následujícím způsobem: Vytvořili jsme vzdálenosti po 20 cm, na kterých jsme odměřili proud, a z rovnice 9.6 vypočetli konstantu rovnu  $k = 30000$ , aby bylo měření co nejpřesnější v rozsahu od 0,2 do 2 metrů. Nakonec přičítáme hodnotu 20, jelikož jsme na začátku odečetli 3,97 mA, které odpovídají právě vzdálenosti 20 cm a zobrazíme pomocí indikátoru. Rovnice 9.5 již znázorňuje příklad výpočtu samotné vzdálenosti kuličky  $x$ . Ten se skládá z odečtení hodnoty 0,00397, čímž odstraníme offset. Dále ji násobíme konstantou  $k$  a přičteme hodnotu 20. Pro zobrazení proudu snímače v ampérech je třeba ji vynásobit tisícem a následně zobrazit pomocí indikátoru. Celému výpočtu předchází číslicový FIR filtr, který počítá průměr ze tří posledních hodnot.

Rovnice 9.6 znázorňuje výpočet úhlu náklonu roviny skládající se z dělení hodnoty ze snímače podílem hodnot rozsahu potenciometru (10) a hodnoty rozsahu náklonu roviny (24°). Následně odečítáme hodnotu 12, abychom dosáhli rozsahu náklonu roviny  $\pm 12^\circ$ .

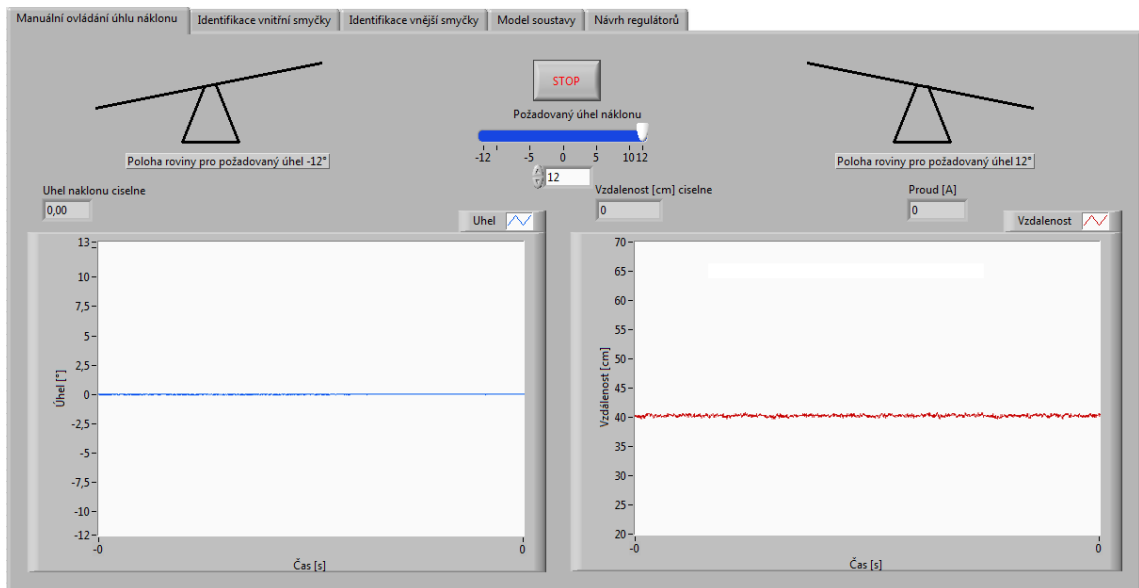
$$k = \frac{(Vzdálenost - 0,2)m}{(proud - 3,97)mA} \cdot 100 = \frac{1,78 - 0,2}{(9,256 - 3,97) \cdot 10^{-3}} \cdot 100 = 30\,000 \quad (9.5)$$

$$\begin{aligned} x &= ((Vzdálenost - 3,97)mA) \cdot k + 20 = \\ &= ((5 - 3,97) \cdot 10^{-3}) \cdot 30\,000 + 20 = 50,9\,cm \end{aligned} \quad (9.6)$$

$$\alpha = \frac{Uhel}{\frac{Rozsah\,potenciometru\,(10)}{Rozsah\,náklonu\,ve\,stupních\,(12^\circ)}} - 12\,(^\circ) = \frac{5}{\frac{10}{24}} - 12 = 0^\circ \quad (9.7)$$

## 9.5.2 Ruční ovládání náklonu

Pro případ, kdy by jsme chtěli ručně ovládat náklon roviny a zobrazit úhel náklonu případně polohu kuličky na rovině můžeme využít část nazvanou *Ruční ovládání*. Na obrázku 9.7 vidíme její čelní panel, pomocí kterého nastavujeme požadovaný úhel náklonu roviny, který program následně přepočítá na vstupní hodnoty servomotoru a zapíše do řídicí proměnné *Střída*. Dále čelní panel obsahuje zobrazovací prvky pro grafické a numerické zobrazení skutečného úhlu náklonu a vzdálenosti kuličky. Tlačítkem *STOP* zastavíme běh této části programu.

