

UNIVERZITA PARDUBICE
Fakulta elektrotechniky a informatiky

ŘÍZENÍ MODULÁRNÍHO MECHANICKÉHO SYSTÉMU

Filipp Poplavko

Diplomová práce

2023

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2019/2020

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Filipp Poplavko**
Osobní číslo: **I18231**
Studijní program: **N2612 Elektrotechnika a informatika**
Studijní obor: **Řízení procesů**
Téma práce: **Řízení modulárního mechanického systému**
Zadávací katedra: **Katedra řízení procesů**

Zásady pro vypracování

Cíl: Cílem práce je navrhnout a aplikovat zvolený řídicí algoritmus mechanického systému – modulární soustavy Quanser QNET QNET Physics and Dynamics osazené v základně pro měření a ovládání NI ELVIS II+.

Teoretická část: Rešerše problematiky týkající se programování v grafickém programovacím jazyce G a tvorby aplikací v LabVIEW s využitím Control Design and Simulation Module. Návrh regulátoru zvolené struktury a nalezení jeho optimálních parametrů.

Implementační část: Tvorba simulačních schémat a souvisejících aplikací pro měření a řízení modulárního mechanického systému v grafickém programovacím jazyce G a ověření jejich činnosti.

Rozsah pracovní zprávy: **cca 50 stran**
Rozsah grafických prací:
Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

VLACH, J.; HAVLÍČEK, J.; VLACH, M. 2008. Začínáme s LabVIEW. Praha: BEN – technická literatura. 247 s. ISBN 978-80-7300-245-9.
PECHOUŠEK, J. 2004. Základy programování v prostředí LabVIEW. Olomouc: Univerzita Palackého v Olomouci. 87 s. ISBN 80-244-0800-7.
DORF, R. C.; BISHOP, R. H. 2005. Modern Control Systems. 10 vyd. New Jersey: Prentice Hall. 881 s. ISBN 0-13-127765-0.

Vedoucí diplomové práce: **Ing. Libor Kupka, Ph.D.**
Katedra řízení procesů

Datum zadání diplomové práce: **7. listopadu 2019**
Termín odevzdání diplomové práce: **15. května 2020**

L.S.

Ing. Zdeněk Němec, Ph.D. v.r.
děkan

Ing. Daniel Honc, Ph.D. v.r.
vedoucí katedry

Prohlášení

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 24. 05. 2023

Filipp Poplavko

Poděkování

Na tomto místě bych rád poděkoval vedoucímu práce panu Ing. Liboru Kupkovi, Ph.D., za pomoc a konzultace při vedení celé práce. Dále bych rád poděkoval i všem učitelům, u kterých jsem po celou dobu studia na vysoké škole studoval, za nabyté znalosti a dovednosti.

V Pardubicích dne 24. 05. 2023

Filipp Poplavko

ANOTACE

Tato diplomová práce se zabývá realizací aplikací pro řízení, měření a ovládání modulární mechanické soustavy v prostředí LabVIEW s využitím simulačních nástrojů Control Design and Simulation Module. Na základě matematického modelu mechanické soustavy je navržen regulátor PID. Činnost aplikace je otestována s navrženým PID regulátorem a je provedeno měření.

KLÍČOVÁ SLOVA

LabVIEW, Control Design and Simulation Module, modulární mechatronický systém, řízení, regulátor.

TITLE

Control of modular mechatronic system.

ANNOTATION

This diploma thesis deals with the implementation of applications for control, measurement and control of a modular mechatronic system in the LabVIEW environment using the Control Design and Simulation Module simulation tools. Based on the mathematical model of the mechanical system, a PID controller is designed. The operation of the application is tested with the designed PID controller and a measurement is made.

KEYWORDS

LabVIEW, Control Design and Simulation Module, modular mechatronic system, control, controller.

OBSAH

	SEZNAM ZKRATEK A ZNAČEK	12
	SEZNAM SYMBOLŮ VELIČIN A FUNKCÍ.....	13
	SEZNAM ILUSTRACÍ	14
	SEZNAM TABULEK.....	17
	ÚVOD	18
1	LABVIEW	19
1.1	Filozofie LabVIEW.....	19
1.1.1	Grafické programování	19
1.1.2	Datový tok.....	19
1.1.3	Datové typy	20
1.1.4	Typy proměnných, operace s clusteru a porovnání s datovým typem array.....	21
1.1.5	Uživatelské rozhraní	22
1.1.6	Čelní panel	23
1.1.7	Blokový Diagram.....	23
1.1.8	Typy bloků	24
1.2	Control Design And Simulation Module	25
1.2.1	Knihovna Simulation	25
1.2.2	Knihovna Control Design	26
1.2.3	Knihovna System Identification.....	27
1.2.4	Knihovna PID	27
1.2.5	Knihovna Fuzzy Logic.....	28
1.2.6	Control & Simulation Loop	29
2	ZAŘÍZENÍ A PŘÍSTROJE	30
2.1	NI ELVIS II+	30
2.2	Quanser QNET Physics And Dynamics Board.....	31
2.2.1	Výuka základních principů fyziky a dynamiky	31
2.2.2	Popis práce QNET Physics and Dynamics Board	32
2.2.3	Hardwarové komponenty	33
2.2.4	Round Electromagnet EM075-6-222	34
2.2.5	Enkodér E8P-512-250-S-H-M-B	35
2.3	QNET 2.0 VTOL Board.....	35
2.3.1	Stručný popis nového zařízení	36
3	POPIS ZÁKLADNÍCH VI A SUBVI	37
4	TEORETICKÝ POPIS REGULÁTORU A REGULACE	41

4.1	PID regulátor.....	41
4.1.1	Popis PID regulátoru.....	41
4.1.2	Komponenty PID regulátoru.....	42
4.2	Metody pro nalezení optimálních parametrů regulátoru.....	43
4.2.1	Integrovaná kritéria kvality	43
5	METODY NASTAVENÍ PID	45
5.1	Experimentální metody kritických parametrů.....	45
5.1.1	Zieglerova-Nicholsova metoda kritických parametrů.....	45
5.1.2	Experimentální Tyreusova-Luybenova metoda kritických parametrů.....	46
5.2	Analytické metody	46
5.2.1	Metoda Cohen-Coon (výpočet pro aproximační přenos).....	46
5.2.2	Metoda Požadovaného Modelu (metoda inverze dynamiky)	47
6	PRAKTICKÁ ČÁST.....	48
6.1	Použité datové typy	48
6.2	Popis hlavního VI systému P&D	49
6.3	Popis konstrukce a činnosti zařízení	55
6.3.1	Volba modulu a fyzikálního experimentu pro řízení	56
6.3.2	Možné způsoby regulace a problémy při implementaci regulace.....	57
6.3.3	Fyzikální specifika chování zařízení.....	57
6.3.4	Rozhodnutí použít k ovládní jiné zařízení.....	58
7	DOPLŇKOVÉ A REGULAČNÍ VI.....	59
7.1	Ukládání dat	59
7.1.1	Část pro zpracování dat.....	60
7.1.2	Část pro ukládání dat.....	61
7.2	PID Regulátor	62
7.3	Saturace	63
7.3.1	Anti-windup modifikace	63
7.3.2	Problémy použití anti-windup modifikace.....	64
7.4	Konečné schéma PID regulátoru a ovládní	65
7.5	Finální verze předního panelu	67
8	EXPERIMENTÁLNÍ IDENTIFIKACE SOUSTAVY.....	69
8.1	Posloupnost práce a měření dat.....	69
8.2	Přechodová charakteristika	69
8.3	Statická charakteristika	70
8.4	Identifikace.....	71
8.4.1	Skript a zpracování dat v MATLABu.....	71
8.4.2	Provádění aproximace.....	73

9	VYPOCET PARAMETRU PID	78
9.1	Metoda ZN určení pomocí relé ve zpětné vazbě.....	78
9.2	Experimentální Tyreusova-Luybenova metoda kritických parametrů.....	80
9.3	Metoda Cohen-Coon	81
9.4	Metoda Požadovaného Modelu.....	82
9.5	Výsledné parametry	84
9.6	Provádění simulací	84
10	REGULACE REÁLNÉHO SYSTÉMU	89
10.1	Úprava podle vypočtených parametrů	89
10.1.1	Metoda ZN	89
10.1.2	Metoda TL.....	89
10.1.3	Metoda CC	90
10.1.4	Metoda Požadovaného Modelu.....	90
10.2	Ruční pokročilé nastavení parametrů.....	92
10.2.1	Obecný popis nastavení	92
10.2.2	Parametry regulátoru a odpovídající dynamika	94
	ZÁVĚR	102
	POUZITÁ LITERATURA	103
	PŘÍLOHY	105

SEZNAM ZKRATEK A ZNAČEK

CC	Cohen-Coon (metoda)
D	Derivační (složka)
ELVIS	Engineering Laboratory Virtual Instrumentation Suite
I	Integrační (složka)
LabVIEW	Laboratory Virtual Instruments Engineering Workbench
MATLAB	Matrix Laboratory
MIMO	System Multiple Input Multiple Output – systém s více vstupy a více výstupy
NI	National Instruments
P	Proporcionální (regulátor)
P&D	Physics and Dynamics
PC	Personal computer – osobní počítač
PD	Proporcionálně-derivační (regulátor)
PI	Proporcionálně-integrační (regulátor)
PID	Proporcionálně-integračně-derivační (regulátor)
QNET	Quanser NI Engineering Trainers
SISO	System Single Input Single Output – systém s jedním vstupem a jedním výstupem
URO	Uzavřený regulační obvod
USB	Universal Serial Bus
VI	Virtual Instruments
VTOL	Vertical Take-off and Landing
ZN	Ziegler–Nichols (metoda)

SEZNAM SYMBOLŮ VELIČIN A FUNKCÍ

D	dopravní zpoždění
$e(t)$	regulační odchylka
e_{\max}	maximální regulační odchylka
E_r	chyba aproximace
R	reléová konstanta
r_0	zesílení regulátoru
r_{0k}	kritické zesílení
t	čas, s
T	doba regulace
T_d	derivační časová konstanta, s
T_f	anti-windup koeficient
T_i	integrační časová konstanta, s
T_k	kritická perioda
U	elektrické napětí, v
$u(t)$	akční veličina
$w(t)$	žádaná hodnota
x	nezávisle proměnná
y	závisle proměnná
$y(t)$	regulovaná veličina
Z	zesílení
κ	relativní překmit

SEZNAM ILUSTRACÍ

Obrázek 1.1 – Typy dat (National Instruments, 2022)	20
Obrázek 1.2 – Čelní panel.....	23
Obrázek 1.3 – Blokový diagram (VI)	24
Obrázek 1.4 – Podknižovny modulu Control and Simulation.....	25
Obrázek 1.5 – Podknižovna Simulation	26
Obrázek 1.6 – Podknižovna Control Design	26
Obrázek 1.7 – Podknižovna System Identification	27
Obrázek 1.8 – Podknižovna PID	28
Obrázek 1.9 – Podknižovna Fuzzy Logic.....	28
Obrázek 1.10 – Control & Simulation Loop.....	29
Obrázek 2.1 – Komponenty NI ELVIS II+ (Quanser, 2015).....	30
Obrázek 2.2 – Qanser QNET Physics and Dynamics Board (Quanser, 2015).....	31
Obrázek 2.3 – Interakce mezi komponentami QNET Physics a Dynamics (Quanser, 2015) ..	32
Obrázek 2.4 – Obecné rozložení QNET Physics and Dynamics (Quanser, 2015).....	33
Obrázek 2.5 – Elektromagnet	35
Obrázek 2.6 – QNET 2.0 VTOL Board (Quanser, 2014).....	35
Obrázek 3.1 – Čelní panel systému QNET Physics and Dynamics Quick Start VI.....	38
Obrázek 3.2 – VI systému QNET Physics and Dynamics Quick Start VI.....	38
Obrázek 3.3 – Čelní panel subVI DIO.....	38
Obrázek 3.4 – VI subsystému DIO	39
Obrázek 3.5 – Čelní panel subVI Encoder Config	39
Obrázek 3.6 – VI subsystému Encoder Config.....	39
Obrázek 3.7 – Ukázka čelního panelu systému QNET Physics and Dynamics	40
Obrázek 4.1 – Zapojení URO s regulátorem ve zpětné vazbě.....	41
Obrázek 4.2 – Schéma PID regulátoru	41
Obrázek 6.1 – Inicializační část programu	50
Obrázek 6.2 – SubVI inicializace enkodéru	51
Obrázek 6.3 – SubVI inicializace řízení elektromagnetů	52
Obrázek 6.4 – Prováděcí část programu	53
Obrázek 6.5 – Závěrečná část programu	55
Obrázek 7.1 – Systém ukládání dat	59

Obrázek 7.2 – Blokové schéma subVI pro zpracování dat.....	60
Obrázek 7.3 – Cykly ukládání dat	61
Obrázek 7.4 – PID regulátor s beznárazovým přepínáním.....	62
Obrázek 7.5 – Saturace	63
Obrázek 7.6 – Doplnění anti-windup modifikace PID regulátoru.....	64
Obrázek 7.7 – Finální schéma PID regulátoru.....	65
Obrázek 7.8 – Finální schéma subVI PID	66
Obrázek 7.9 – Finální čelní panel subVI PID.....	66
Obrázek 7.10 – Finální čelní panel systému Quanser QNET 2.0 VTOL Board.....	67
Obrázek 8.1 – Přejížděvací charakteristika	69
Obrázek 8.2 – Statická charakteristika	70
Obrázek 8.3 – Statická charakteristika systému a přibližná linearizace	70
Obrázek 8.4 – Aproximace modelem 1. řádu	74
Obrázek 8.5 – Aproximace modelem 2. řádu	74
Obrázek 8.6 – Aproximace modelem 3. řádu	75
Obrázek 8.7 – Aproximace modelem 4. řádu	75
Obrázek 8.8 – Aproximace modelem 5. řádu	76
Obrázek 9.1 – Relé ve zpětné vazbě	78
Obrázek 9.2 – Kritické kmity s relé ve zpětné vazbě	79
Obrázek 9.3 – Aproximace modelem 1. řádu s dopravním zpožděním.....	82
Obrázek 9.4 – Aproximační model (tlumená kmitavá soustava s dopravním zpožděním D) ..	83
Obrázek 9.5 – Simulační obvod v Simulinku	85
Obrázek 9.6 – Simulace podle metody ZN.....	85
Obrázek 9.7 – Simulace podle metody TL	86
Obrázek 9.8 – Simulace podle metody CC.....	86
Obrázek 9.9 – Simulace podle metody MPM překmit 0%.....	87
Obrázek 9.10 – Simulace podle metody MPM překmit 10%.....	87
Obrázek 9.11 – Simulace podle metody MPM překmit 20%.....	88
Obrázek 10.1 – Chování reálného systému s laděním regulátoru metodou ZN.....	89
Obrázek 10.2 – Chování reálného systému s laděním regulátoru metodou TL.....	90
Obrázek 10.3 – Chování reálného systému s laděním regulátoru metodou CC	90
Obrázek 10.4 – Chování reálného systému s laděním regulátoru MPM (překmit 0%).....	91
Obrázek 10.5 – Chování reálného systému s laděním regulátoru MPM (překmit 10%).....	91
Obrázek 10.6 – Chování reálného systému s laděním regulátoru MPM (překmit 20%).....	92

Obrázek 10.7 – Chování reálné soustavy při ručním ladění, I složka = 2	94
Obrázek 10.8 – Chování reálné soustavy při ručním ladění, I složka = 1,5	95
Obrázek 10.9 – Chování reálné soustavy při ručním ladění, I složka = 0,9	95
Obrázek 10.10 – Chování reálné soustavy při ručním ladění, D složka = 0,8.....	96
Obrázek 10.11 – Chování reálné soustavy při ručním ladění, D složka = 1,3.....	96
Obrázek 10.12 – Chování reálné soustavy při ručním ladění, D složka = 1,8.....	97
Obrázek 10.13 – Chování reálné soustavy při ručním ladění, D složka = 3.....	97
Obrázek 10.14 – Chování reálné soustavy při ručním ladění, D složka = 5.....	98
Obrázek 10.15 – Chování reálné soustavy při ručním ladění, verze 1: I=0,3 D=1,4	98
Obrázek 10.16 – Chování reálné soustavy při ručním ladění, verze 2: I=0,3 D=1,8	99

SEZNAM TABULEK

Tabulka 2.1 – Komponenty ELVIS II+ (Quanser, 2015)	31
Tabulka 2.2 – Nomenklatura komponent QNET Physics and Dynamics (Quanser, 2015).....	33
Tabulka 5.1 – Stavitelné parametry PID regulátoru pro ZN metodu kritických parametrů	45
Tabulka 5.2 – Stavitelné parametry analogových regulátorů pro Tyreusovu-Luybenovu metodu kritických parametrů	46
Tabulka 5.3 – Relativní překmit κ a jemu odpovídající volitelný parametr α pro MPM"	47
Tabulka 8.1 – Výsledky aproximace	76
Tabulka 9.1 – Vypočítané parametry regulátoru metodou ZN	80
Tabulka 9.2 – Vypočítané parametry regulátoru	84
Tabulka 9.3 – Vypočítané parametry regulátoru a integrální kritéria kvality	88
Tabulka 10.1 – Ruční doladění, I složka	99
Tabulka 10.2 – Ruční doladění, D složka, dopad na překmit	100
Tabulka 10.3 – Ruční doladění, D složka, vliv na hladkost změny $y(t)$	100
Tabulka 10.4 – Konečné doporučené parametry	100

ÚVOD

Cílem této práce je navrhnout a vytvořit řídicí systém pro modulární mechatronický systém Quanser QNET Physics and Dynamics Board v prostředí LabVIEW.

Laboratorní vybavení Quanser QNET Physics and Dynamics Board osazené v základně pro měření a ovládání NI ELVIS II+ je vzdělávací model reálných systémů, určený pro výukové a experimentální účely, který umožňuje studentům seznámit se s provozem a ovládáním skutečného pracovního zařízení.

Laboratorní zařízení Quanser jsou navržena speciálně pro práci na bázi NI ELVIS II + v prostředí LabVIEW společnosti National Instruments, která vyvíjí software pro automatizované simulační, testovací a analytické systémy.