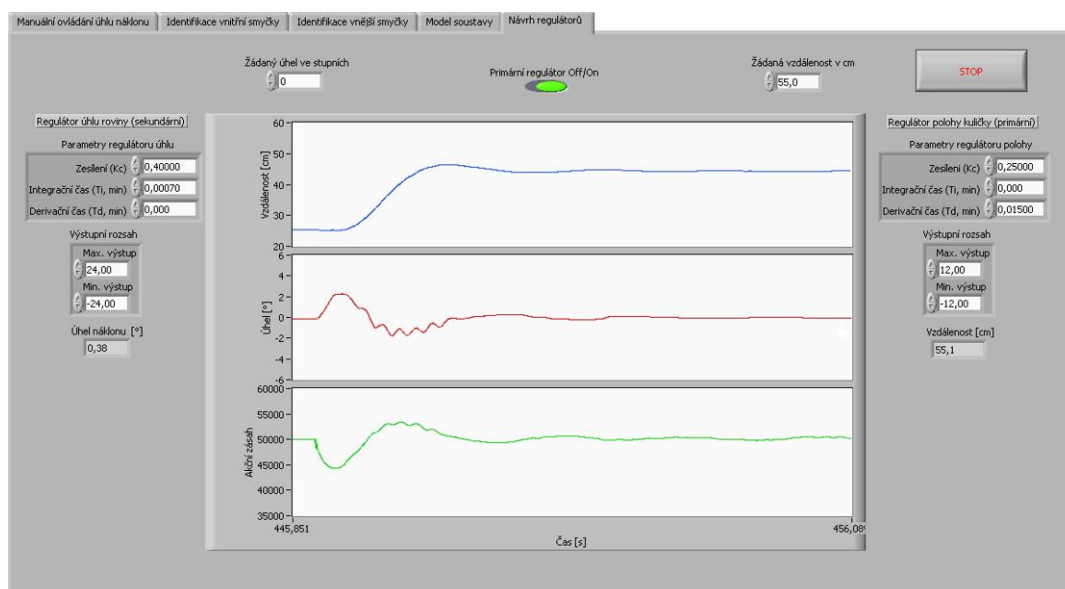


Obr. 9.7: Čelní panel programu

# 10 STUDENTSKÁ APLIKACE KULICKA

## 10.1 Uživatelské prostředí aplikace

Uživatelské prostředí, které vidíme na obrázku 10.1 se skládá z pěti záložek odpovídající jednotlivým krokům zadání laboratorní úlohy, to jest *Manuální ovládání úhlu náklonu*, *Identifikace vnitřní smyčky*, *Identifikace vnější smyčky*, *Model systému* a *Návrh regulátorů*. Dále pak každá záložka obsahuje různé ovládací či zobrazovací prvky dle potřeb. Například záložka *Návrh regulátorů* obsahuje časové průběhy vzdálenosti kuličky, úhlu náklonu a akčního zásahu regulátoru do soustavy, dále pak prvky pro nastavení parametrů obou regulátorů, prvek pro zapojení vnějšího regulátoru do regulační smyčky a dva prvky pro nastavování požadovaného úhlu a vzdálenosti.

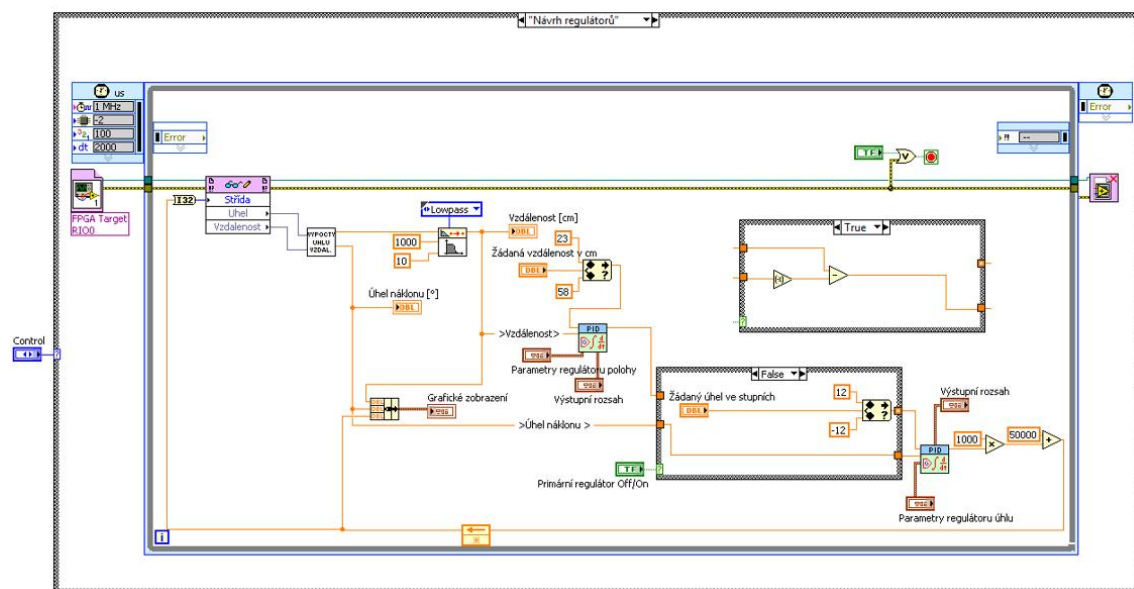


Obr. 10.1: Uživatelské prostředí aplikace Kulicka

## 10.2 Programová část

Hlavní část kódu, to jest přepínání mezi jednotlivými částmi programu plní Case struktura ovládaná prvkem *Control*, což nám zajistí, že každá část je zcela nezávislá na jakékoliv jiné. V části *Návrh regulátorů* zobrazené na obrázku 10.2 používáme program FPGA modulu popsané v kapitole 9.3 a subVI popsané v kapitole 9.5.1. Dále vidíme Case strukturu, která ovládá zapojení vnějšího regulátoru do soustavy

kaskádní regulace. Programová část *Identifikace vnitřní smyčky* je popsána v kapitole 7.1, část *Identifikace vnější smyčky* v kapitole 7.2. Záložka pojmenovaná *Model systému* se skládá pouze z vizuální části, když část programová je vytvořena v Matlab Simulink.



Obr. 10.2: Aplikace Kulicka programová část

## 11 ZÁVĚR

Cílem bakalářské práce bylo seznámení se z přípravkem kulička na nakloněné rovině a navrhnout systém řízení za pomoci platformy CompactRIO. Následně vytvořit laboratorní úlohu včetně teoretického úvodu a vzorového vypracování.

V první části práce jsme se seznámili s již existujícím přípravkem kulička na nakloněné rovině, programovatelným systémem CompactRIO a prostředím LabVIEW určeným k jeho programování.

Následně jsme realizovali náhradu snímání vzdálenosti kuličky, kdy jsme místo odporového použili laserový snímač. Ten má však minimální vzdálenost od měřeného objektu 20 cm a z tohoto důvodu bylo nutné vytvořit systém pro uchycení snímače. Po dokončení systému uchycení jsme museli na opačnou stranu roviny přidělat protizávaží pro její vyvážení. Dále jsme již připojili všechny snímače a servopohon k platformě CompactRIO a k napájecímu zdroji.

V další části práce jsme se zaměřili na zpracování signálů ze snímačů úhlu náklonu a vzdálenosti kuličky. Snímané signály jsme následně pomocí algoritmu přepočítali na skutečný úhel náklonu, respektive vzdálenost kuličky. Pro řízení natočení servomotoru jsme museli vytvořit pulzně-šířkovou modulaci. Tato část kódu je umístěná v modulu FPGA a jako vzor pro jeho tvorbu nám posloužil příklad (PWM Generation - NI9505) obsažený v samotném programu LabVIEW. Po odzkoušení a doladění pohybu roviny a snímání veličin jsme pozornost přesunuli k samotnému řízení modelu. Prvotní pokusy řízení pomocí odchylkového regulátoru ztroskotaly na velkém ovlivňování náklonu roviny pozicí kuličky. Z tohoto důvodu jsme nemohli splnit zadání v bodě 4, jelikož jedním regulátorem ať už jakéhokoliv typu není možné systém řídit. Pro řízení jsme tedy použili kaskádní regulaci, ve které bude vnitřní smyčka řízena PI regulátorem a vnější smyčka regulátorem PD. Po úspěšném experimentálním nastavení regulátoru z důvodu ověření správnosti postupu jsme se zaměřili na identifikaci systému.