Laboratorní soustava je řízena pomocí aplikace vytvořené v LabVIEW s využitím grafického programovacího jazyka G a s využitím modulu Control Design and Simulation Module. Aplikace je kromě funkcí přímého řízení doplněna také o systém ukládání dat a vybraný regulátor typu PID.

Pro úpravu parametrů regulátoru byly použity aproximační metody v prostředí MATLAB a simulace byla provedena také v prostředí Simulink. Pro řízení laboratorního zařízení jsou použity parametry regulátoru vypočítané pro řízení reálného systému, dále je prováděn další výzkum fungování systému a hledání různých možností kombinací parametrů regulátoru pro požadované optimální řízení systému.

1 LABVIEW

1.1 FILOZOFIE LABVIEW

Laboratory Virtual Instrument Engineering Workbench zkráceně LabVIEW, také nazývané jazyk G, je grafické vývojové prostředí od firmy National Instruments, která je hlavním představitelem v oblasti virtuální instrumentace. LabVIEW umožňuje vyvíjet aplikace určené k měření, sběru dat, řízení přístrojů, vizualizaci, ale dále také nachází uplatnění i v průmyslové automatizaci.

Cílem výše uvedené virtuální instrumentace je nahradit fyzické zařízení jeho softwarovou podobou, tento pojem se promítá i do aplikací, s kterými pracuje prostředí LabVIEW a to s virtuálními instrumenty tzn. VI. Každý VI se skládá ze dvou oken:

- čelního panelu,
- blokového diagramu.

Vykonávání programu v grafickém vývojovém prostředí LabVIEW jednoznačně řídí datový tok. Princip datového toku spočívá v provedení dané části kódu, nazývané uzem, když jsou na jejím vstupu všechny požadované informace potřebné k výpočtu. Po provedení výpočtu jsou na výstupu uzlu definované výsledné informace připravené k dalšímu použití (Vlach, 2008).

Hlavním cílem LabVIEW je nahradit dočasně nebo trvale reálné prostředky virtuálním řešením vytvořeného pomocí G-jazyka. Toto řešení umožňuje jednodušší a levnější návrh nových aplikací a následné provádění změn jejich parametrů.

Aplikace v prostředí LabVIEW je naprogramovaná v G-jazyku. Aplikace je tvořena ze dvou sdružených oken čelního panelu a blokového diagramu (Vlach, 2008).

1.1.1 Grafické programování

Prostředí LabVIEW využívá grafické programování, což je metoda programování, která umožňuje uživateli vytvářet programy pomocí grafických bloků a propojovat je datovými vodiči. Výhodou grafického programování je, že oproti textovému programování není potřeba znát syntaxi programovacího jazyka a k vytvoření fungujícího programu stačí sestavit jeho logiku jako vývojový diagram. Eliminuje také řadu možných syntaktických chyb při programování a také chyb v proměnných.

1.1.2 Datový tok

Bloky jsou propojeny «datovými dráty» - určují pořadí provádění kódu a přenášejí data mezi bloky. Technicky je každý blok sadou kódu, který provádí operace s daty a má vstupy a

výstupy (I/O – input/output). K těmto blokovým I/O se připojují vodiče, které nastavují hodnoty pro určité proměnné v blocích, umožňují nastavení vstupních hodnot tak, aby fungovaly, nebo výstup na výstup v důsledku provádění kódu.

1.1.3 Datové typy

Většina typů proměnných je standardní jako v jiných programovacích jazycích, ale některé jsou specifické a používají se pouze v LabVIEW. V programu LabVIEW lze typ dat přenášených po vodiči porozumět vizuálně pomocí grafického znázornění vodiče: barva označuje typ dat (Integer, Double, Boolean) a změny vzhledu, jako je tloušťka nebo tečkovitost, označe typ proměnné - bitová jednotková proměnná, array nebo cluster.

	Scalar	1D Array	2D Array	Color	Color RGB	Cluster	
Numeric				Orange	rgb(45, 45, 255)		Brown
				Blue	rgb(226, 134, 13)		Brown
Boolean				Green	rgb(42, 149, 42)		Pink
String				Pink	rgb(255, 11, 255)		Pink
Path				Dark Green	rgb(1, 128, 128)		Pink
Reference				Dark Green	rgb(1, 128, 128)		Pink
Hardware Resource				Purple	rgb(132, 9, 132)		Pink
Variant				Purple	rgb(132, 9, 132)		Pink
Waveform				Brown	rgb(160, 64, 0)		Pink
Class				Red	rgb(193, 0, 97)		Pink

Obrázek 1.1 – Typy dat (National Instruments, 2022)

- **Numeric (číselný typ):**
 Floating point – číslo s pohyblivou řádovou čárkou, zobrazené jako oranžové svorky. Může být reprezentována jako single (32-bit), double (64-bit), extended (128-bit) precision (jednoduchá, dvojitá nebo rozšířená přesnost). Číslo s plovoucí desetinnou čárkou může být komplexné;

- Integer – typ celého čísla, zobrazený jako modré svorky. Existují tři reprezentace celých čísel: 8, 16 a 32 bitů. Jeden bit lze použít pro znaménko čísla, pokud je číslo celé číslo se znaménkem.
- Boolean - booleovský typ, zobrazený jako zelené terminály. Booleovský typ může nabývat pouze dvou hodnot: 0 (FALSE) nebo 1 (TRUE).
- String – typ řetězce, zobrazený jako růžové koncovky. Datový typ řetězec obsahuje text ve formátu ASCII.
- Path – cesta k souboru zobrazená v zobrazení terminálu. Cesta k souboru se blíží typu řetězce, ale LabVIEW ji naformátuje pomocí standardní syntaxe pro používanou platformu.
- Array – Pole obsahují datové typy základních prvků a mají příslušnou barvu.
- Cluster – klastry zahrnují různé typy dat. Datový typ clusteru se zobrazí hnědě, pokud jsou všechny jeho prvky číselné, ale pokud jsou prvky clusteru data různých typů, zobrazí se růžově.
- Waveform – datový typ signálu je shluk prvků obsahující data, počáteční časovou hodnotu a časový interval mezi měřeními.
- Dynamic – dynamický typ, zobrazený jako tmavě modré svorky. Kromě dat signálu obsahuje dynamický typ další informace, jako je název signálu nebo datum a čas, kdy byl přijat. Většina expresních uživatelských rozhraní přijímá a/nebo vrací data dynamického typu. Data dynamického typu lze nasměrovat na jakýkoli prvek zobrazení nebo vstupní pole, které přijímá numerická, booleovská data nebo data typu signálu. (Anon., LabVIEW – grafické programování, b. r.; Anon., Data v LabVIEW, b. r.)

1.1.4 Typy proměnných, operace s clusteru a porovnání s datovým typem array

Numeric – obsahuje prvek hodnot/dat.

Array – je to datový typ, který ukládá uspořádanou sadu prvků stejného typu.

Cluster – obsahuje několik proměnných. Může obsahovat proměnné různých typů.

Podrobnější vysvětlení toho, jak různé typy dat fungují:

$U_{min} = 0$ – minimální napětí

$U_{max} = 6$ – maximální napětí

$U_{input} = 2$ – vstupní napětí

$Name$ = napětí – název fyzikální veličiny

V tomto případě jsou proměnné: Umin, Umax, Uinput, Name. 0, 6, 2, napětí – jejich odpovídající hodnoty. Datový typ může být int nebo double. Pro «Name» to bude string. Můžete z nich vytvořit array nebo shluk. Pokud vytvořím array Us a zahrnu data z Umin, Umax, Uinput, proměnná $Us = [0, 6, 2]$. 0, 6, 2 – 3 samostatné hodnoty proměnných uložené v jedné proměnné Us.

Pokud vytvořím cluster Uc, zahrnu do něj proměnné Umin, Umax, Uinput. V tomto případě proměnná, která je shlukem $Uc = [Umin, Umax, Uinput]$ - obsahuje několik dalších proměnných, které zase odpovídají hodnotám 0, 6, 2 – každá hodnota pro svou proměnnou. Zároveň hodnota proměnné Name nemůže být v řetězci, protože mají různé datové typy, ale proměnná může být stále v clusteru. Je možné uložit některé datové typy do řetězců jiného datového typu, pokud jsou nejprve převedeny na vhodný typ, ale k tomu musí být možné převést na tento typ data samotná, tj. číslo typu int nebo double na string a naopak, ale nemůžete převést nečíselné znaky z řetězce na číselný datový typ, jako double.

Technicky mohou existovat vícerozměrné shluky - jedná se o shluky obsahující jiné shluky, což dává smysl pro práci s velkým seskupeným množstvím dat, ale jedná se spíše o referenční informace: v této práci se takové struktury nepoužívají z důvodu nezbytnosti, jelikož s tím program nepracuje s několika skupinami dat a nepřenáší je do různých spustitelných částí a z fyzického vybavení to nevyplývá - pravděpodobně by to mohlo dávat smysl a využít to v případě práce s více zařízeními, která fungují v komplexním, vzájemně funkčně propojeném - kdy datový soubor z jednoho zařízení lze využít v provozu druhého a zároveň i v jakýchkoliv dalších částech celkového integrovaného systému.

V prostředí LabVIEW se datové vodiče používají k propojení více datových terminálů. I/O pole musí být kompatibilní s datovými typy, které jim jsou předávány po drátech. Například nemůžete připojit výstupní pole k poli číselného vstupu. Kromě toho musí být povaha spojení správná. Vodiče by měly být připojeny pouze k jednomu zdroji dat a alespoň jednomu poli pro zadávání dat. Nemůžete například připojit 2 zobrazovací prvky. Komponenty, které určují kompatibilitu připojení, jsou datový typ ovládacího a/nebo zobrazovacího prvku a datový typ vstupního/výstupního pole. (Anon., Data v LabVIEW, b. r.)

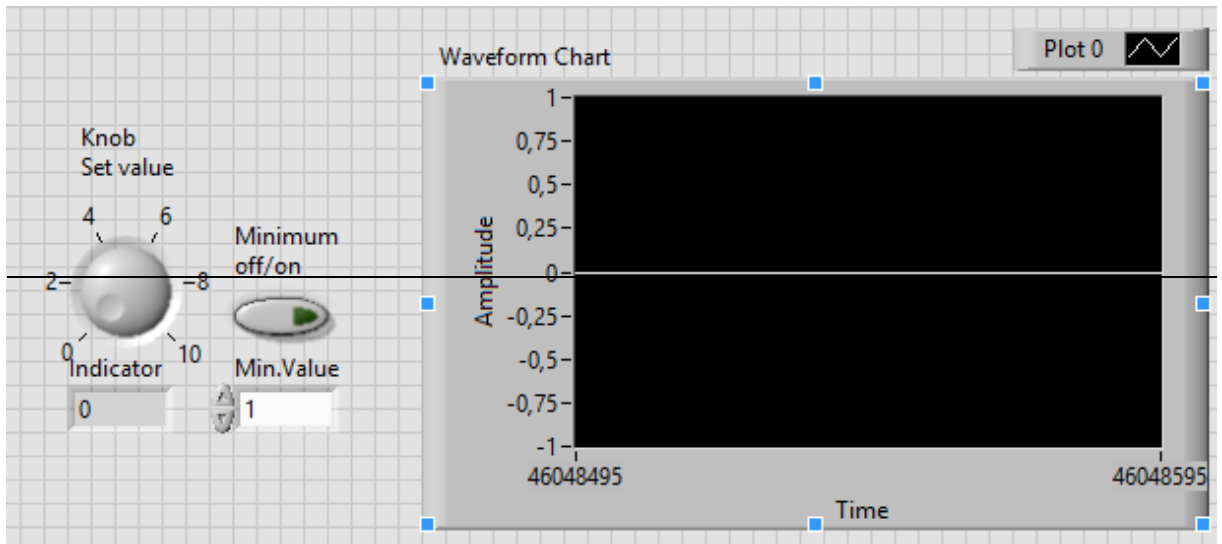
1.1.5 Uživatelské rozhraní

Uživatelské rozhraní prostředí LabVIEW má podobu čelního panelu měřicího přístroje, a to je jeden z důvodů, proč se produkt prostředí LabVIEW nazývá virtuální přístroj.

Vytváření nového virtuálního přístroje se skládá ze dvou sdružených oken a to z uživatelského rozhraní, které se nazývá čelní panel, a z blokového diagramu (Vlach, 2008).

1.1.6 Čelní panel

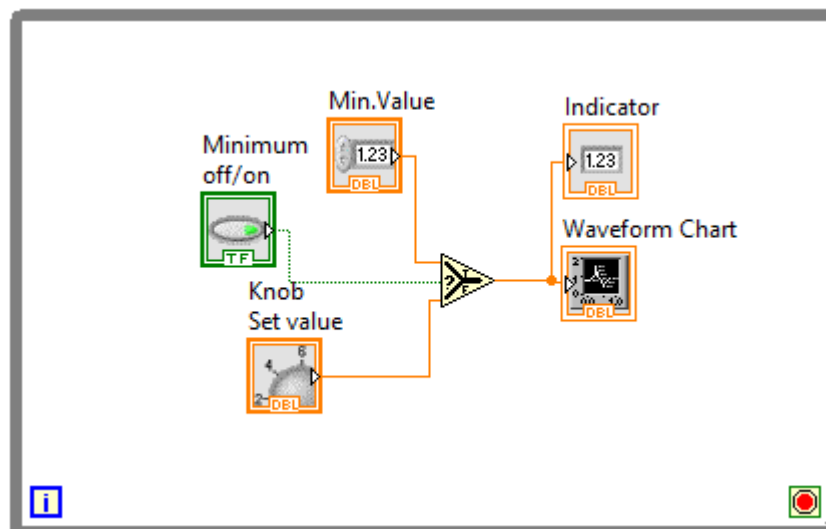
Čelní panel je interaktivní grafické rozhraní pro koncového uživatele, tvoří uživatelské rozhraní aplikace a udává její chování a vzhled. Přes objekty čelního panelu je možné řídit běh aplikace, zadávat parametry a získávat informace o zpracovaných výsledcích. Objekty je možné jakkoliv měnit a upravovat a programově nastavovat (Pechoušek, 2004; Vlach, 2008).



Obrázek 1.2 – Čelní panel

1.1.7 Blokový Diagram

V okně blokového diagramu uživatel definuje samotný algoritmus programu. Blokový diagram je tvořen funkcemi, programovacími strukturami, terminály, které reprezentují ovládací prvky čelního panelu, datovými cestami a bloky. Datové cesty spojující prvky diagramu a jsou tvořeny pomocí „drátů“, které zprostředkovávají datový tok. Blokový diagram je zdrojovým kódem aplikace (Pechoušek, 2004).



Obrázek 1.3 – Blokový diagram (VI)

1.1.8 Typy bloků

Bloky VI lze rozdělit na 2 typy:

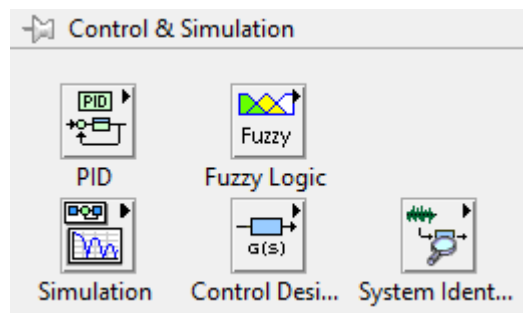
Ovládací a indikační bloky – jsou typu bloku, který se zobrazuje jak na VI, tak na předním panelu, a když tento objekt vytvoříte v jednom okně programu, automaticky se vytvoří ve druhém okně. Zpravidla umožňují uživateli povolit/zakázat (přepnout) libovolnou funkci v programu, nastavit nebo zobrazit hodnotu proměnné, zobrazit data v grafu. Jsou to objekty uživatelského rozhraní, které mu umožňují interakci s programem: nastavit potřebné hodnoty, ovládat zařízení, dozvědět se o jeho stavu.

Logické bloky – typ bloků, které se zobrazují pouze v okně VI, a plní funkci vnitřní logiky programu a provádějí různé transformace nebo práci s proměnnými. Mohou to být algebraické operace, operace porovnávání, přepínání, zápis, generování hodnot.

Běžný uživatel by měl pro práci se zařízením používat pouze bloky předního panelu, zatímco pro programátora je potřeba VI-okno. V jistém smyslu je také možné definovat funkci «designer» job, která by konfiguraci rozhraní předního panelu a typu a uspořádání bloků, datových drátů a skupin bloků na VI udělala tak, aby umožňovala pro uživatele pohodlnější a srozumitelnější práci s oběma okny. To samozřejmě platí i pro pojmenování proměnných a definici skupin bloků. I u interních proměnných má smysl používat intuitivní názvy, stejně jako volitelně psát komentáře k jednotlivým funkcím programu, které by je vysvětlily. To může zjednodušit práci s programem jak pro uživatele třetí strany, tak pro samotného programátora, pokud se program stane poměrně rozsáhlým a složitým, nebo pokud bude potřeba se k němu po nějaké době vrátit. (Halvorsen, 2017)

1.2 CONTROL DESIGN AND SIMULATION MODULE

S modulem LabVIEW Control Design and Simulation lze vytvářet modely zařízení a řízení modelů pomocí přenosové funkce, stavového prostoru nebo nulového pólu. Analyzovat systém pomocí nástrojů, jako je skoková odezva, mapy pólů a nul a Bodeho grafy. Simulovat lineární, nelineární a diskrétní systémy s širokými možnostmi řešení. S modulem NI LabVIEW Control Design and Simulation Module lze analyzovat chování modelu s otevřenou smyčkou, navrhovat regulátory s uzavřenou smyčkou, simulovat online a offline systémy a provádět fyzické implementace. (National Instruments, 2023d; Halvorsen, 2017)



Obrázek 1.4 – Podknižovny modulu Control and Simulation

1.2.1 Knihovna Simulation

Hlavním účelem knihovny Simulation je proces simulace, který zahrnuje použití softwaru pro analýzu chování dynamických systémů. Tento proces vede ke snížení nákladů a urychlení vývoje produktu.

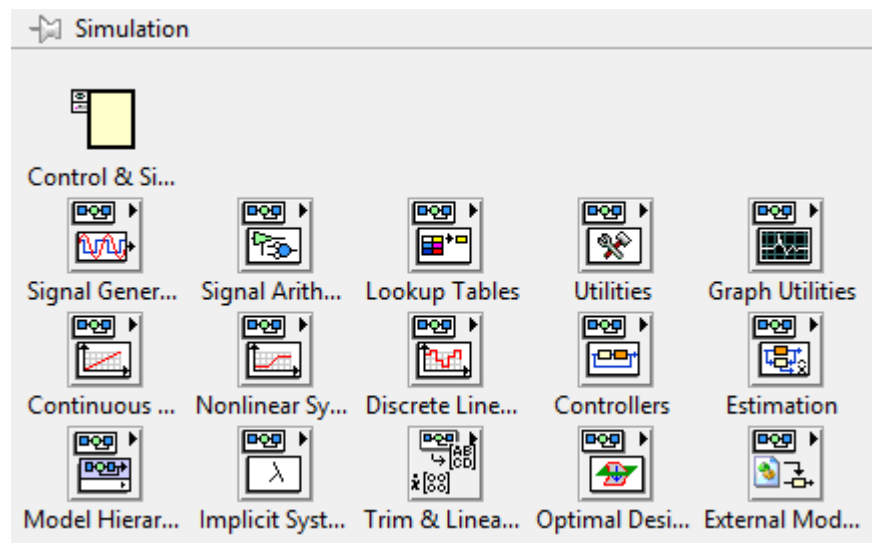
Hlavním blokem knihovny Simulation je blok Control & Simulation Loop, který udává parametry a čas simulace (doba simulace, krok, metoda řešení). Control & Simulation Loop také definuje vizuální hranici simulačního diagramu, všechny prvky simulace musí být umístěny uvnitř této hranice.

Knihovna Simulation obsahuje spoustu podknižoven, které jsou potřeba k simulaci. Mezi možnostmi, které poskytují podknižovny, patří generování signálů, aritmetické operace se signály, umožňuje tvorbu spojitých signálů, tvorbu regulátorů a další.

Hlavní funkce v paletě Simulace jsou:

- Control & Simulation loop – Všechny funkce simulace je nutné umístit do řídicí a simulační smyčky nebo do simulačního subsystému.
- Continuous Linear Systems – reprezentace spojitých lineárních systémů pomocí diferenciálních rovnic v simulačním schéma.

- Signal arithmetic – funkce k provádění základních aritmetických operací se signály v simulačním systému. (Halvorsen, 2017)

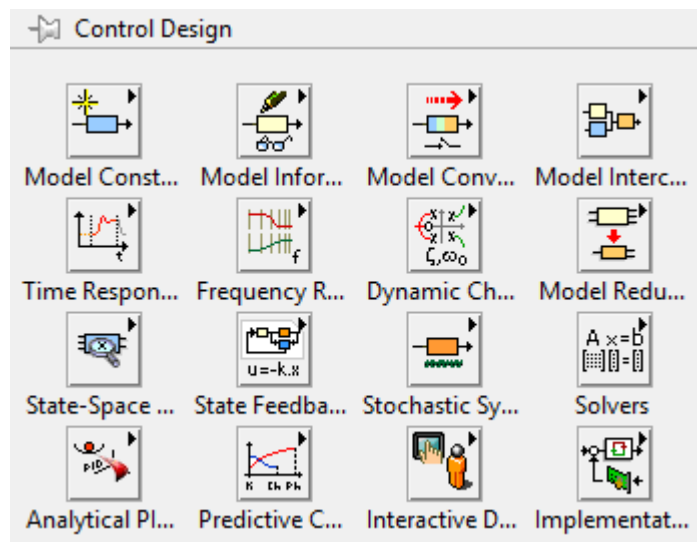


Obrázek 1.5 – Podknihovna Simulation

1.2.2 Knihovna Control Design

Účelem knihovny Control design je proces, který zahrnuje vytvoření matematického modelu popisujícího fyzický systém, analýzu matematického modelu za účelem zjištění dynamických charakteristik a návrh regulátoru pro dosažení požadovaných dynamických charakteristik.

Řídicí systém obsahuje prvky, které řídí a regulují fyzický systém. V tomto případě se řídicí systém skládá z regulátoru, senzoru a akčního členu. (National Instruments, 2023d).

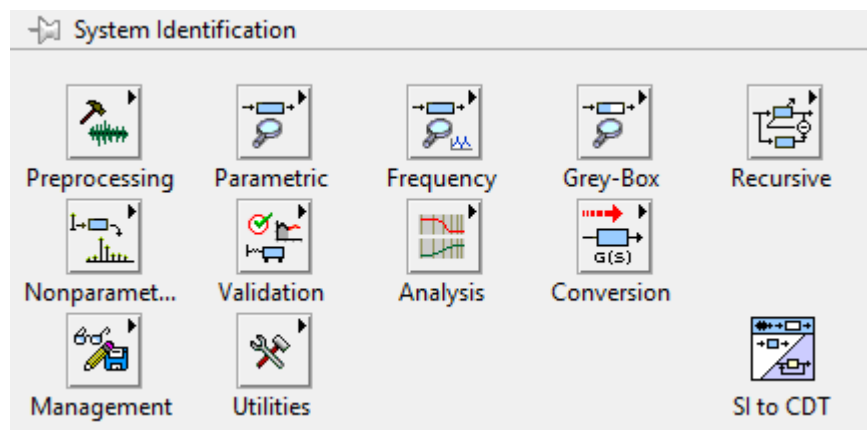


Obrázek 1.6 – Podknihovna Control Design

Základním stavebním prvkem knihovny simulation je programová struktura Control & Simulation Loop. Blok Control&Simulation Loop udává parametry simulace a definuje plochu simulačního blokového diagramu, do které se musí prvky simulace vkládat. (National Instruments, 2023a)

1.2.3 Knihovna System Identification

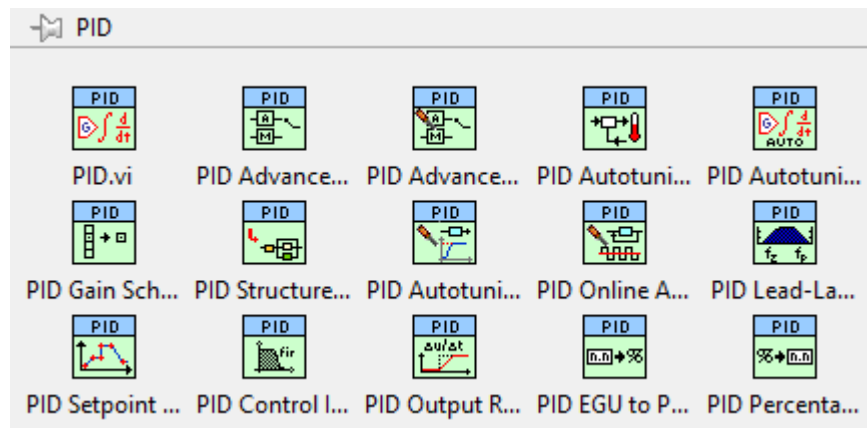
LabVIEW System Identification Toolkit kombinuje nástroje pro sběr dat s algoritmy identifikace systému pro modelování systému. Lze použít LabVIEW System Identification Toolkit k nalezení empirických modelů ze skutečných systémových informací na základě odezvy na vybrané buzení. (Halvorsen, 2017)



Obrázek 1.7 – Podknihovna System Identification

1.2.4 Knihovna PID

Algoritmus PID je nejběžnějším řídicím algoritmem používaným v průmyslu. Proporcionálně integračně derivační (PID) regulátor je zpětnovazební regulátor, který lze použít k řízení procesů, jako jsou systémy vytápění a chlazení, monitorování hladiny kapalin, řízení průtoku, regulace tlaku atd. PID VI lze použít s hardwarem National Instruments, jako je zařízení DAQ, I/O modul FieldPoint nebo deska GPIB, k připojení počítače k systému a vývoji řídicích aplikací v LabVIEW. Je možné také použít knihovnu PID s matematickými a logickými funkcemi LabVIEW k vytvoření blokových diagramů pro skutečné řídicí strategie.

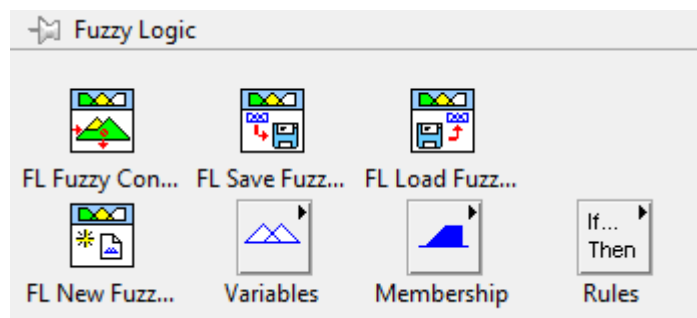


Obrázek 1.8 – Podknihovna PID

1.2.5 Knihovna Fuzzy Logic

Fuzzy logika je metoda rozhodování na základě pravidel pro expertní systémy a řízení procesů. Fuzzy logika se liší od tradiční booleovské logiky tím, že umožňuje částečnou příslušnost v množině. K řízení procesů reprezentovaných subjektivními lingvistickými popisy lze použít fuzzy logiku. Získaná data lze zapojit do fuzzy regulátoru a implementovat rozhodování v reálném čase nebo řízení fyzického systému. Je možné také použít výstupy fuzzy regulátoru s analogovým výstupním hardwarem DAQ k implementaci řízení procesů v reálném čase.

(National Instruments, 2023c)



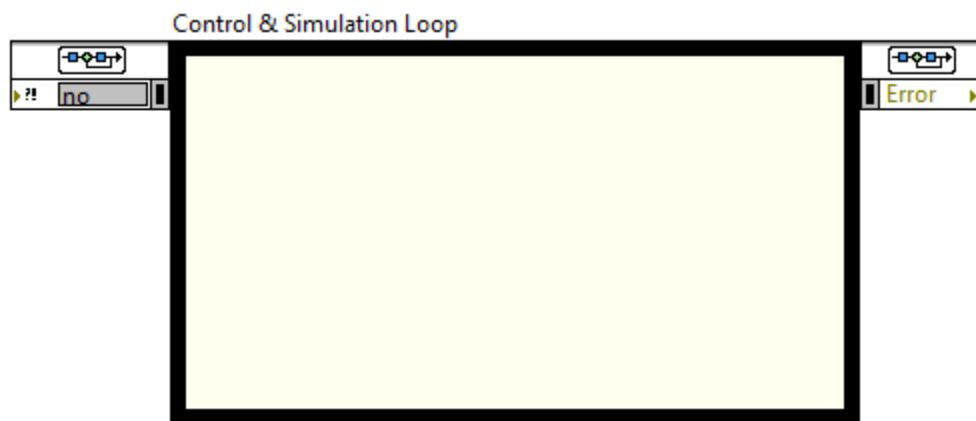
Obrázek 1.9 – Podknihovna Fuzzy Logic

Knihovna Fuzzy Logic na obrázku 1.9 umožňuje použití fuzzy logiky k návrhu a řízení fuzzy systémů. Fuzzy systém je systém proměnných, které mají určitý stupeň příslušnosti. Fuzzy regulátor používá uživatelem definovaná pravidla pro řízení fuzzy systémů, které jsou závislé na hodnotách vstupních proměnných. (National Instruments, 2023b)

1.2.6 Control & Simulation Loop

Provádí simulační diagram, dokud řídicí a simulační smyčka nedosáhne času ukončení simulace nebo dokud funkce zastavení simulace nezastaví provádění programově. Všechny simulační funkce je nutné umístit do řídicí a simulační smyčky nebo do simulačního subsystému. (National Instruments, 2023d)

K ovládání zařízení ve VI se používá Control & Simulation Loop k nastavení parametrů simulace a cyklickému běhu programu. Hodnotu parametrů lze nastavit v okně nastavení samotného bloku, nebo lze hodnoty přivádět na vstup z jiných bloků. Některé z parametrů používaných v simulaci jsou: čas, který nastavuje dobu trvání simulačního cyklu, minimální a maximální délka jednoho kroku.



Obrázek 1.10 – Control & Simulation Loop

2 ZAŘÍZENÍ A PŘÍSTROJE

2.1 NI ELVIS II+

NI Engineering Laboratory Virtual Instrumentation Suite (NI ELVIS) II+ je modulární inženýrské vzdělávací laboratorní zařízení vyvinuté speciálně pro akademickou půdu. Díky praktickému přístupu mohou pedagogové pomoci studentům získat praktické, experimentální dovednosti. NI ELVIS II+ je vybaven jedním kompaktním modulem s integrovanými 12 nejběžněji používanými přístroji v laboratoři, včetně osciloskopu s rozšířenou funkcí, digitálního multimetru, funkčního generátoru, variabilního napájecího zdroje a Bodeho analyzátoru. K těmto různým zařízením lze připojit počítač pomocí USB plug-and-play a sestavovat obvody na odnímatelné prototypové desce. (National Instruments, b. r.)



Obrázek 2.1 – Komponenty NI ELVIS II+ (Quanser, 2015)

Tabulka 2.1 – Komponenty ELVIS II+ (Quanser, 2015)

Číslo	Popis
1	NI ELVIS II+
2	Spínač napájení prototypové desky
3	LED napájení prototypové desky
4	LED napájení základního modulu
5	Napájecí kabel pro NI ELVIS II+
6	USB propojení mezi PC a NI ELVIS II+

2.2 QUANSER QNET PHYSICS AND DYNAMICS BOARD

QNET Physics and Dynamics board je praktická platforma pro výuku a demonstraci základů dynamického modelování a analýzy. Tato deska, navržena exkluzivně pro platformu NI ELVIS a software LabVIEW, umožňuje studentům řešit řadu nejběžnějších dynamických systémů, včetně tření, systémů sil, kinematiky částic, energie a hybnosti, mechanických vibrací a dalších. (National Instruments, 2015)



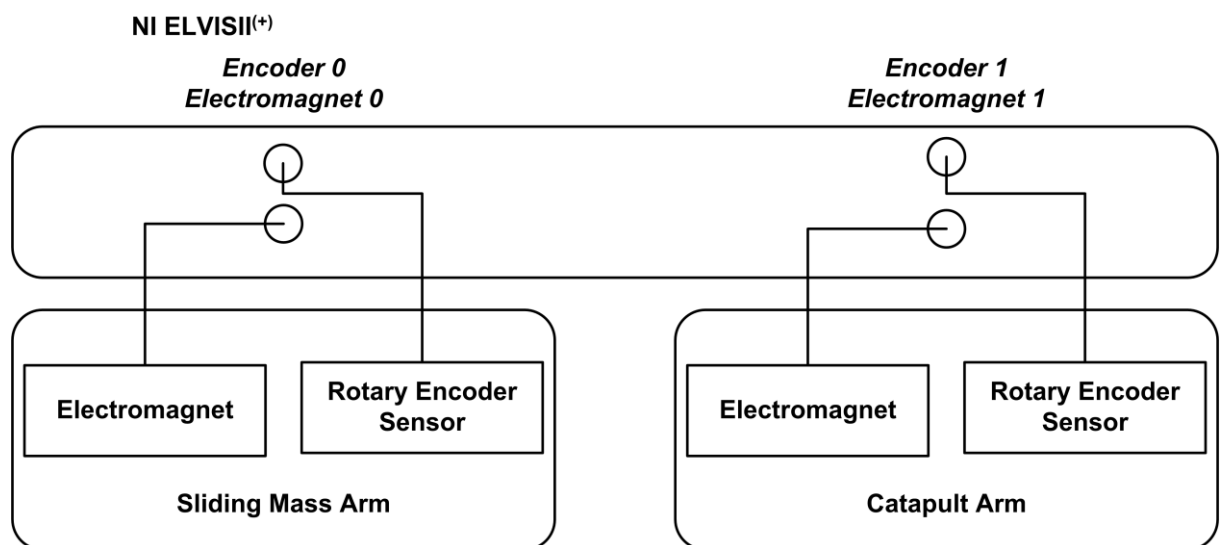
Obrázek 2.2 – Qanser QNET Physics and Dynamics Board (Quanser, 2015)

2.2.1 Výuka základních principů fyziky a dynamiky

Každý inženýrský program zahrnuje alespoň jeden kurz zaměřený na inženýrskou dynamiku a základní fyzikální principy. QNET Physics and Dynamics Board nabízí ideální rozšíření pro takové, typicky čistě matematické kurzy, poskytující praktickou platformu pro výuku a demonstraci základů dynamického modelování a analýzy.