Identifikaci vnitřní smyčky jsme provedli z přechodových charakteristik, kdy jsme jako vstupní signál uvažovali skokovou změnu požadovaného úhlu přepočítanou na vstupní hodnoty serva a jako výstupní signál jsme uvažovali skutečný úhel náklonu. V případě vnější smyčky jsme identifikovali soustavu uzavřené vnitřní smyčky s již navrženým regulátorem, vstupní a výstupní data zůstala stejná jako v předchozím případě. Pro vytvoření modelů soustav jsme využili identifikační nástroj programu Matlab. Pro získané modely soustav jsme metodou frekvenčních charakteristik navrhli PI regulátor pro vnitřní, respektive PD pro vnější smyčky. U PI regulátoru jsme museli přepočítat tvar zápisu regulátoru a před zápisem regulátorů do programu LabVIEW jsme museli přepočítat jejich časové konstanty dle rovnice (7.6). Výsledný tvar PI regulátoru je uveden v rovnici (7.7), tvar PD regulátoru v rovnici 7.12.

V programu Matlab Simulink jsme si vytvořili model reálného přípravku. Ten mimo jiné obsahuje identifikovaný model soustavy a PI regulátor.

Z jednotlivých částí programu jsme vytvořili aplikaci Kulicka, tak aby byla uživatelsky přijatelná a jednoduše ovladatelná. Je rozdělena do pěti záložek, z nichž každá je určena pro samostatnou úlohu a to ať už identifikaci, ruční ovládání náklonu či záložky určené k ověřování funkčnosti regulátorů.

Nakonec jsme navrhli laboratorní úlohy na téma Návrh řízení pro přípravek kulička na nakloněné rovině. Zadání úlohy včetně teoretického úvodu a vzorového vypracování je přiloženo jako příloha bakalářské práce.

Závěrem bychom chtěli podotknout, že použití snímače vzdálenosti ILR1030-8 nebylo nejšťastnější volbou. Tento snímač je určen především pro měření větších vzdáleností (až 8 m), kde rozdíl vzdálenosti 1 cm nehraje příliš velkou roli. V našem případě však měříme v vzdálenost maximálně 60 cm a proto je přesnost tohoto snímače i po filtraci nedostačující. V porovnání s původním snímáním polohy (odporový dělič) došlo ke zhoršení přesnosti měření. Z tohoto důvodu by nebylo špatné zamyslet se nad výměnou snímače za přesnější, popřípadě nad použitím úplně jiného způsobu snímání polohy kuličky, například použitím řádkové kamery umístěné nad nakloněnou rovinou.

## LITERATURA

- [1] NI cRIO-9076 [online]. [cit. 2012-12-28] Dostupné z URL:  
<<http://sine.ni.com/ds/app/doc/p/id/ds-354/lang/cs>>.
- [2] *FPGA Fundamentals* [online]. [cit. 2012-11-27]  
Dostupné z URL: <<http://www.ni.com/white-paper/6983/en>>.
- [3] *What is a Real-Time Operating System (RTOS)?* [online]. [cit. 2012-11-28]  
Dostupné z URL: <<http://www.ni.com/white-paper/3938/en#toc2>>.
- [4] NI 9401 [online]. [cit. 2012-12-28] Dostupné z URL:  
<<http://sine.ni.com/ds/app/doc/p/id/ds-86/lang/cs>>.
- [5] NI 9203 [online]. [cit. 2012-12-26] Dostupné z URL:  
<<http://sine.ni.com/ds/app/doc/p/id/ds-189/lang/cs>>.
- [6] NI 9215 [online]. [cit. 2012-12-28] Dostupné z URL:  
<<http://sine.ni.com/ds/app/doc/p/id/ds-192/lang/cs>>.
- [7] Diametral P230R51D [online]. 2012 [cit. 2013-05-02]. Dostupné z URL:  
<<http://www.diametral.cz/ac-dc-zdroje/dc-regulovatelne-zdroje/laboratorni/laboratorni-zdroj-p230r51d-2x-030v/4a-1x-5v/3a.html>>.
- [8] HS-322 [online]. [cit. 2012-12-30] Dostupné z URL:  
<[http://www.servocity.com/html/hs-322hd\\_standard\\_deluxe.html](http://www.servocity.com/html/hs-322hd_standard_deluxe.html)>.
- [9] ILR 1030-8 [online]. [cit. 2012-12-30] Dostupné z URL:  
<<http://www.micro-epsilon.de/download/manuals/man--optoNCDT-ILR-103x--de-en.pdf>>.
- [10] KLÁN, Petr. *Metody zlepšení PI regulace*. In: [online]. 2001 [cit. 2013-05-11].  
Dostupné z URL:  
<<http://www.odbornecasopisy.cz/download/au120104.pdf>>.
- [11] NATIONAL INSTRUMENT. *LabVIEW User manual*. Austin, Texas, 2003.
- [12] *Graphical Programming* [online]. [cit. 2012-11-29] Dostupné z URL:  
<<http://www.ni.com/gettingstarted/labviewbasics/dataflow.htm>>.
- [13] Blaha, P., Vavrín, P.: *Řízení a regulace I*. Elektronické skriptum VUT.

## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

CompactRIO	Kompaktní programovatelné vstupně/výstupní zařízení – Compact reconfigurable I/O
DRAM	Dynamic Random Access Memory
Compact DAQ	Samostatný systém pro sběr dat – Data acquisition systems
FPGA	Programovatelné hradlové pole – Field Programmable Gate Array
WES7	Programovatelné hradlové pole – Windows Embedded Standard 7
LUT	Vyhledávací tabulka – Lookup Table
CLB	Programovatelné Logické bloky – Configurable Logic Block
Flip-Flops	Zpoždění uzavřené smyčky – Delay Locked Loop
VI	Virtální přístroj – Virtual instrument
NIST	Národní institut standardů a technologie – National Institute of Standards and Technology



# SEZNAM PŘÍLOH

## A Laboratorní úloha

Návrh řízení pro přípravek kulička na nakloněné rovině	50
A.1 Zadání	50
A.2 Teoretický úvod	50
A.2.1 Přípravek kulička na nakloněné rovině	50
A.2.2 Regulátor	51
A.2.3 Kaskádní regulace	51
A.2.4 System identification toolbox	52
A.2.5 Návrh regulátoru pomocí frekvenčních charakteristik	53
A.2.6 Aplikace Kulicka	54

## B Vzorové vypracování

B.1 Zadání	55
B.2 Vypracování	56
B.3 Závěr	59
B.4 Obsah přiloženého CD	60

# A LABORATORNÍ ÚLOHA

## NÁVRH ŘÍZENÍ PRO PŘÍPRAVEK KULIČKA NA NAKLONĚNÉ ROVINĚ

Laboratorní úloha se zabývá identifikací a řízením dynamického systému kulička na nakloněné rovině, s využitím platformy NI CompactRIO. Pro řízení využíváme kaskádní regulaci, a to z důvodu kompenzace vlivu kuličky na náklon roviny.