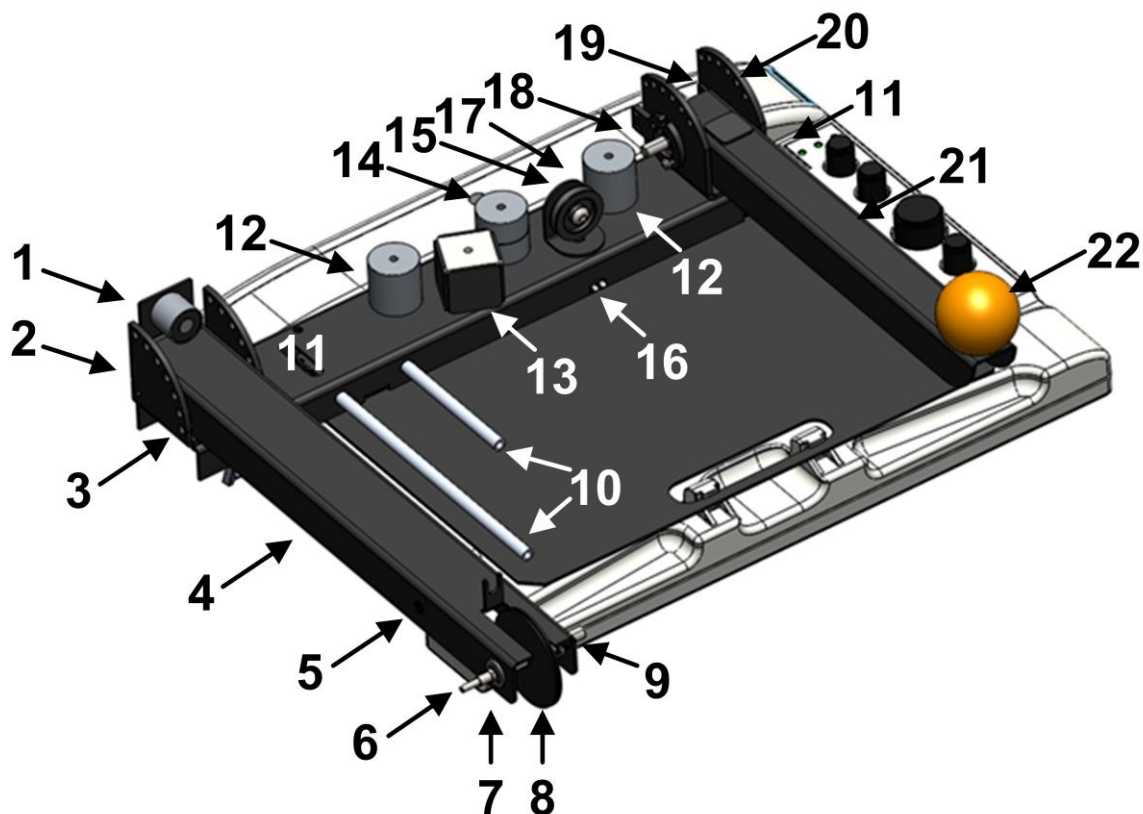
2.2.2 Popis práce QNET Physics and Dynamics Board

Deska QNET Physics and Dynamics se skládá z katapultu s nastavitelným úhlem uvolnění a posuvného ramena s nastavitelným sklonem. Úhly katapultu a posuvného ramene hmoty jsou měřeny optickými enkodéry. Chování experimentů s katapultem a posuvnou hmotou je řízeno elektromagnety, které uvolňují posuvné rameno a rameno katapultu. Deska QNET Physics and Dynamics obsahuje také sadu klasických mechanických prvků, jako jsou pružiny, kladky, hmoty s různými povrchovými materiály (PETG, styren, pryž, teflon) a kyvadla, které lze nastavit v různých konfiguracích pro širokou škálu experimentů.



Obrázek 2.3 – Interakce mezi komponentami QNET Physics a Dynamics (Quanser, 2015)

2.2.3 Hardwarové komponenty



Obrázek 2.4 – Obecné rozložení QNET Physics and Dynamics (Quanser, 2015)

Hlavní komponenty obsahující QNET Physics and Dynamics jsou označeny na obrázku 2.4 a jsou uvedeny v tabulce 2.2.

Tabulka 2.2 – Nomenklatura komponent QNET Physics and Dynamics (Quanser, 2015)

Číslo	Popis
1	Elektromagnet
2	Nastavitelný kloub ramene posuvné hmoty
3	Čep pro nastavení úhlu ramene posuvné hmoty
4	Rameno posuvné hmoty
5	Kuželová koncovka pro experiment s kladkou
6	Experimentální oscilační terminál
7	Otočné kolo kladky
8	Kladka pro experiment s posuvnou hmotností
9	Enkodér ramene
10	Oscilační experimentální tyče

Tabulka 2.2 – Nomenklatura komponent QNET Physics and Dynamics - pokračování

11	5-pinový kodér a 2-pinové konektory elektromagnetu
12	100g závaží
13	Vícestranná posuvná krychle
14	50g závaží
15	Volně pohyblivá kladka
16	+5V LED
17	PCI konektor
18	Enkodér ramen katapultu
19	Nastavitelný úhel uvolnění katapultového ramene
20	Čep pro nastavení úhlu spuštění katapultu
21	Rameno katapultu
22	Míček s ramenem katapultu

Zařízení QNET Physics a Dynamics má 4 hlavní prvky, se kterými lze komunikovat prostřednictvím LabVIEW:

2 elektromagnety

2 enkodéry

Kanál 0 odpovídá elektromagnetu a enkodéru souvisejícímu s posuvnou hmotou, kanál 1 odpovídá dvojici elektromagnetu a enkodéru katapultu.

2.2.4 Kulatý elektromagnet EM075-6-222

Kulatý elektromagnet EM075-6-222 je vysoce kvalitní, precizně vyrobený produkt navržený pro použití v různých průmyslových a komerčních aplikacích. S průměrem cca 19 mm a schopností udržet až 2,7 Kg. Díky své hmotnosti je tento elektromagnet ideální pro použití v automatizovaných třídících a montážních systémech, robotických strojích a dalších aplikacích, kde je vyžadováno silné a spolehlivé magnetické pole. Napájecí zdroj 6 V DC umožňuje efektivní a spolehlivý provoz, zatímco odolná konstrukce zajišťuje dlouhou životnost i v náročných provozních podmínkách.



Obrázek 2.5 – Elektromagnet

2.2.5 Enkodér E8P-512-250-S-H-M-B

Enkodér používaný k měření úhlové polohy katapultu a ramene posuvné hmoty na QNET Physics and Dynamics je optický hřídelový enkodér. Poskytuje 2048 impulzů na otáčku v kvadrantním režimu (512 řádků na otáčku).

2.3 QNET 2.0 VTOL BOARD

QNET 2.0 Vertical Take-off and Landing (VTOL) Board je ideální pro výuku a demonstraci základů letové dynamiky a řízení letu při vertikálním vzletu a přistání. Systém vyvinutý exkluzivně pro platformu NI ELVIS a software LabVIEW lze snadno nakonfigurovat tak, aby řídil let trenážeru pomocí různých metod ovládání. (Quanser, 2014)

VTOL NI ELVIS Controls Device – Quanser QNET VTOL Board 2.0 pro NI ELVIS II/II+ je přídatná aplikační deska pro NI Engineering Laboratory Virtual Instrumentation Suite II (NI ELVIS II) nebo NI ELVIS II+. Toto zařízení bylo vyvinuto pro vzdělávání, aby usnadnilo praktické, aktivní učení dynamiky letu s vrtulníkem s 1 stupněm volnosti (DOF) zahrnující kinematiku, vertikální vzlet, PI a PID řízení. Aplikační deska poskytuje ventilátor s proměnnou rychlostí namontovaný na konzolovém rameni s optickým enkodérem s vysokým rozlišením pro měření polohy. (National Instruments, 2014)



Obrázek 2.6 – QNET 2.0 VTOL Board (Quanser, 2014)

2.3.1 Stručný popis nového zařízení

Vzhledem k fyzické a technické nemožnosti implementace ovládání na zařízení Quanser QNET Physics and Dynamics Board bylo pro regulaci použito další laboratorní zařízení, které podporuje ovládání pomocí LabVIEW a regulaci – Quanser QNET 2.0 VTOL. Jedná se o desku s enkodérem a ramenem namontovaným na vodorovné hřídeli připojené k enkodéru. Na jedné straně je pevné závaží, na druhé ventilátor, který vytváří proud vzduchu směřující dolů, což umožňuje ovlivnit stav systému.

Na desce jsou 3 porty:

Amplifier (zesilovač) - připojený k ventilátoru. Ovládá ventilátor.

Encoder 0 - je ve stejném svazku vodičů se zesilovačem. Enkodér namontovaný na ventilátoru pro měření jeho otáčení.

Encoder 1 - Připojen k enkodéru na hřídeli ramene. Umožňuje měřit natočení celého systému.

3 POPIS ZÁKLADNÍCH VI A SUBVI

V rámci této kapitoly bude popsána použitá softwarová součást práce v programu LabVIEW, použité bloky, jejich účel a také účel blogových skupin sdružených do společných systémů.

NI poskytuje několik souborů LV nazvaných «VI» pro práci s QNET Physics and Dynamics board založenou na ELVIS II+ pomocí programu LabVIEW. Mezi ně patří:

- QNET Physics and Dynamics Quick Start VI - základní schéma pro rychlé zahájení práce s přístrojem, umožňující základní ovládání a měření.
- Řada různých variací Quick Start - pro různé fyzikální experimenty a měření. Nebyly použity, protože během práce bylo použito VI «QNET Physics and Dynamics Quick Start VI» a bylo pro plánovanou regulaci vhodnější.

Složka «SubVIs» obsahující několik podsystémů - sadu vzájemně propojených různých bloků, které v kombinaci plní v rámci hlavního systému nějakou samostatnou nezávislou funkci.

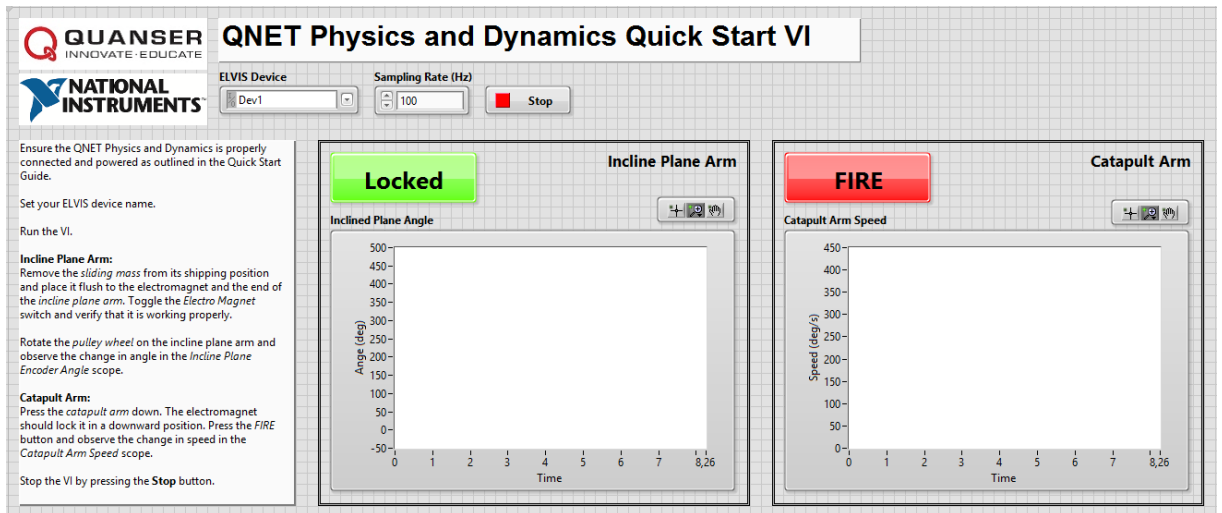
- DIO – inicializace výstupů zařízení pro ovládání
- Encoder Config - inicializace vstupů zařízení pro měření
- Firing Subsystem - nastavení provozních podmínek katapultu

Na blokovém diagramu se objevují jako jeden blok se sadou vstupů a výstupů, ale lze je otevřít v samostatném okně jako samostatné VI s vlastním oknem Přední panel a Blokovým diagramem.

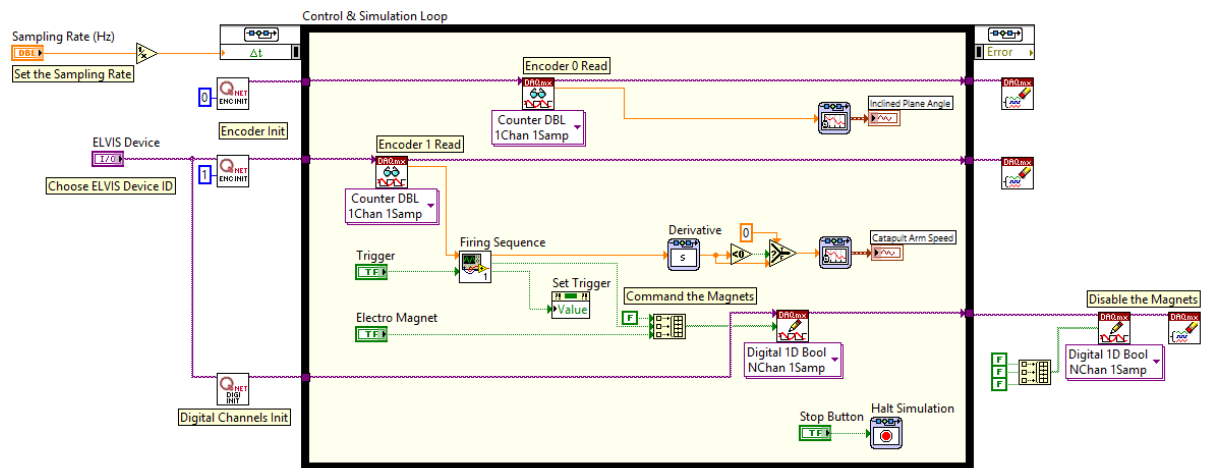
Použití subVI výrazně snižuje prostor zabraný na blokovém diagramu hlavního programu a také zlepšuje čitelnost vizuálního kódu. Kromě toho použití subVI umožňuje vícenásobné opětovné použití stejného kódu v jednom nebo více hlavních programech.

Samostatným plusem a mínusem je, že subVI se ukládá jako samostatný soubor, ke kterému jsou přidruženy další spustitelné soubory, což na jedné straně vede k pohodlné možnosti použití stejného řešení v různých systémech, stejně jako k jeho přidání a zlepšení, ale neznalost této nebo jakékoli chyby může vést k ukončení všech souvisejících programů VI a s individuálním řešením pro samostatný program nebo jeho samostatnou kódovou sekci je třeba opustit subVI a použít soukromé řešení.

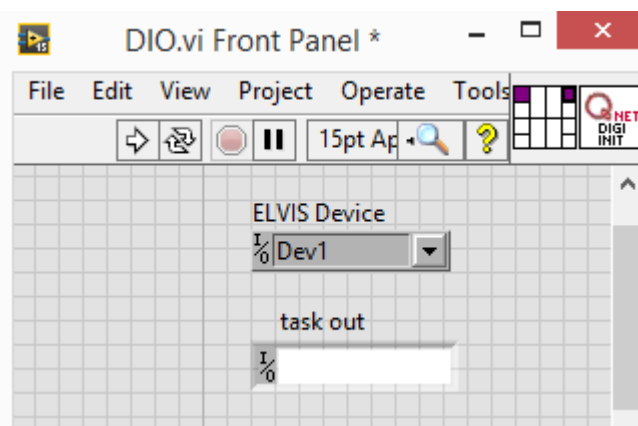
Vstupy a výstupy subVI mohou být konfigurovány uživatelem: jejich počet, umístění a který vstup nebo výstup bloku subVI je přidružen ke kterému prvku v něm.



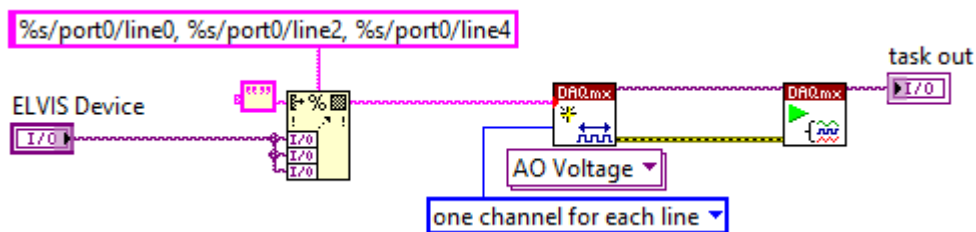
Obrázek 3.1 – Čelní panel systému QNET Physics and Dynamics Quick Start VI



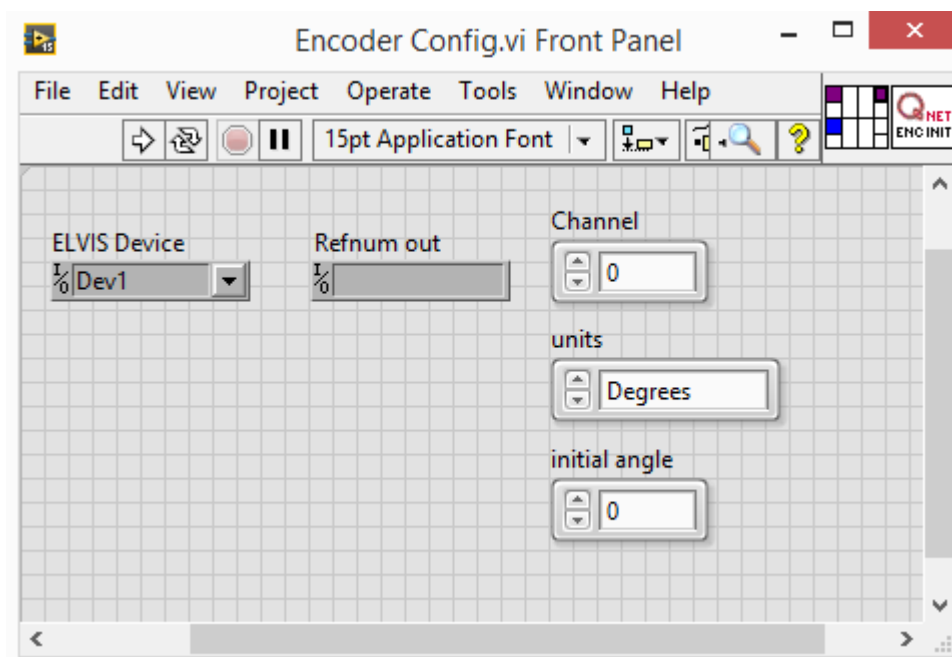
Obrázek 3.2 – VI systému QNET Physics and Dynamics Quick Start VI



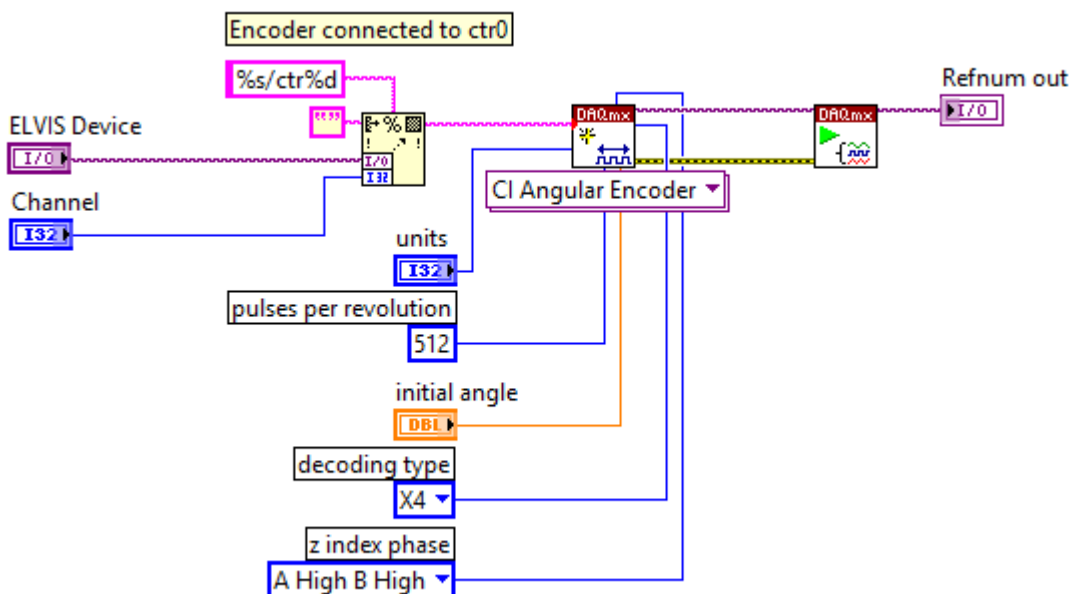
Obrázek 3.3 – Čelní panel subVI DIO



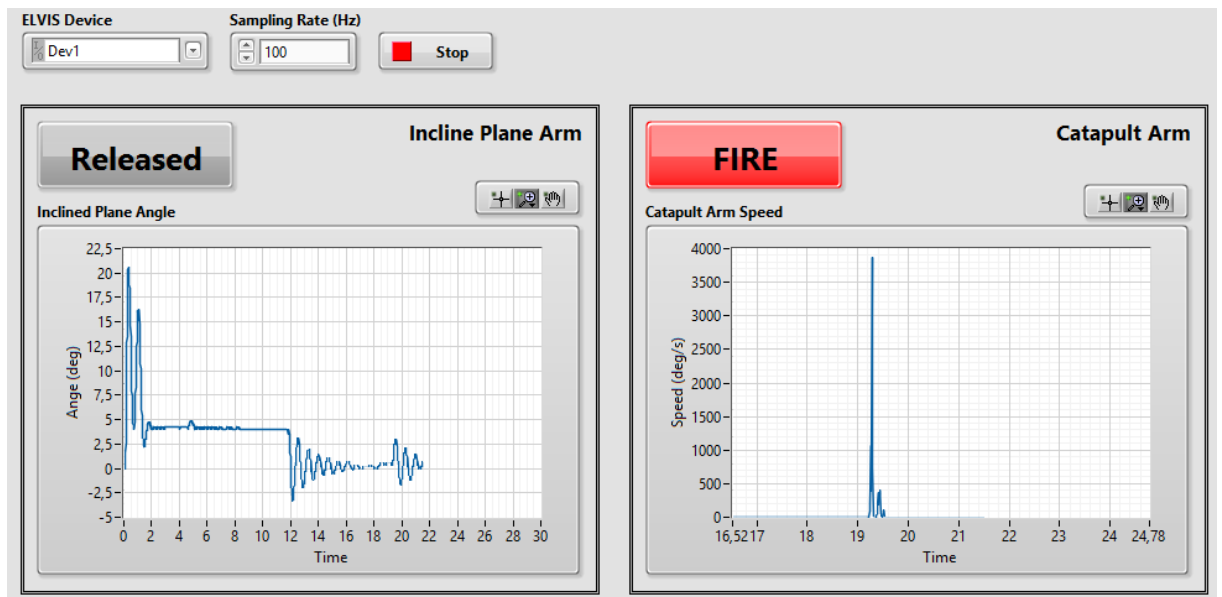
Obrázek 3.4 – VI subsystému DIO



Obrázek 3.5 – Čelní panel subVI Encoder Config



Obrázek 3.6 – VI subsystému Encoder Config

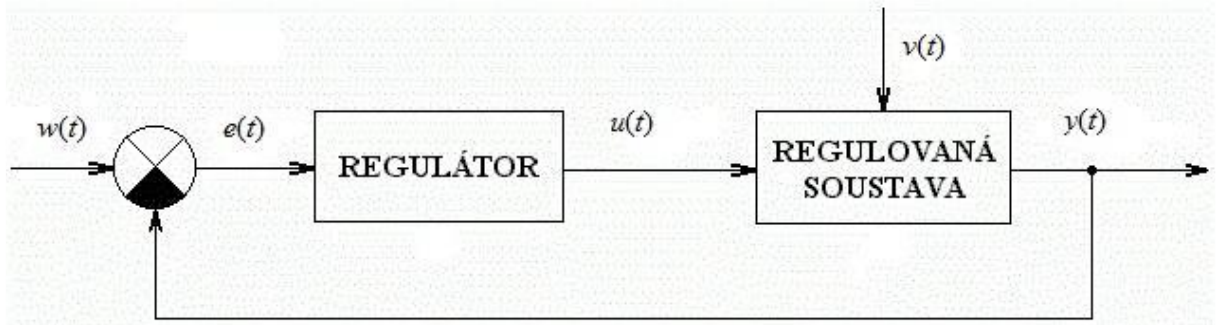


Obrázek 3.7 – Ukázka čelního panelu systému QNET Physics and Dynamics

4 TEORETICKÝ POPIS REGULÁTORU A REGULACE

Úkolem řízení je pomocí řídicího systému cíleně působit na řízený systém, tak aby byl výsledek řízení v souladu s cílem řízení. Použití informace o stavu řízeného systému zvýší kvalitu řízení – zavedení zpětné vazby. Poruchy ovlivňují výsledek řízení – působí na řízený subsystém. Informaci o poruchách lze také využít v řídicím subsystému a kompenzovat jejich vliv dříve, než se projeví na výsledku řízení.

(Honc, 2018)

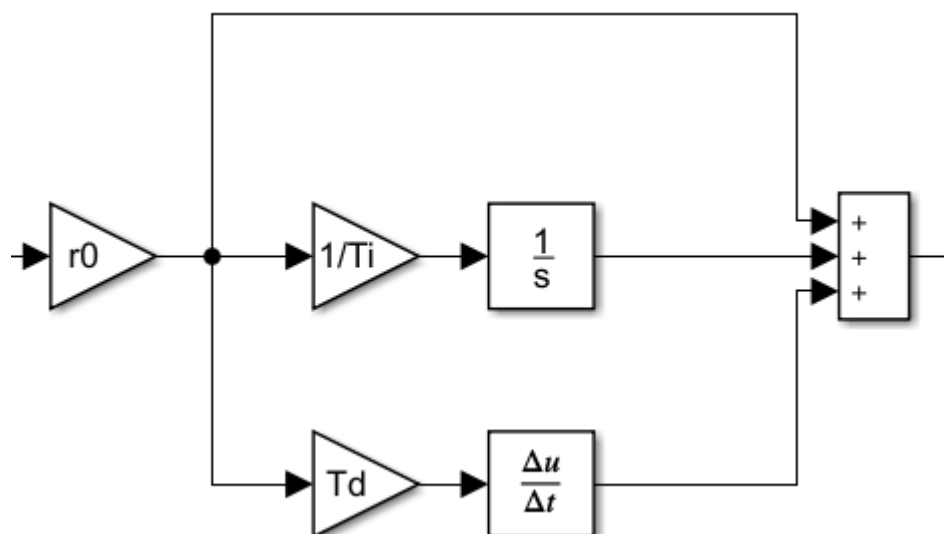


Obrázek 4.1 – Zapojení URO s regulátorem ve zpětné vazbě

4.1 PID REGULÁTOR

4.1.1 Popis PID regulátoru

PID je proporcionálně integračně diferenciální regulátor. PID regulátory jsou široce používány v moderních přesných řídicích systémech. Fyzikální veličina měřená senzorem je přivedena na vstup regulátoru, regulátor v souladu se svým algoritmem generuje akční veličinu. To způsobí změnu regulované veličiny.



Obrázek 4.2 – Schéma PID regulátoru

PID regulátor obsahuje tři základní složky:

1. Proporcionální složka.

Proporcionální člen jednoduše vynásobí regulační odchylku.

2. Integrovaná složka.

Umožní zohlednit předchozí průběh řízení a řízení postupem času stále zpřesňovat. Regulátor sečte všechny předchozí hodnoty regulační odchylky a provede pro ně korekci. Jakmile systém dosáhne žádané hodnoty (například dosáhne nastavené teploty), regulační odchylka se bude blížit nule a integrovaná složka bude mít v průběhu času stále menší vliv na činnost regulátoru.

3. Derivační složka.

Tato složka násobí derivaci naměřené hodnoty. Derivace je rychlost změny fyzikální veličiny. Například čím rychleji teplota v systému stoupá (nebo klesá), tím větší bude odpovídající derivace. Diferenciální složka umožňuje regulátoru reagovat odlišně na prudké a plynulé změny regulované veličiny v systému, čímž dochází k snížení jejího kmitání. (Anon., Popis funkcí PID regulátoru, b. r.)

4.1.2 Komponenty PID regulátoru

Regulátor, který pouze zesiluje regulační odchylku e , se nazývá proporcionální, neboli P regulátor. P regulátor je popsán rovnicí:

$$u = r_0 e \quad (4.1)$$

kde u – akční veličina,

r_0 –proporcionální zesílení,

e – regulační odchylka.

Dalším regulátorem je integrovaná, neboli I regulátor, kde akční veličina u je rovna integrálu regulační odchylky e . I regulátor je popsán rovnicí

$$u(t) = r_{-1} \int_0^t e(t) dt \quad (4.2)$$

kde

r_{-1} – integrovaná zesílení,

Derivační člen, jinak nazýván D člen, pouze regulační odchylku e derivuje, tento typ regulátoru je používán pouze u kombinovaných regulátorů. D člen je popsán rovnicí

$$u(t) = r_1 \frac{de(t)}{dt} \quad (4.3)$$

kde r_1 – derivační zesílení,

Kombinace hlavních typů regulátorů zlepšují vlastnosti regulačního obvodu. Kombinované regulátory zahrnují PI regulátor, PD regulátor a PID regulátor.

Dynamické chování PID regulátoru můžeme popsat vztahem

$$u(t) = r_0 \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (4.4)$$

kde r_0 – proporcionalní zesílení PID regulátoru,

T_i – integrační časová konstanta PID regulátoru,

T_d – derivační časová konstanta PID regulátoru,

Provede-li se Laplaceova transformace za předpokladu splnění nulových počátečních podmínek, můžeme rovnici výše upravit na přenos PID regulátoru, který je dán vztahem

$$U(s) = r_0 \left[1 + \frac{1}{T_i s} + T_d s \right] E(s) \quad (4.5)$$

4.2 METODY PRO NALEZENÍ OPTIMÁLNÍCH PARAMETRŮ REGULÁTORU

4.2.1 Integrální kritéria kvality

Integrální kritéria kvality poskytují obecné posouzení doby regulace a stupně odchylky regulované veličiny od ustálené hodnoty v přechodném procesu v souhrnu, aniž by bylo nutno hledat obojí samostatně.

IE – Lineární regulační plocha

$$J_{IE} = \int_0^{\infty} e(t) dt = \int_0^{\infty} (w - y)(t) dt \quad (4.6)$$

$$e = w - y$$

$$y \rightarrow w$$

$$e \rightarrow 0$$

kde e – regulační odchylka,

w – žádaná hodnota,

y – výstupní hodnota.

Plocha bude tím menší, čím rychleji dojde k odeznění přechodného jevu a čím menší bude odchylka. Proto se doporučuje volit parametry systému tak, aby bylo dosaženo minima tohoto integrálního odhadu.

IAE – Absolutní regulační plocha

$$J_{IAE} = \int_0^{\infty} |e(t)| dt \quad (4.7)$$

Tedy součet absolutních hodnot všech oblastí pod přechodnou křivkou. Ale vypočítat to z koeficientů rovnice je obtížné.

V tomto ohledu se v obecném případě používá kvadratické integrální kritérium kvality.

ISE – kvadratická regulační plocha (VŠB – TU Ostrava, b. r.)

$$J_{ISE} = \int_0^{\infty} e^2(t) dt = \int_0^{\infty} (w - y)^2(t) dt \quad (4.8)$$

5 METODY NASTAVENÍ PID

5.1 EXPERIMENTÁLNÍ METODY KRITICKÝCH PARAMETRŮ

Experimentální metody jsou založeny na přímé práci se skutečným zařízením a někdy mohou vyžadovat přidání dalších fyzických zařízení. Při zpracování této práce byly použity simulační metody v prostředí MATLAB a Simulink.

5.1.1 Zieglerova-Nicholsova metoda kritických parametrů

Tato metoda uzavřeného regulačního obvodu je vhodná pouze pro systémy vyšších řádů a je u ní možné uvažovat dopravní zpoždění. Jedním z rysů této metody je poměrně agresivní naladění, které u některých systémů nemusí zaručit stabilitu. (Dušek, 2023)

Kritické zesílení r_{0k} a kritická perioda T_k jsou vlastnosti systému, které vyjadřují jeho vlastnosti související se stabilitou. Lze je stanovit pouze pro systémy, ve kterých lze nestabilitu dosáhnout pomocí zpětné vazby: amplitudo-fázová frekvenční charakteristika by měla zasahovat do 2. kvadrantu, tzn. systém je 3. nebo vyššího řádu.

Je založena na reálném řídicím obvodu, který je přiveden na mez oscilační stability vyřazením integrační složky ($T_i \rightarrow \infty$) a derivační složky ($T_d \rightarrow 0$) zvýšením zesílení regulátoru.

Poté se odečte kritická perioda T_k a kritické zesílení r_{0k} regulátoru. Hodnoty parametrů pro vybranou variantu regulátoru jsou vypočteny na základě tabulky 5.1. (Dušek, 2023; Vítečková, 2011)

$$r_{0k} = \frac{4A}{\pi e_{max}} \quad (5.1)$$

Tabulka 5.1 – Stavitelné parametry PID regulátoru pro ZN metodu kritických parametrů

Regulátor	r_0	T_i	T_d
P	$0,5r_{0k}$	—	—
PI	$0,45r_{0k}$	$0,83T_k$	—
PID	$0,6r_{0k}$	$0,5T_k$	$0,125T_k$

Ziegler-Nicholsova metoda kritických parametrů je výhodná především proto, že nevyžaduje žádné znalosti o vlastnostech řízené soustavy a protože pracuje s reálným systémem a regulátorem, i když pro práci s reálným systémem vyžaduje další vybavení. Ale může být také

použit v simulaci, pokud je známá funkce popisující model systému. Jejím omezením je, že musí přivést řídicí obvod na hranici oscilační stability, tzn. by měly kolísat, což většina reálných systémů neumožňuje a v případě práce s reálnými systémy není vždy možné je z hlediska časových nákladů prakticky uvést na mez stability.

(Dušek, 2023; Vítečková, 2011)

5.1.2 Experimentální Tyreusova-Luybenova metoda kritických parametrů

Kritické parametry pro seřízení analogových regulátorů PI a PID využívá Tyreusova-Luybenova metoda (tab. 4.2), která je však dost konzervativní. (Vítečková, 2011)

Tabulka 5.2 – Stavitelné parametry analogových regulátorů pro Tyreusovu-Luybenovu metodu kritických parametrů

Regulátor	r_0	T_i	T_d
PI	$0,31r_{0k}$	$2,2T_k$	—
PID	$0,45 r_{0k}$	$2,2T_k$	$0,16T_k$

Pro zvolený typ regulátoru se z tabulky 5.1 a 5.2 určí hodnoty jeho stavitelných parametrů.

5.2 ANALYTICKÉ METODY

Tento typ metody se opírá o matematické výpočty a metody pro aproximaci reálného modelu různými způsoby a nevyžaduje použití simulace v prostředí Simulink.

5.2.1 Metoda Cohen-Coon (výpočet pro aproximační přenos)

Určení parametrů variant PID regulátoru výpočtem z aproximace soustavy přenosem prvního řádu s dopravním zpožděním, a při výpočtu parametrů regulátoru používá parametry: dopravní zpoždění D , doba regulace T a zesílení Z . Z nich se počítá α a τ , které spolu s parametrem dopravního zpoždění D umožňují vypočítat parametry pro různé kombinace regulátoru: P, PI, PD, PID.

Princip: umístění dominantních pólů URO, nastavení na potlačení poruchy, tlumení na $\frac{1}{4}$ periody kmitů při poruše ve tvaru jednotkového skoku. (Dušek, 2023)

Stavitelné parametry analogových regulátorů pro Cohen-Coonovu metodu:

$$G_s(s) = \frac{Z}{Ts + 1} e^{-Ds} \quad (5.2)$$

$$r_0 = \frac{1,35}{a} \left(1 + \frac{0,18t}{1-t} \right) \quad (5.3)$$

$$T_i = \frac{2,5 - 2,0t}{1 - 0,39t} D \quad (5.4)$$

$$T_d = \frac{0,37 - 0,37t}{1 - 0,81t} D \quad (5.5)$$

5.2.2 Metoda požadovaného modelu (metoda inverze dynamiky)

Určení parametrů PID výpočtem z aproximačního modelu (tlumená kmitavá soustava s dopravním zpožděním D) tak, aby přenos řízení splňoval požadovaný tvar

Princip: volitelný parametr α od $1/e/D$ do $\pi/2/D$ vyjadřuje relativní překmit κ od 0% do 107% (trvalé kmity) s různou periodou kmitů. (Dušek, 2023)

Stavitelné parametry PID regulátorů pro metodu požadovaného modelu:

$$G_s(s) = \frac{Z e^{-Ds}}{T^2 s^2 + 2ETs + 1} \quad (5.6)$$

$$r_0 = \frac{\alpha^* 2ET}{DZ} \quad (5.7)$$

$$T_i = \frac{\alpha^* 2ET}{DZ} \quad (5.8)$$

$$T_d = \frac{T}{2E} \quad (5.9)$$

Tabulka 5.3 – Relativní překmit κ a jemu odpovídající volitelný parametr α pro MPM

κ	$\alpha^* = \alpha D$
0%	0,3679
5%	0,5206
10%	0,5944
15%	0,6378
20%	0,6881
25%	0,7446
30%	0,7996
35%	0,8505
40%	0,8999
45%	0,9499
50%	1,002

6 PRAKTICKÁ ČÁST

Obecně lze postup praktické části popsat takto:

Nejprve byl v LabView vytvořen systém pro ukládání dat chování zařízení QNET 2.0 VTOL Board. Dále se provedou skokové změny aplikovaného napětí $u(t)$ a do textového souboru se zaznamená příslušná reakce systému ve formě změny výstupní hodnoty $w(t)$ a času t odpovídající každému změna dat.