### A.1 Zadání

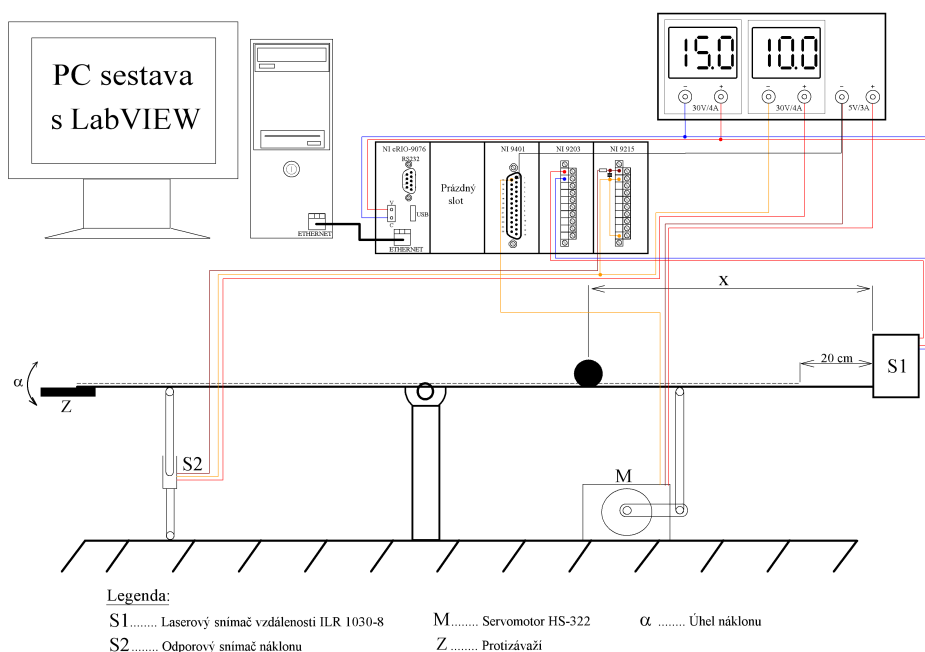
1. Seznamte se s přípravkem kuličky na nakloněné rovině, jeho modelem a aplikací *Kulicka*,
2. Proveďte identifikaci vnitřní smyčky přípravku (soustava *Požadovaný úhel náklonu – Uhel*) s využitím záložky *Identifikace vnitřní smyčky*, a nástroje Matlab System identification toolbox,
3. Návrh PI regulátoru:
  - a) Metodou frekvenčních charakteristik navrhnete PI regulátor pro vnitřní smyčku, aby byla fázová bezpečnost  $70^\circ$ , pozor na tvar regulátoru a zápis do LabVIEW,
  - b) Ověřte funkčnost právě navrženého regulátor na reálném přípravku, případně ho doladte (záložka *Návrh regulátorů*),
4. S využitím navrženého regulátoru a soustavy identifikujte vnější smyčku *Požadovaný úhel náklonu – Uhel* (zpětnovazební obvod vnitřní smyčky s již navrženým PI regulátorem) pomocí záložky *Identifikace vnější smyčky*,
5. Návrh PD regulátoru:
  - a) Před návrhem PD regulátoru přidejte k soustavě dva integrátory
  - b) Metodou frekvenčních charakteristik navrhnete PD regulátor pro vnější smyčku s fázovou  $30^\circ$ , pozor na zápis do LabVIEW upravte vhodně hodnotu zesílení viz. teoretický úvod,
  - c) Ověřte funkčnost právě navrženého regulátor na reálném přípravku, případně ho doladte (záložka *Návrh regulátorů*),

### A.2 Teoretický úvod

#### A.2.1 Přípravek kulička na nakloněné rovině

Obrázek A.1 znázorňuje schéma přípravku, včetně připojení všech jeho částí k platformě NI CompactRIO. Přípravek se skládá z roviny, kterou pohybuje pomocí

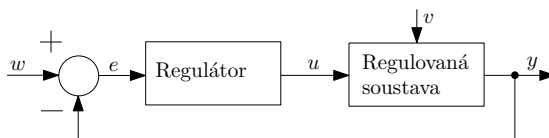
servomotoru a jejíž úhel snímáme díky odporovému snímači, polohu kuličky snímáme díky laserovému snímači vzdálenosti. Všechna data zpracováváme v PC s aplikací *Kulicka* vytvořenou v prostředí LabVIEW, pomocí které následně provádíme řízení přípravku.



Obr. A.1: Schéma přípravku kulička na nakloněné rovině

## A.2.2 Regulátor

Regulátorem nazýváme zařízení, které prostřednictvím akční veličiny působí na veličinu regulovanou tak, aby ji udržovala na námi vybrané hodnotě a regulační odchylka byla co nejmenší, nejlépe nulová. Na obrázku A.2 vidíme regulační obvod, který obsahuje regulátor, regulovanou soustavu a zápornou zpětnou vazbu.