V programu Matlab je vytvořen skript, který aproximuje zaznamenaná data reálného systému, jako výsledek jeho práce přenosová funkce, která přibližně popisuje chování reálného zařízení.

To umožňuje pracovat s virtuálním systémem, jehož chování se přibližně – co možná nejvíce blíží chování skutečného, provádět výpočty pro úpravu parametrů regulátoru a také simulovat jeho chování.

Výsledný matematický model systému slouží k nalezení parametrů regulátoru, které by mohly co neoptimálněji regulovat reálný systém, pomocí několika metod nastavování PID regulátoru, pro které se používají různé metody: parametry v procesu re-aproximace k popisu chování systému jiným způsobem, v prostředí MATLAB.

V důsledku výše popsaného zpracování dat, jakož i podrobněji níže, jsou získány parametry všech regulátorů slozek. Každá použitá metoda vede k vlastním parametrům a pro matematický popis kvality regulace při aplikaci různých parametrů v prostředí Simulink slouží simulace chování přibližného systému řízeného PID regulátorem s různými parametry získanými aplikací různých metod. používá se získávání parametrů, K tomu se používají integrální kritéria kvality regulace, která umožňují získat matematickou charakteristiku. To umožňuje určit pravděpodobně/teoreticky nejlepší variantu nastavení regulátoru.

Nalezené parametry jsou nakonec použity v regulátoru, který reguluje reálný systém.

6.1 POUŽITÉ DATOVÉ TYPY

Nejčastěji používané datové dráty a jejich odpovídající datové typy:

- Fialová – kanál úloh «Task», nastavuje/definuje pořadí provádění svých funkcí po blocích.
- Oranžová – digitální (numerický, DBL - double) datový tok. Přenáší informace o fyzikálních hodnotách veličin: nastavené minimum, maximum, parametry napětí, požadovaná hodnota, úhel natočení a další fyzikální veličiny z toho vypočítané. Má číselný typ s hodnotou mezi čárkou, to znamená, že číslo nesmí být celé číslo a může to být zlomek.

- Zelená – booleovský datový tok. Přenáší hodnotu «0» nebo «1» a používá se k přepínání vnitřní logiky programu a externího zařízení mezi režimy zapnuto a vypnuto. Přepíná elektromagnety, ukončuje simulační smyčku, nastavuje počáteční hodnoty pro porty.

Některé datové dráty jiných typů:

- Růžová – řetězec (string), udává údaje o počtu a typu portů, zřejmě nebo řádků. Používá se také k ukládání hodnot do textového souboru.
- Modrá – číselný datový typ Int (integer) - akceptuje pouze celočíselné hodnoty a používá se k označení čísel nebo počtu portů nebo pulzů na otáčkoměřů. Podle velikosti ho dělíme, na 64b, 32b, 16b a 8b.
- Světle modrá – nastavuje cestu pro ukládání dat do textového souboru.
- Hnědá – waveform, speciální datový typ obsahující data z časové oblasti, obsahuje tyto položky:
 - t_0 – čas začátku průběhu
 - dt – časový interval mezi jednotlivými body průběhu
 - Y – waveform data
- shluk (cluster), což je proměnná obsahující další proměnné. Používá se k zobrazení dat v grafu křivek a nastavení parametrů pro PID regulátor.

6.2 POPIS HLAVNÍHO VI SYSTÉMU

V LabVIEW se systémy nazývají «VI» a subsystémy se nazývají «subVI».

Počáteční hlavní VI obsahuje 2 subVI:

Encoder Init – inicializuje enkodéry, v tomto systému jsou to 2 rotační enkodéry. V počátečním VI je inicializace čtení dat kodéru implementována pomocí dvou subVI se specifickým portem specifikovaným pro každý kodér.

Digital Channels Init – umožňuje interakci s výstupy a změnu dat pro ovládání zařízení jako skutečného systému. Inicializuje oba kanály pro dva elektromagnety.

Celý ovládací program lze rozdělit na 3 části, z nichž každou lze rozdělit na několik dalších komponent:

1. Úvodní část

Inicializuje systémové vstupy/výstupy, nastavuje data portu pro interakci s komponentami zařízení. Jeho funkcí je připojit přístup k interakci s různými částmi zařízení. Spustí se, když je povoleno VI.

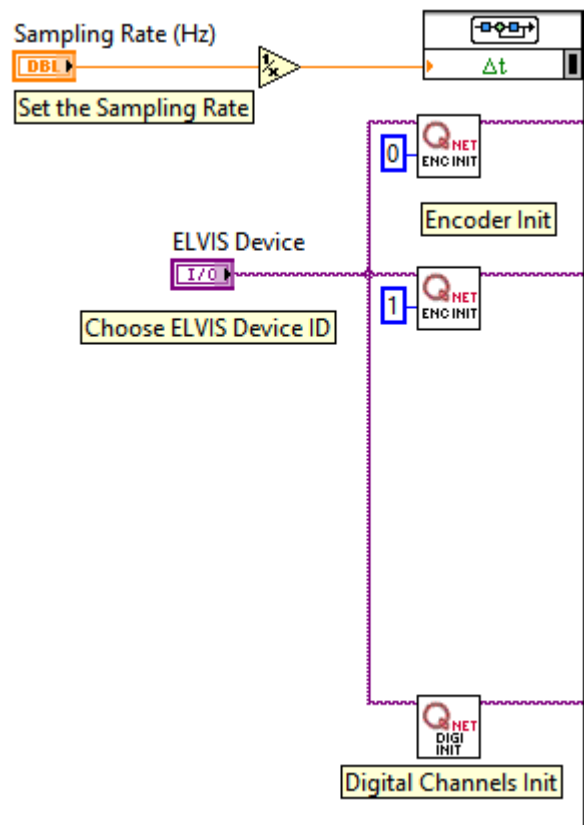
2. Hlavní část.

Obsahuje všechny funkce programu.

Funguje během celé činnosti programu, cyklicky prochází interakcemi mezi bloky a aktualizuje data mezi bloky umístěnými uvnitř struktury řídicí a simulační smyčky.

3. Závěrečná část.

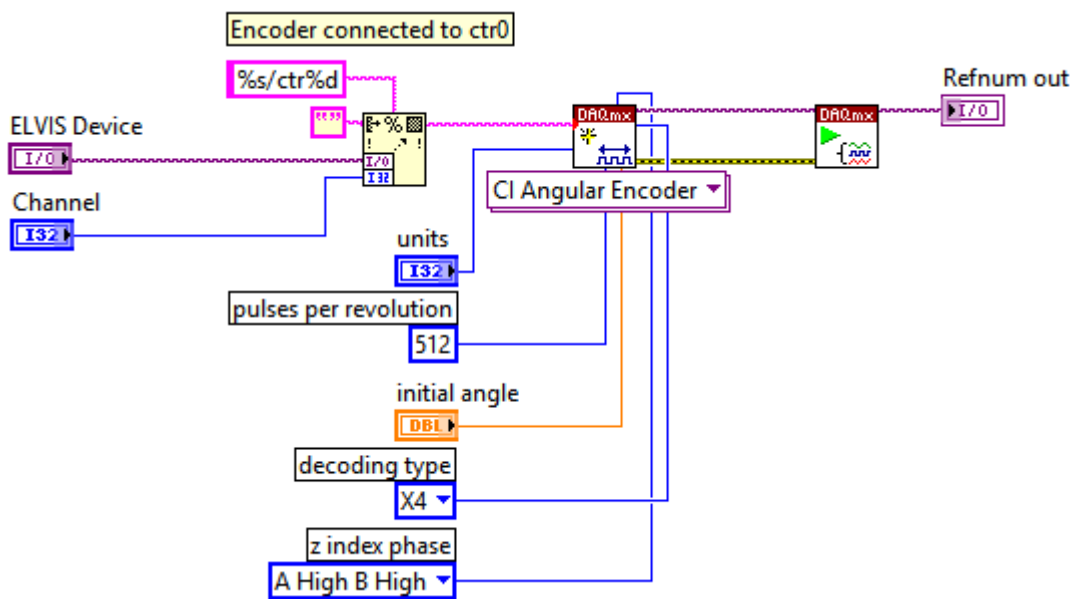
Ukončí hardware, když se program ukončí, a uvede fyzický hardware a program do daného/nastaveného stavu. To obvykle znamená přípravu hardwaru pro další použití: zapnutí nebo vypnutí hardwarových položek, nastavení parametrů spouštění, vymazání dat z minulého použití programu.



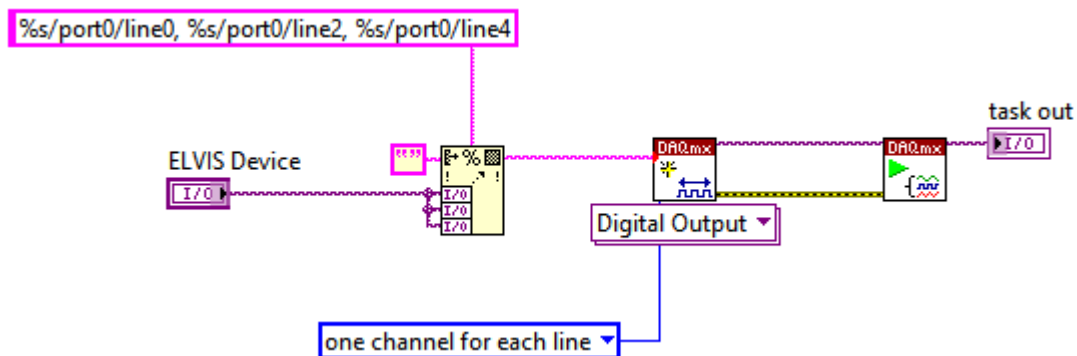
Obrázek 6.1 – Inicializační část programu

Inicializační část programu se skládá z:

1. Choose ELVIS Device ID – výběr zařízení NI ELVIS, se kterým chceme pracovat. K počítači lze připojit několik zařízení a měřicích karet, takže je třeba vybrat to, které potřebujeme.
2. Set the Sampling Rate – nastavuje frekvenci provádění hlavní smyčky programu Control & Simulation Loop.
3. Encoder Init – 2 SubVI inicializace enkodérů, které měří úhel natočení – každý SubVI inicializuje svůj vlastní enkodér, u kterého se externě nastavuje informace o portu.
4. Digital Channels Init – inicializace napájení elektromagnetů. Oba porty jsou inicializovány na stejném subVI.



Obrázek 6.2 – SubVI inicializace enkodéru



Obrázek 6.3 – SubVI inicializace řízení elektromagnetů

Popis hlavních bloků první části programu, který inicializuje vstupní a výstupní porty systému:

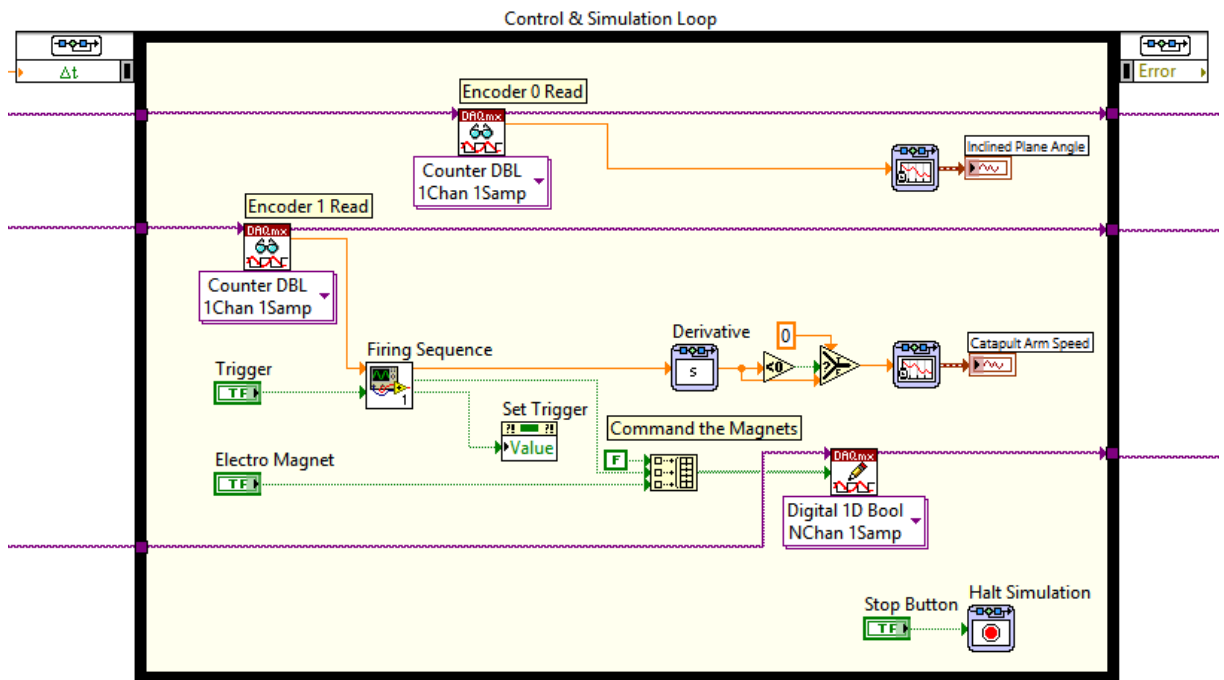
1. DAQmx Create Channel – blok představující konfiguraci inicializovaného portu a zařízení s ním spojené. Definuje typ a vlastnosti připojeného zařízení a také nastavuje číslo portu, ke kterému je připojeno - u kódérů se tato informace nastavuje z vnějšku subVI.

CI Angular Encoder – inicializuje vstupní porty, které umožňují přijímat informace z obou enkodérů.

Digital Output – inicializuje ovládání obou elektromagnetů přepnutím napájení do nich.

2. DAQmx Start Task – je blok, který inicializuje Task, informace o něm jsou uvedeny v bloku DAQmx Create Channel. Umožňuje dále provádět operace se zařízením inicializovaným pomocí bloku Start Task.

3. Zbývající bloky v subVI – jsou proměnné typu Integer, Double a String, které specifikují informace pro virtuální zařízení, aby mohlo interagovat se skutečným zařízením.



Obrázek 6.4 – Prováděcí část programu

Popis bloků druhé části spouštěcího programu, cyklicky spouštějícího kód, který řídí systém a aktualizuje data přijatá ze senzorů.

Skládá se z částí:

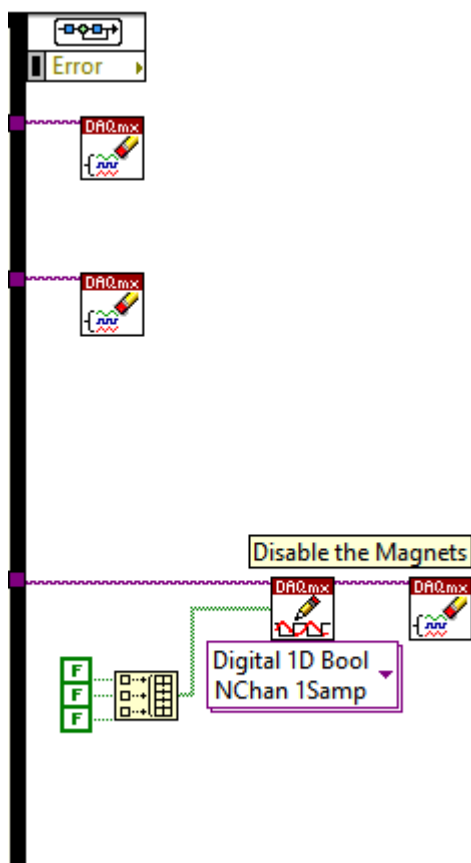
1. Encoder 0 Read and Inclined Plane Angle – větev, která přijímá data o aktuálním náklonu z enkodéru instalovaného na nakloněné rovině a zobrazuje je v grafu na předním panelu.
2. Firing Sequence and Catapult Arm Speed – podobná větev pro druhý enkodér, který měří úhel natočení katapultu, doplněná o několik logických bloků a blok Derivative pro výpočet rychlosti katapultu ve stupních za sekundu.
3. Command the Magnets – pracuje s Digital Output Channels, přes které umožňuje přepínat napájení obou elektromagnetů stisknutím tlačítek na předním panelu.
4. Halt Simulation – nejedná se o trvale prováděnou operaci ve smyčce, umožňuje ukončit provádění smyčky stisknutím tlačítka «Stop» na předním panelu a přejít k provádění závěrečné části programu.

Použité bloky:

1. DAQmx Read (Counter DBL 1Chan 1Samp) – na základě přivedeného Task signálu čte data ze zařízení ELVIS a převádí datový tok do formátu Double
2. Simulation Time Waveform. Slouží k transformaci dat z dvojitého typu na signál pólového průběhu, přidáním vhodného časového měřítka k měřenému datovému toku.
3. Waveform chart. Následně se data přenesou do jednotky výstupu dat pro uživatele zobrazením grafu na předním panelu.
4. DAQmx Write (Digital 1D Bool NChan 1Samp – funguje podobně jako DAQmx Read, ale místo čtení dat ze zařízení do něj naopak přenáší data, přiváděná přes VI z tlačítek na předním panelu Slouží k napsání příkazu pro zapnutí nebo vypnutí obou elektromagnetů každého zvlášť. Do jednoho bloku je prostřednictvím jednoho datového toku přiváděna informace o stavu dvou tlačítek současně vložením ovládacího signálu do 2D pole.
5. Firing Sequence. SubVI zodpovědný za spínací logiku elektromagnetu katapultu nastavením spouště.
6. Další bloky:
 1. Trigger, Electro Magnet, Stop Button – tlačítka na předním panelu, která umožňují ovládání systému.
 2. Set Trigger – Nastavuje hodnotu stavu spouštění použitou pro spouštěcí sekvenci.
 3. Command the Magnets - Kombinuje jednotlivé proměnné z každého tlačítka v poli.
7. Halt Simulation. Blok, který ukončí simulační smyčku, když je mu přidělena vhodná booleovská hodnota.

Program lze také ukončit tlačítkem «Abort Execution» umístěným na horní liště LabVIEW, ale tím se přeruší provádění programu ve fázi, ve které byl stisknut, jako v případě přerušení v případě chyby. Při ukončení programu pomocí tlačítka ukončení na předním panelu, které iniciuje konec simulační smyčky, aplikace provede závěrečnou část programu – bloky v řetězcích umístěných za Control & Simulation Loop.

To umožní ukončit provádění programu, připravit zařízení a aplikaci pro další použití: vymazat data ze vstupních kanálů a grafiky odpovídající každému kanálu a vypnout elektromagnety.



Obrázek 6.5 – Závěrečná část programu

Popis bloků třetí části programu, který dokončuje práci se zařízením:

1. DAQmx Clear Task – vymaže Task a uvolní kanál. Dokončuje práci se všemi jednotkami QNET Physics and Dynamics Board.
2. DAQmx Write (Digital 1D Bool NChan 1Samp) – zapisuje booleovské hodnoty 0 pro elektromagnety a vypíná je v posledním cyklu Task.
3. Build Array and Boolean Constants – nastaví logickou nulu na kanálech elektromagnetů a vypne je před ukončením programu.

6.3 POPIS KONSTRUKCE A ČINNOSTI ZAŘÍZENÍ

Zařízení má 4 hlavní prvky, se kterými lze komunikovat prostřednictvím LabVIEW:

- 2 elektromagnety
- 2 enkodéry

Kanál 0 odpovídá elektromagnetu a enkodéru souvisejícímu se sliding mass (posuvná hmota), kanál 1 odpovídá elektromagnetu a enkodéru katapultu.

V LabVIEW existuje několik kategorií typů kanálů:

Vstup/výstup:

- Vstup – kanály, které měří data ze zařízení, v případě QNET Physics and Dynamics Board se jedná o měření úhlu natočení.
- Výstup – kanály, které řídí vstupní hodnoty dodávané do zařízení, obvykle se jedná o generování napětí.

Typ kanálu:

- Digital (Digitální) – Používá booleovskou proměnnou, obvykle ke čtení stavu nebo k zapnutí nebo vypnutí jednotky.
- Analog (Analogový) – definuje datový tok číselného typu, který umožňuje změnit požadovaný parametr.
- Counter – čítač, čte data.

Kodéry jsou inicializovány vstupem čítače. To umožňuje sledovat v reálném čase změny v údajích z kodéru, získaných s určitou frekvencí v závislosti na použitém kodéru.

Elektromagnety jsou inicializovány pomocí digitálního výstupu. To umožňuje zapínat a vypínat napájení elektromagnetu.

6.3.1 Volba modulu a fyzikálního experimentu pro řízení

Nejzřejmější varianta ovládní se jeví/vypadá jako změna přivedeného napětí na elektromagnet na kanálu 0 pomocí posuvné hmoty – magnet by tedy mohl držet zátěž v určité vzdálenosti, vytáhnout ji nebo naopak uvolnit to, což mu umožní klouzat dále. Enkodér čte informace a tak lze buď přímo měnit natočení ramene kyvadla měřené enkodérem, nebo pomocí těchto dat dále vypočítat rychlost, zrychlení, případně vzdálenost zátěže, nebo jiné parametry na nich založené.

Jako příklad si vezmeme regulaci, která by udržela «klouzavou hmotu» v určité vzdálenosti od magnetu a umožnila by ji přitahovat nebo nechat sklouznout do určité vzdálenosti od magnetu. Další možností je držení určité hodnoty na kodéru, jako pokračování vývoje této možnosti – držení obráceného kyvadla ve stabilním stavu.

6.3.2 Možné způsoby regulace a problémy při implementaci regulace

Zde vyvstává řada technických a fyzických problémů implementace regulace:

1. Je technicky nemožné změnit napětí přes virtuální analogový kanál v LabVIEW VI na elektromagnetu, podporuje pouze digitální kanál, který umožňuje zapínat a vypínat elektromagnet.

To je ve skutečnosti může být řešeno pulzním regulátorem typu PWM, který pulzuje vysokofrekvenční napájecí zdroj v krátkých intervalech.

2. Elektromagnet ovlivňuje objekt regulace, který je zároveň i řídicí částí systému, teoreticky regulaci umožňuje. Vzhledem k tomu, že posuvná hmota se musí oddálit od ovládacího členu, aby ovlivnila kodér, vytváří to fyzická omezení pro ovládání - posuvná hmota se může od elektromagnetu vzdálit na malou vzdálenost, jinak se dostane mimo elektromagnetické pole a jakékoli regulace nebo ovládání nebo obecně vliv na systém se stává nemožným.

3. Potenciálním řešením by bylo použití pružiny místo kluzného bloku – v tomto případě mechanický člen, na který působí elektromagnet, současně zůstává pod vlivem magnetického pole, ale jeho natažení umožňuje měnit parametry systému a hodnotu na enkodéru. V tomto případě může magnet přitáhnout pružinu k sobě a postupně ji stlačit.

V QNET Physics and Dynamics Board je však upevnění pružiny technicky implementováno tak, že jej nelze připevnit k magnetu nebo před magnetem – k tomu je elektromagnet odstraněn a pružina je připevněna k desce místo k magnetu.

4. Žádné z těchto řešení, i když jsou učiněny určité předpoklady, není pro toto zařízení proveditelné, alespoň se standardními elektromagnety používanými v QNET Physics and Dynamics Board, protože oblast působení jejich magnetického pole je extrémně malá – cca 1 až 2 mm. I když hypoteticky budeme uvažovat o regulaci tohoto systému, má to určité dale popsána specifika.

6.3.3 Fyzikální specifika chování zařízení

Kolečko, kterým prochází kovové lanko, je na jedné straně připojeno k enkodéru a na druhé straně je k němu bod pro připevnění nějaké hmoty jako kyvadlo. Poněkud problematické je nastavit systém tak, aby od začátku umožňoval pomalý pohyb klouzající hmoty – vzhledem k tomu, že ve standardním klidovém stavu je kyvadlo spuštěno dolů, když je magnet vypnutý. Kolečko se velmi rychle otočí o cca 13 stupňů, poté zůstává v jedné poloze a vytváří určitou odporovou sílu v důsledku tření lanka, zpočátku však kvůli jeho nepřítomnosti a rolování

kolečka klouzající hmota má výrazné zrychlení, což vede k tomu, že toto zrychlení trhavě posouvá pohybující se hmotu za oblast působení elektromagnetického pole a působí proti síle přitažlivosti magnetického pole v této fázi, zatímco hmota je stále v oblasti působení magnetu.

6.3.4 Rozhodnutí použít k ovládní jiné zařízení

Elektromagnety nepracují s kanálem typu Analog Output – programově je možné takový Output inicializovat pro práci s napětím (na VI je označen jako AO Voltage), nastavit jej tak, aby pracoval s požadovaným kanálem a vytvořit řídicí obvod s DAQmx Write typu Analog DBL 1Chan 1Samp, který by měl měnit napětí přiváděné do elektromagnetu, a k němu připojený blok, který nastavuje hodnotu požadovaného napětí pomocí proměnné číselného typu, technicky však, vybavení Quanser QNET Physics and Dynamics Board nepodporuje schopnost měnit napětí dodávané do elektromagnetů pro řízení jejich generovaného elektromagnetického pole.

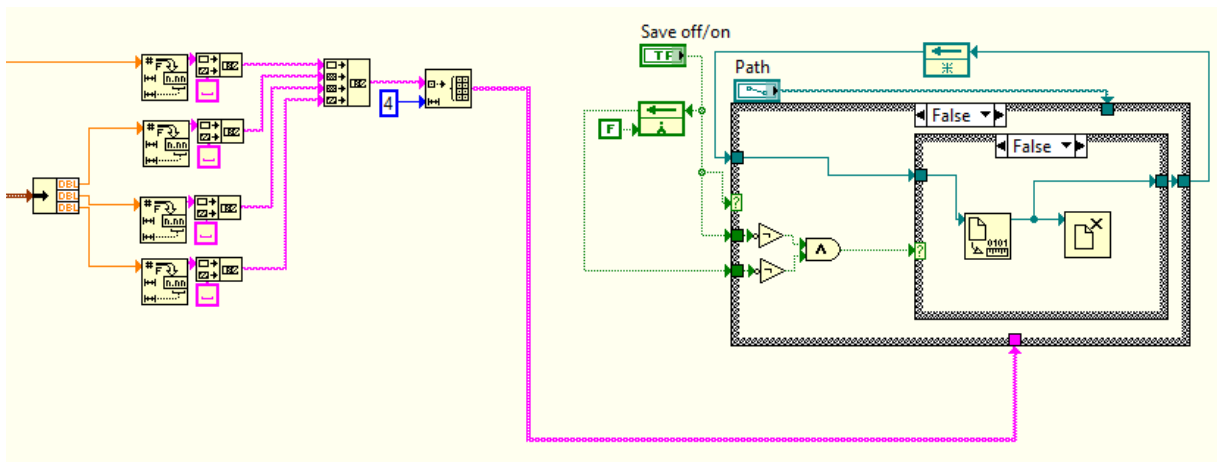
Tyto informace byly získány jak při praktické práci s Quanser QNET Physics and Dynamics Board a Quanser QNET Rotary Inverted Pendulum Board 2.0 pro laboratorní zařízení NI ELVIS II/II+ a testováním implementovaných a startovací «Quick Start» VI na nich (byly testovány střídavě VI implementované na obou zařízeních a VI poskytnuté na jeho zdrojích pro počáteční práci od NI) a byla obdržena přímá odpověď o nemožnosti změny napětí pomocí LabVIEW od technické podpory společnosti Quanser (Consulting Inc), která vyrábí toto zařízení a jeho údržbu.

7 DOPLŇKOVÉ A REGULAČNÍ VI

Hlavní výkonná část je doplněna o následující moduly jak pro implementaci regulace, tak pro ukládání dat, která mohou být následně použita pro nastavení regulatoru, a pro další účely.

7.1 UKLÁDÁNÍ DAT

Modul pro ukládání dat. Implementováno dvěma strukturami Case, které ukládají datový proud do textového souboru. Všechna data jsou v každém cyklu zapsána do řádku, poté jsou zapsána do dalšího řádku v dalším cyklu, čímž se vytvoří sloupce s daty pro každou jednotlivou hodnotu.



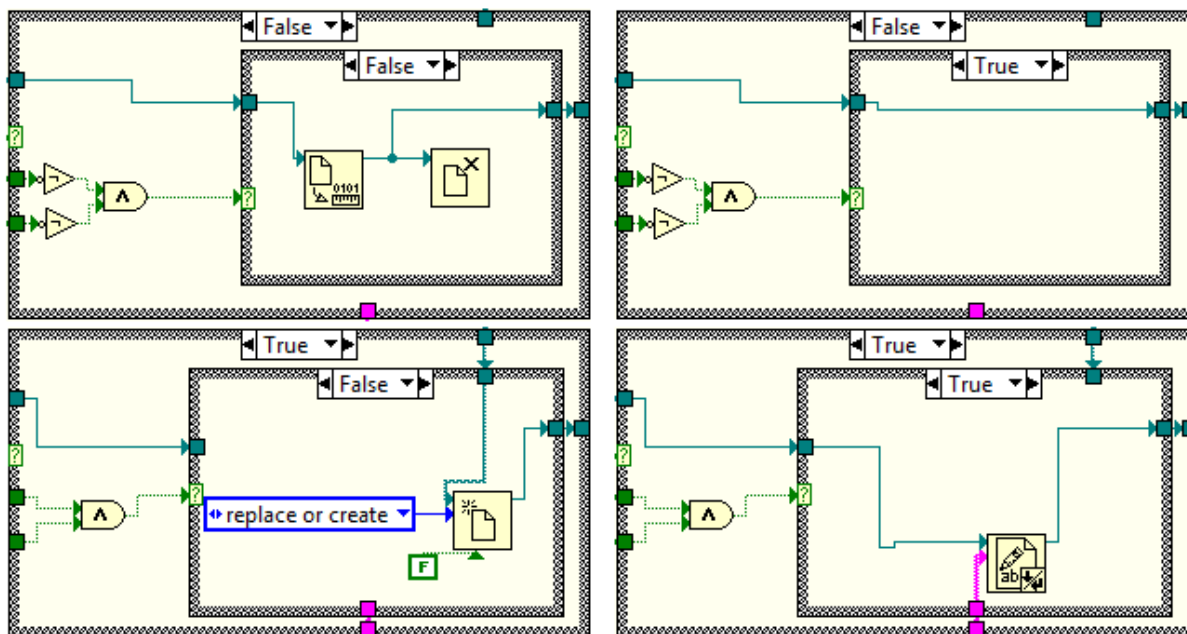
Obrázek 7.1 – Systém ukládání dat

Systém ukládání dat se skládá ze dvou hlavních částí:

- Zpracování toků dat a jejich převod na datový typ vhodný pro ukládání.
- Ukládání dat do souboru.

7.1.2 Část pro ukládání dat

Po stisknutí tlačítka se začnou zapisovat data do textového souboru podél cesty nastavené uživatelem.



Obrázek 7.3 – Cykly ukládání dat

Popis základních prvků ukládání dat:

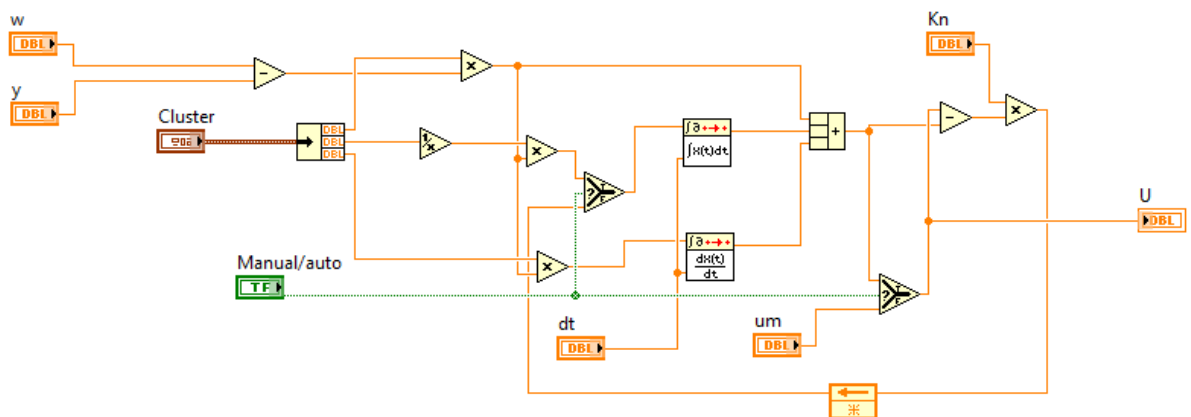
1. Open/Create/Replace File – blok, který vytváří soubor ve složce, ke které má nadefinovanou cestu.
2. Replace or create constant – poskytuje bloku Open/Create/Replace File informaci, co přesně má se souborem na zadané cestě udělat.
3. Write to Text File – zapíše přijatá data do zadaného textového souboru.
4. Get Size File – blok, který zjišťuje velikost souboru a je předřazen bloku pro ukončení ukládání.
5. Close Binary File – uzavření souboru.
6. Feedback Node – blok, který zpožďuje přivedený signál o 1 hodnotu. Umožňuje připojit výstup blokového řetězce k vlastnímu vstupu.

Při ukládání je cesta pro vytvoření a zápis souboru pro ukládání dat specifikována uživatelem ve speciálním řádku na předním panelu. Po spuštění skriptu se vytvoří soubor se zadaným názvem, nebo pokud již existuje, bude nahrazen novým.

Díky tomu, že jsou data uložena v textovém souboru, lze je v MATLABu volně používat bez nutnosti překladu do zpracovatelské podoby a dalších operací s datovým souborem, jako je převod souboru z binárního na textový.

7.2 PID REGULÁTOR

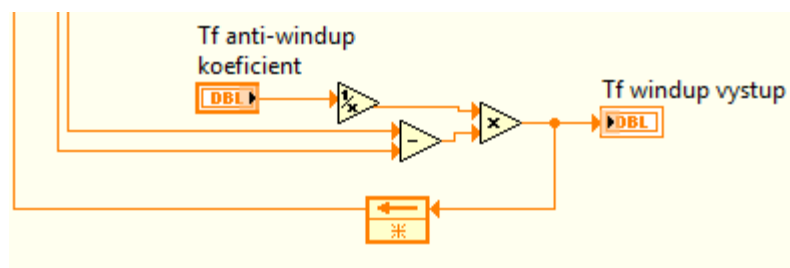
PID regulátor v bloku subVI je diskretní PID regulátor s beznázorovým přepínáním mezi automatickou a manuální regulací napětí pro ovlivnění ovládacího prvku systému. V podstatě se jedná o strukturu běžného PID regulátoru.



Obrázek 7.4 – PID regulátor s beznázorovým přepínáním

Popis základních prvků PID regulátoru:

1. Cluster - obsahuje hodnoty složek P, I, D.
2. Manuální / auto - umožňuje přepínat mezi ručním nastavením přímého napětí a režimem automatické regulace pomocí regulátoru, ve kterém uživatel nastaví požadovanou hodnotu.
3. Select - připojeno k tlačítku, přijímá booleovskou proměnnou a dvě numerické, vypisuje hodnotu numerické proměnné odpovídající nastavené booleovské hodnotě. Provádí logické přepínání mezi režimy ovládání.
4. Integral, Derivative - speciální bloky jako integrátor a derivátor ve standardním zapojení PID regulátoru. Od běžných bloků se liší nutností připojení k portu také parametrem vzorkovací frekvence dt . Frekvence aplikovaná na tyto bloky musí odpovídat frekvenci nastavené pro řídicí a simulační smyčku, jinak PID vypočítá nesprávné parametry kvůli nesouladu frekvence s PID zařízením.