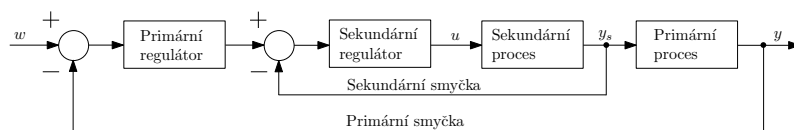


Obr. A.2: Schéma regulačního obvodu

## A.2.3 Kaskádní regulace

Někdy nazývána také jako regulace s pomocnou regulační veličinou. Na obrázku A.3 je blokové schéma kaskádní regulace. Kaskádní regulace je přínosná v přípa-

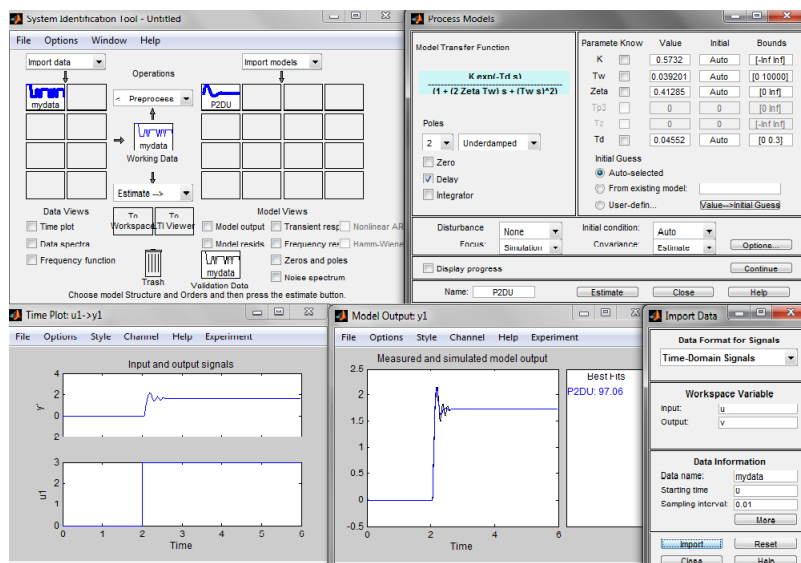
dech, kdy můžeme regulovaný proces rozdělit na dvě na sobě nezávislé části, což je zároveň základním předpokladem. Oproti regulaci pomocí odchylkového regulátoru dosáhneme kaskádní regulací výrazného zkvalitnění a zrychlení regulace.



Obr. A.3: Uspořádání smyček při kaskádní regulaci

## A.2.4 System identification toolbox

Jedná se o nástroj programu Matlab, k identifikaci soustav z naměřených vstupních a výstupních dat. Na obrázku A.4 si popíšeme použití tohoto nástroje. Nejprve si upravíme naměřené signály do tvaru s počátkem v nule a rostoucím charakterem. V záložce *Import data* nastavíme počáteční čas na hodnotu 0, vzorkovací frekvenci na hodnotu 0,01 s pro identifikaci vnitřní smyčky, 0,002 s pro vnější smyčku a dáme *Import*. Následně v záložce *Estimate* zvolíme *Process models* a vybereme si řád soustavy, zda chceme v soustavě reálné či imaginární kořeny jmenovatele či dopravní zpoždění a provedeme identifikaci tlačítkem *Estimate*. Pro zobrazení procentuálního srovnání identifikovaného modelu a reálných dat zaškrtneme *Model output*. Při dostatečně vysoké shodě modelu s reálnými daty (kolem 95%), je lepší vyzkoušet několik typů modelu, abychom pro další postup měli správně zvolený model soustavy, si jej přetáhneme do *Workspace*, kde s ním můžeme dále pracovat.



Obr. A.4: Ukázka použití System identification toolboxu

## A.2.5 Návrh regulátoru pomocí frekvenčních charakteristik

Frekvenční charakteristika je grafickým vyjádřením amplitudy  $|F(j\omega)|$  a fáze  $\varphi(\omega)$  frekvenčního přenosu systému v závislosti na frekvenci v logaritmických souřadnicích. Tato metoda návrhu spočívá se tvarování frekvenční charakteristiky otevřeného obvodu tak, abychom dosáhli co nejvhodnějšího tvaru. Tvarování provádíme v logaritmických souřadnicích. Při samotném návrhu však musíme dodržet dva požadavky, kterými jsou:

1. Co nejvyšší hodnota frekvence řezu  $\omega_{\text{ř}}$ , která určuje rychlost přechodového děje.  $\omega_{\text{ř}}$  určíme jako frekvenci, při které amplitudová charakteristika nabývá hodnoty  $0 \text{ dB}$ .
2. Co nejvyšší hodnota fázové bezpečnosti, což nám zajistí malý překmit na přechodové charakteristice. Tento požadavek se dá také vyjádřit tak, že amplitudová část charakteristiky má procházet osou  $0 \text{ dB}$  pod sklonem  $-20 \text{ dB/dek}$  a držet jej v co největším okolí  $\omega_{\text{ř}}$ .

### Posouzení stability

Pro posouzení jak daleko jsme při návrhu od nestability používáme pojmy fázová a amplitudová bezpečnost.

- a) Amplitudová bezpečnost nám říká, jakou hodnotou můžeme vynásobit aktuální hodnotu zesílení v otevřené smyčce, abychom uzavřenou smyčku přivedli na mez stability. Nejčastěji se uvádí v decibelech.
- b) Fázová bezpečnost je záporně vzatá změna fáze otevřeného obvodu, kterou přivedeme uzavřená obvod na mez stability.

### Návrh regulátoru

Nejprve vybereme vhodný typ regulátoru dle požadavků a parametrů soustavy dále pokračujeme dle čtyř níže uvedených bodů:

1. Vykreslíme frekvenční charakteristiku soustavy v logaritmických souřadnicích,
2. Napíšeme si přenos požadovaného regulátoru,
3. Dle tvaru frekvenční charakteristiky regulátoru nastavíme časovou konstantu regulátoru  $T$  dle druhého z výše uvedených požadavků,
4. Požadavek na fázovou bezpečnost  $x^\circ$  splníme nastavením správné hodnoty zesílení regulátoru,

Pro zápis regulátorů do aplikace *Kulicka* je nutné přepočítat časovou konstantu dle rovnice (A.1).

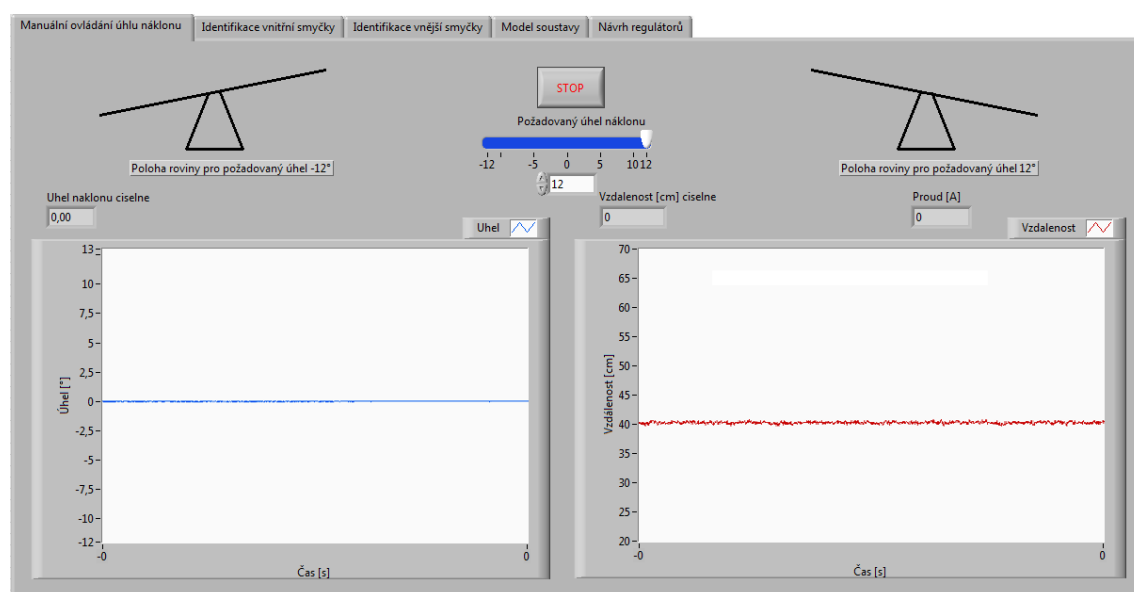
$$T_{labview} = \frac{T}{60} \quad (\text{A.1})$$

U PD regulátoru je třeba přepočítat i zesílení, a to dle vztahu (A.2), kde konstanta  $A = 23 \text{ cm/s}^{2\circ}$  reprezentuje zrychlení kuličky na jeden stupeň náklonu roviny.

$$K_c = \frac{K}{A} = \frac{6,7}{23} = 0,29 \quad (\text{A.2})$$

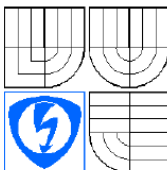
### A.2.6 Aplikace Kulicka

Uživatelské prostředí se skládá z pěti záložek odpovídající jednotlivým krokům zadání laboratorní úlohy, to jest *Manuální ovládání úhlu náklonu*, *Identifikace vnitřní smyčky*, *Identifikace vnější smyčky*, *Model soustavy* a *Návrh regulátorů*. Dále pak každá záložka obsahuje různé ovládací či zobrazovací prvky dle potřeb. Na obrázku A.5 vidíme záložku *Manuální ovládání úhlu náklonu*, pomocí které můžeme ručně nastavovat úhel náklonu roviny a sledovat v grafech skutečný úhel a polohu kuličky.



Obr. A.5: Uživatelské prostředí aplikace Kulicka

## B VZOROVÉ VYPRACOVÁNÍ

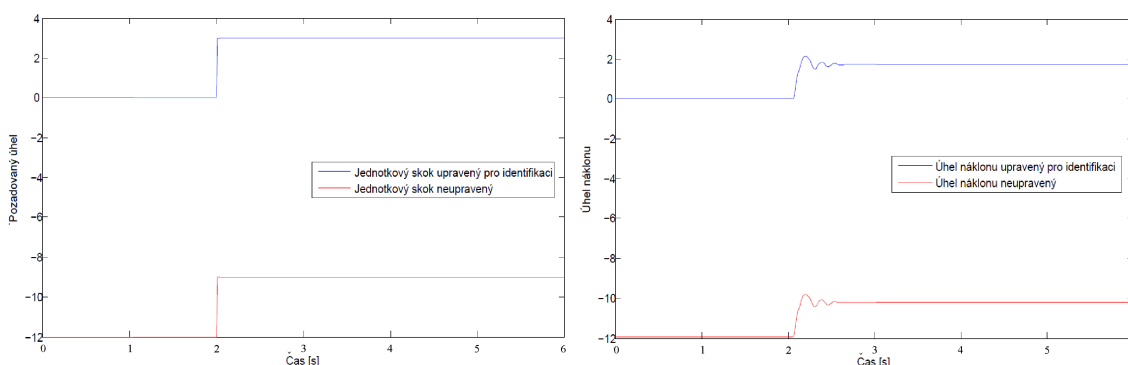
 <b>VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ</b>	Předmět	
	Jméno <b>Martin Řezáč</b>	
	Ročník	Studijní skupina
	Spolupracovali:	Měřeno dne
Kontroloval	Hodnocení	Dne
Číslo úlohy	Název úlohy <b>Návrh řízení pro přípravku kulička na nakloněné rovině</b>	

### B.1 Zadání

1. Seznamte se s přípravkem kuličky na nakloněné rovině, jeho modelem a aplikací *Kulicka*,
2. Proveďte identifikaci vnitřní smyčky přípravku (soustava *Požadovaný úhel náklonu – Uhel*) s využitím záložky *Identifikace vnitřní smyčky*, a nástroje Matlab System identification toolbox,
3. Návrh PI regulátoru:
  - a) Metodou frekvenčních charakteristik navrhnete PI regulátor pro vnitřní smyčku, aby byla fázová bezpečnost  $70^\circ$ , pozor na tvar regulátoru a zápis do LabVIEW,
  - b) Ověřte funkčnost právě navrženého regulátoru na reálném přípravku, případně ho doladte (záložka *Návrh regulátorů*),
4. S využitím navrženého regulátoru a soustavy identifikujte vnější smyčku *Požadovaný úhel náklonu – Uhel* (zpětnovazební obvod vnitřní smyčky s již navrženým PI regulátorem) pomocí záložky *Identifikace vnější smyčky*,
5. Návrh PD regulátoru:
  - a) Před návrhem PD regulátoru přidejte k soustavě dva integrátory
  - b) Metodou frekvenčních charakteristik navrhnete PD regulátor pro vnější smyčku s fázovou  $30^\circ$ , pozor na zápis do LabVIEW upravte vhodně hodnotu zesílení viz. teoretický úvod,
  - c) Ověřte funkčnost právě navrženého regulátoru na reálném přípravku, případně ho doladte (záložka *Návrh regulátorů*),

## B.2 Vypracování

Nejprve jsem se seznámil s přípravkem s aplikací Kulka prostřednictvím záložky *Manuální ovládání úhlu náklonu*. Následně jsme se pustili do vypracování úlohy. Pro naměření vstupně/výstupních signálů k identifikaci vnitřní smyčky jsme využili záložku *Identifikace vnitřní smyčky*, která automaticky spustí měření vstupního a výstupního signálu a následně tyto průběhy vyexportuje do programu Excel. V tomto programu jsme data pouze uložili do formátu .xls, který je vhodný pro přesunutí do programu Matlab. Pro využití těchto průběhů k identifikaci je třeba je nejprve upravit do správného tvaru (počátek v hodnotě 0 a rostoucí průběh). Vstupní průběh jsme upravili příkazem  $u = (\text{skok}(1 : \text{end}, 2) + 12)$ , výstupní pak příkazem  $v = (\text{uhel}(1 : \text{end}, 2) + 11, 93)$ , vliv těchto úprav vidíme na obrázku B.1.



Obr. B.1: Úprava signálů pro identifikaci

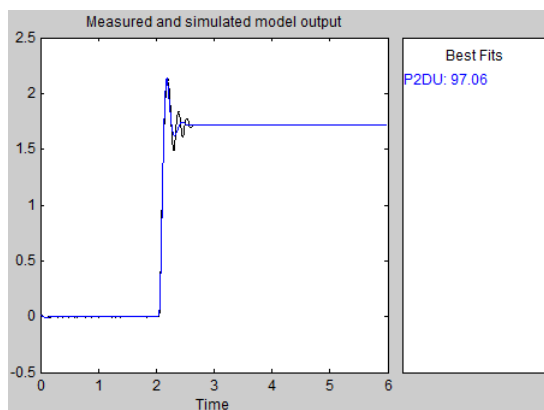
Takto upravené průběhy jsme použili k identifikaci pomocí System identification toolbox s periodou vzorkování 0,01 s. Zde jsme použitím modelu soustavy 2. řádu s komplexními kořeny a dopravním zpožděním dosáhli shody 97,06% s naměřenými průběhy (viz obrázek B.2). Přenos výsledného modelu soustavy je uveden v rovnici B.1

$$F_S(p) = \frac{0,5732}{0,001537p^2 + 0,03237p + 1} \cdot e^{-0.0455p} \quad (\text{B.1})$$

Na identifikovanou soustavu jsme následně dle požadavků uvedených v teoretickém úvodu a fázovou bezpečnost  $70^\circ$  navrhli metodou frekvenčních charakteristik PI regulátor. Přenos tohoto regulátoru je uveden v rovnici (B.2).

$$F_{RPI}(p) = 11,85 \cdot \frac{0,034 \cdot p + 1}{p} \quad (\text{B.2})$$





Obr. B.2: Shoda naměřené a identifikované odezvy

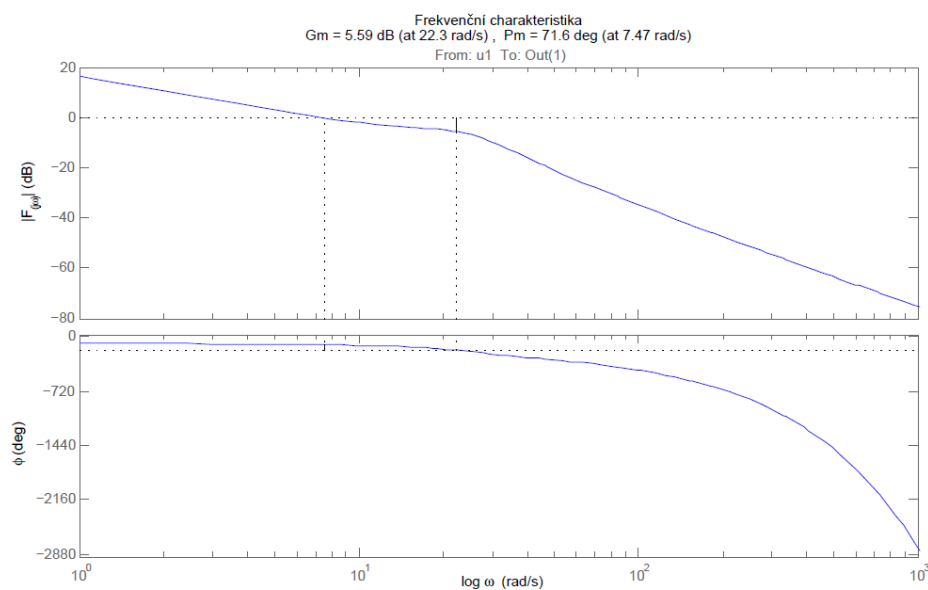
Pro zadání regulátoru do aplikace *Kulicka* bylo nutné přepočítat tvar regulátoru dle rovnice (B.3) a časovou konstantu  $T_i$  dle rovnice (B.4). Upravený přenos má tvar uvedený v rovnici (B.5).

$$F_{RPI}(p) = 11,85 \cdot \frac{0,034p + 1}{p} = 11,85 \cdot 0,034 \cdot \left(1 + \frac{1}{0,034p}\right) \quad (\text{B.3})$$

$$T_i \text{ labview} = \frac{T_i}{60} = \frac{0,034}{60} = 0,00057 \quad (\text{B.4})$$

$$T_{RPILB}(p) = 0,4029 \cdot \left(1 + \frac{1}{0,00057p}\right) \quad (\text{B.5})$$

Na obrázku B.3 si můžeme prohlédnout frekvenční charakteristiku otevřeného obvodu vnitřní smyčky.



Obr. B.3: Frekvenční charakteristika otevřeného obvodu vnitřní smyčky

Následně jsme pomocí záložky *Identifikace vnější smyčky* a podle návodu na čelním panelu aplikace provedli identifikaci vnější smyčky. Data jsme vyexportovali přes Excel do Matlabu, kde jsem pomocí System identification toolbox s periodou vzorkování  $0,002\text{ s}$  provedli návrh modelu soustavy, který je 2. řádu s reálnými kořeny (rovnice B.6) a s naměřenými daty se shoduje z  $95,43\%$ . Následně jsme dle zadání k modelu přidali dva integrátory. Na tuto soustavu jsme navrhli metodu frekvenčních charakteristik PD regulátoru S fázovou bezpečností  $30^\circ$ .

$$F_S(p) = \frac{1,004}{0.007356p^2 + 0.1715p + 1} \quad (\text{B.6})$$

Výsledný přenos navrženého regulátoru vidíme v rovnici (B.7).

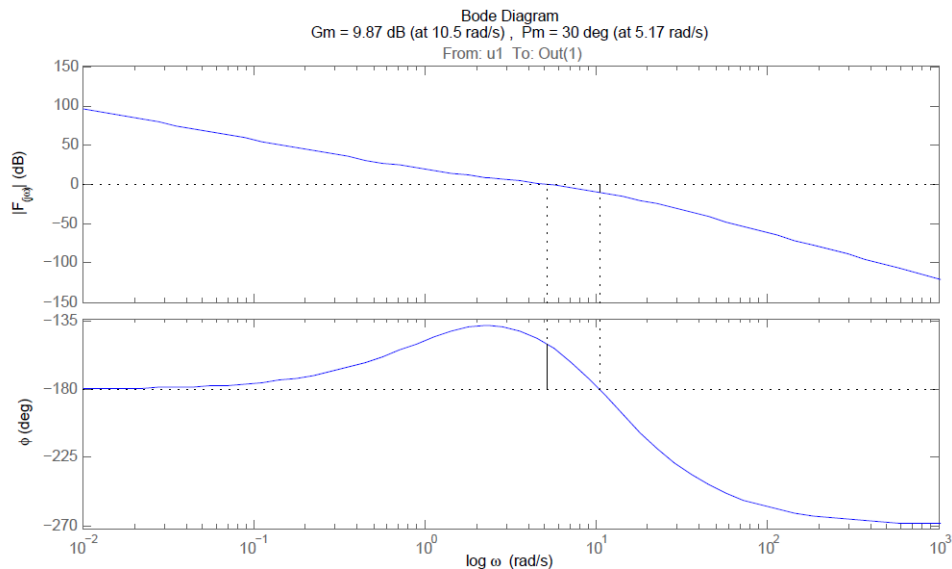
$$F_{RPD}(p) = 6,7 \cdot (0,9 \cdot p + 1) \quad (\text{B.7})$$

Pro zápis do aplikace je opět nutné přepočítat časovou konstantu dle rovnice B.4. V tomto případě je třeba přepočítat i zesílení regulátoru, a to podle vztahu (B.8), kde konstantu  $A$  máme zadanou do tvaru. Výsledná přenos regulátoru vidíme v rovnici B.9.

$$K_c = \frac{K}{A} = \frac{6,7}{23} = 0,29 \quad (\text{B.8})$$

$$F_{RPD}(p) = 0,29 \cdot (0,015 \cdot p + 1) \quad (\text{B.9})$$

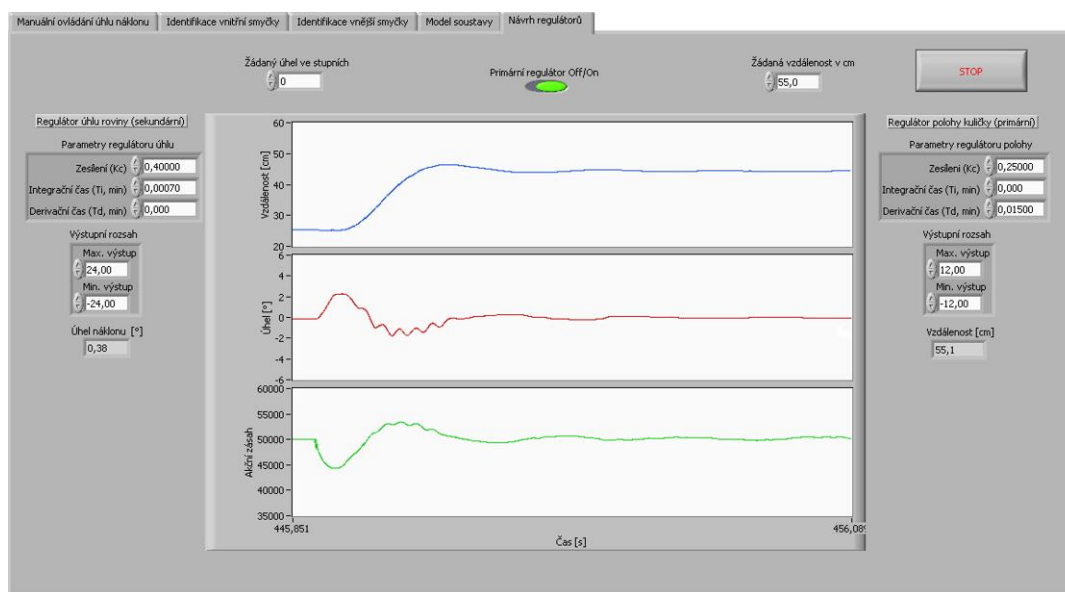
Na obrázku B.4 vidíme frekvenční charakteristiku otevřeného obvodu vnější smyčky.



Obr. B.4: Frekvenční charakteristika otevřeného obvodu vnější smyčky

Pomocí záložky *Návrh regulátorů* jsme ověřili funkčnost PD regulátoru. Pro regulaci reálného přípravku ho není třeba nijak upravovat. Na obrázku B.5 vidíme

časový průběh odezvy na skok požadované vzdálenosti kuličky z 25 cm na 55 cm, což je takřka pře celý rozsah roviny.



Obr. B.5: Odezva na skok požadované veličiny

### B.3 Závěr

Po seznámení se s přípravkem a modelem vytvořeného v Matlab Simulink jsme začali s identifikací vnitřní smyčky. Tu jsme provedli pomocí záložky *Identifikace vnitřní smyčky* a následného přesunutí dat přes program Excel do prostředí Matlab. Pro vytvoření modelu soustavy jsme využili System identification toolbox. Pro model soustavu 2. řádu s komplexními kořeny a zpožděním jsme dosáhli shody 97.06% s naměřenými daty. Přenos identifikované soustavy je uveden v rovnici (B.1). Pro tuto soustavu jsme metodou frekvenčních charakteristik navrhli dle zadaných požadavků PI regulátor, jehož přenos upravený pro zápis do aplikace Kulicka je uveden v rovnici (B.5). Pomocí záložky *Identifikace vnější smyčky* jsme provedli identifikaci uzavřeného obvodu vnitřní smyčky s již navrženým PI regulátorem. Data jsme si opět přesunuli do Matlabu a provedli vytvoření modelu soustavy. Model soustavu 2. řádu s reálnými kořeny jsme dosáhli shody 95,43% s naměřenými daty. Přenos identifikované soustavy je uveden v rovnici (B.6). Model soustavy jsme rozšířili o dva póly v počátku a metodou frekvenčních charakteristik navrhli PD regulátor. Přepočítali jsme časovou konstantu regulátoru a zesílení dle vztahu (B.8). Výsledný přenos regulátoru je uveden v rovnici (B.9). Navržené regulátory jsme následně odzkoušeli pomocí záložky *Návrh regulátorů*.

## B.4 Obsah přiloženého CD

Labview:

- Projekt Kulicka.lvproj obsahující aplikaci Kulicka

Matlab:

- Filtr.mat obsahující koeficienty Butterworthova filtru
- Kulicka\_model.mdl s vytvořeným modelem přípravku kulička na nakloněné rovině

Dokumentace:

- Elektronická verze bakalářská práce na téma Laboratorní úlohy pro přípravek s kuličkou na nakloněné rovině