Obrázek 7.6 – Doplnění anti-windup modifikace PID regulátoru

K realizaci je nutné použít, stejně jako v případě realizace cyklu pro ukládání dat, Feedback Node, jelikož bez něj je v LabVIEW technicky možné připojit výstup dalšího bloku v blokovém řetězci na vstup jednoho z předchozích, ale to způsobí softwarovou chybu a program nebude fungovat.

Při nastavení parametrů regulátoru výrazně odlišných od optimálních může systém začít nekontrolovatelně kolísat v důsledku výpočtu obrovských hodnot napětí, výrazně mimo pracovní rozsah. Doplnění anti-windup modifikace nezaručuje stabilizaci chodu samotného systému, ale umožňuje regulátoru rychle se vrátit do požadovaného rozsahu hodnot, omezeného jím ovládaným zařízením, a pokračovat v regulaci. Bez něj může návrat do tohoto rozsahu trvat značnou dobu.

Při použití windup modifikace regulátoru je také značně zjednodušeno ruční ladění jeho parametrů, neboť v případě klasické implementace PID regulátoru může být jeho výstupní vypočtená hodnota výrazně větší či menší než rozsah, ve kterém zařízení funguje, a to vede k problémům, jako jsou extrémně velké překmity a výstupní hodnoty a dlouhá doba pro návrat hodnot do provozního rozsahu na krátkou dobu.

7.3.2 Problémy použití anti-windup modifikace

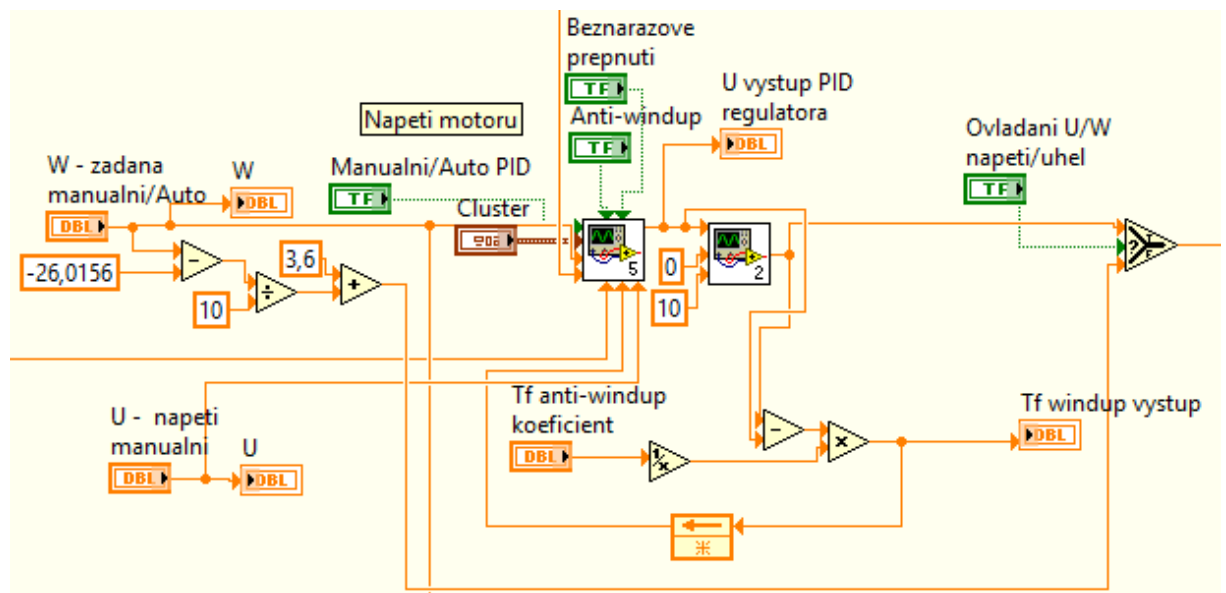
V LabVIEW může extrémně vysoký překmit způsobit chyby ve výpočtu hodnot při použití modifikace proti navíjení – pravděpodobně to může být způsobeno buď příliš velkými hodnotami generovanými modifikací a nepodporovanými datovým typem použitým k ukládání. vypočítané hodnoty pomocí LabVIEW nebo se vyskytují v kombinaci s normálním integrálním členem, ale program je náchylný k chybám, pokud je vliv integrálního členu regulátoru příliš velký. Problém zřejmě nastává právě při výpočtu hodnot: v tomto případě může být hodnota zobrazená na regulátoru zobrazena jako NaN – hodnota s plovoucí

desetinnou čárkou Not-a-Number, ke které dochází při provedení matematicky neplatné operace, jako například je dělení nulou nebo výpočet druhé odmocniny záporné hodnoty. V případě takové chyby je nutné otevřít subVI regulátoru a přepočítat hodnoty jednorázovým spuštěním podprogramu - provede pouze jeden výpočetní cyklus a úpravou bude moci opět vypočítat a přidat součást proti zavětrování.

7.4 KONEČNÉ SCHÉMA PID REGULÁTORU A OVLÁDÁNÍ

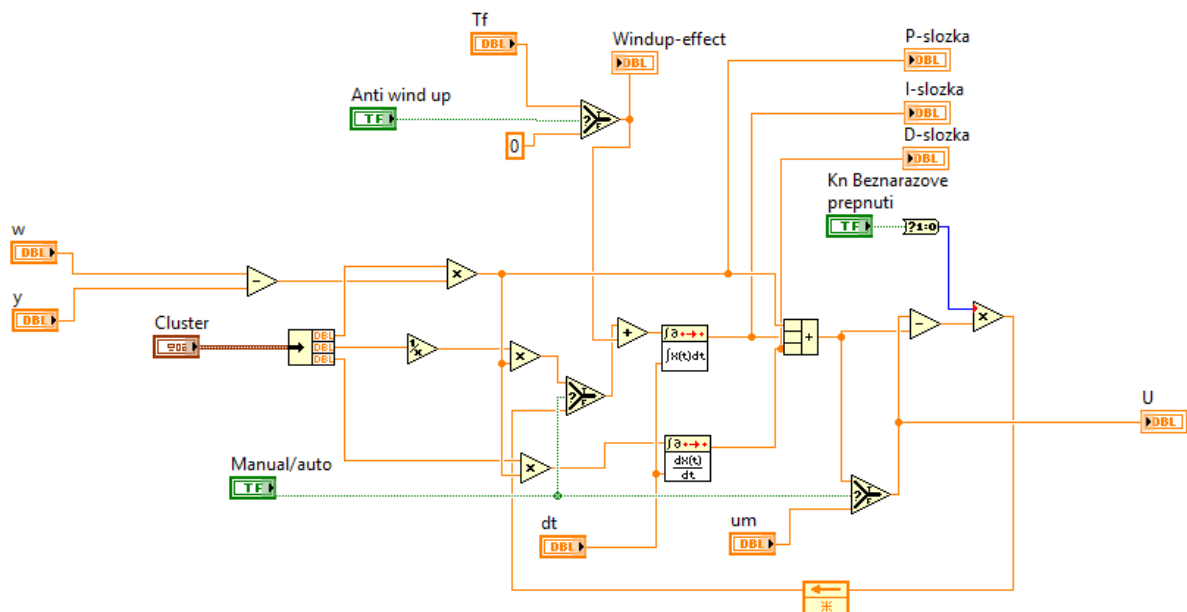
Finální verze je zapojena sériově: blok subVI PID s beznárazovým přepínáním, na něj navazující blok subVI Saturace, a také anti-windup modifikace umístěná na hlavním VI mimo subsystemy, spojující výstup saturačního bloku s regulátorem přes Feedback Node.

Obvod ovládání umožňuje ovládat napětí nebo úhel natočení bez použití regulátoru.

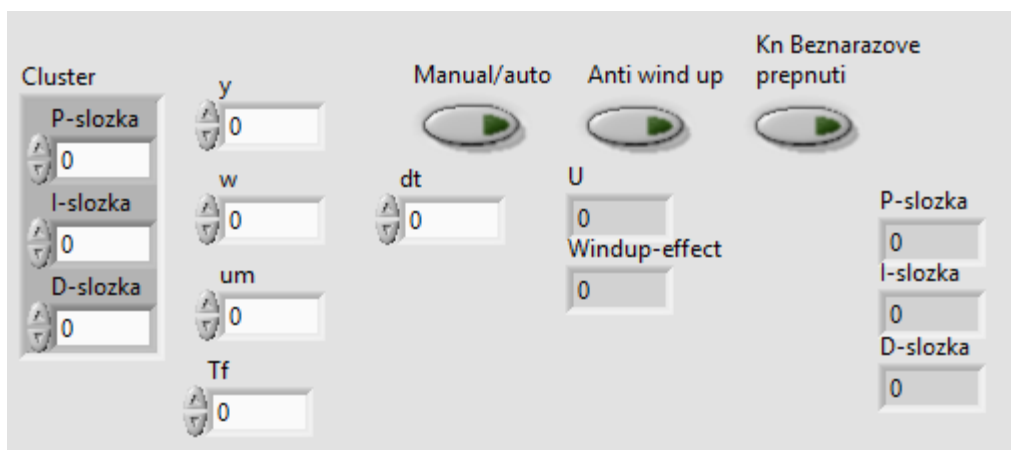


Obrázek 7.7 – Finální schéma PID regulátoru

SubVI PID bylo také poněkud změněno: byla přidána funkce pro implementaci windup efektu a přepínání jeho činnosti, stejně jako zobrazení aktuálního vlivu na vypočítané napětí z každé součásti regulátoru.

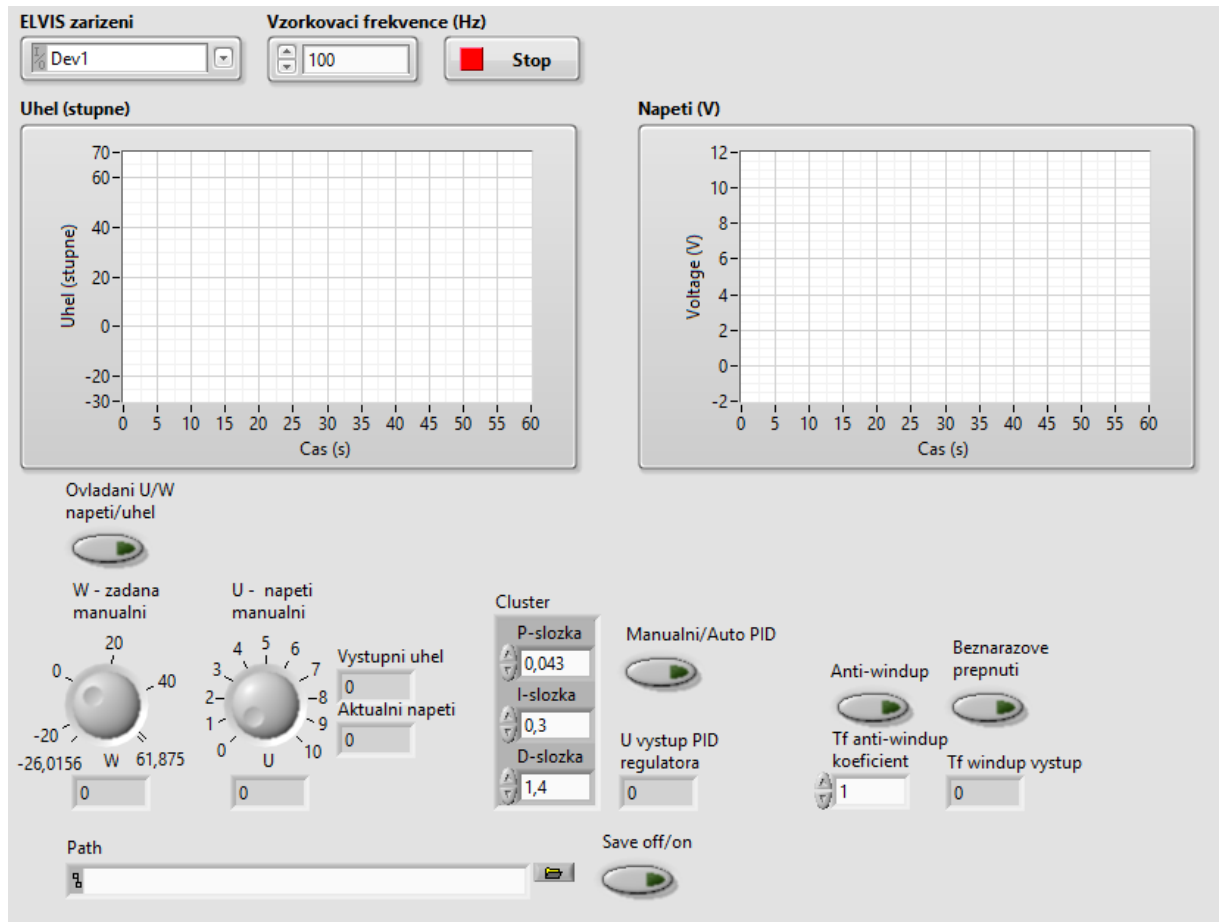


Obrázek 7.8 – Finální schéma subVI PID



Obrázek 7.9 – Finální čelni panel subVI PID

7.5 FINÁLNÍ VERZE PŘEDNÍHO PANELU



Obrázek 7.10 – Finální čelní panel systému Quanser QNET 2.0 VTOL Board

Konečná verze programového panelu vám umožňuje:

Nastavení frekvenci provádění regulační smyčky.

Změna napájecí napětí ventilátoru přímo.

Nastavení požadovanou hodnotu $w(t)$ v lineárním rozsahu provozu zařízení bez použití regulátoru.

Nastavení hodnoty P, I, D slozek PID regulátoru.

Přepínání ovládání mezi ručním ovládáním napětí $u(t)$ a úhlem natočení $w(t)$.

Přepínání mezi ručním a automatickým ovládáním pomocí knoflíku.

Nastavení anti-windup koeficientu a povolení anti-windup modifikaci a beznárazové přepínání

Ukládání dat do textového souboru na vlastní cestu, s možností spuštění a ukončení ukládání před nebo během provádění programu stisknutím tlačítka.

Zobrazuje hodnoty:

Základní: nastavení napětí, nastavený úhel natočení, zadana hodnota $w(t)$, aktuální napětí.

Dodatečné: výstupní (vypočtená) hodnota napětí PID, nastavení hodnoty anti-windup koeficientu a vlivu anti-windup modifikace regulátoru.

Stav booleovských proměnných: typ ručního ovládání, ruční / automatické ovládání a zahrnutí anti-windup úpravy a beznárazového přepínání, lze určit podle vzhledu tlačítka – v případě přepnutí se na něm rozsvítí zelená kontrolka.

Zobrazuje průběh výstupní hodnoty $y(t)$ a napětí $u(t)$.

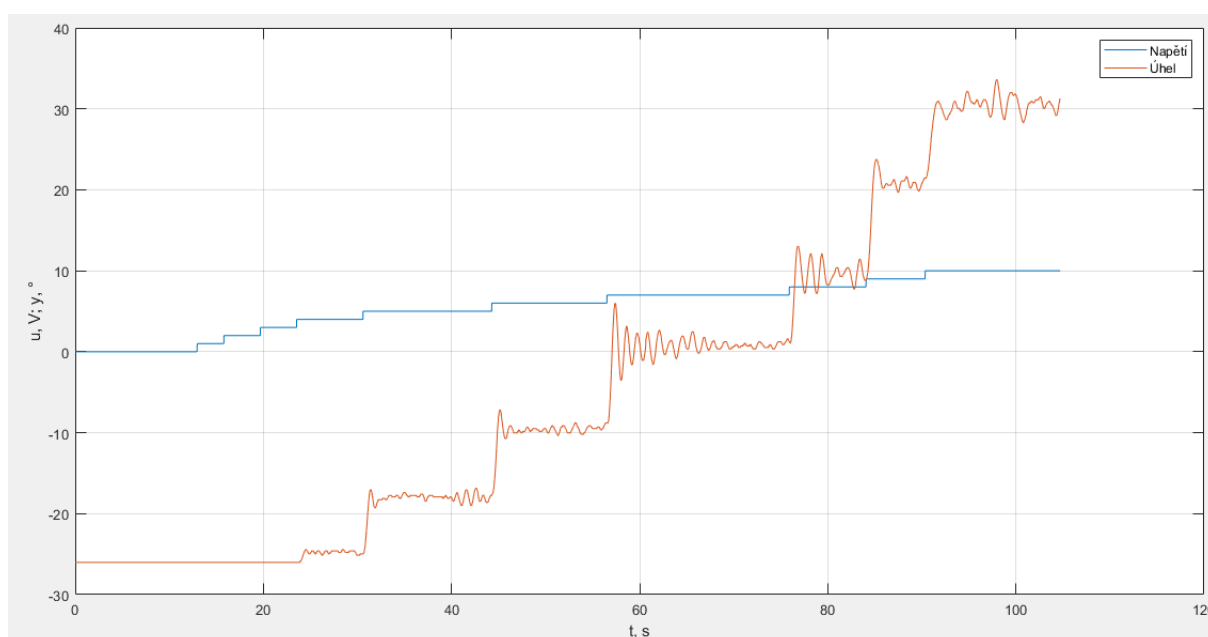
PID subVI také zobrazuje úroveň vlivu proudu každého ze složek na konečnou vypočítanou výstupní hodnotu regulátoru.

8 EXPERIMENTÁLNÍ IDENTIFIKACE SOUSTAVY

8.1 POSLOUPNOST PRÁCE A MĚŘENÍ DAT

Nejprve byla změřena odezva systému a změna úhlu natočení v závislosti na napětí aplikovaném na ventilátor a data byla zapsána do textového souboru. Napětí se měnilo od 0 do 10 voltů v krocích po 1 voltu. Dále byl v prostředí MATLAB použit skript pro zpracování dat – podle napětí zaznamenaného v textovém souboru, odpovídajícímu úhlu natočení na kodéru, a času, který umožňuje jak vytvořit časovou osu X na grafu. a ukázat a sledovat dynamiku procesu.

8.2 PŘECHODOVÁ CHARAKTERISTIKA

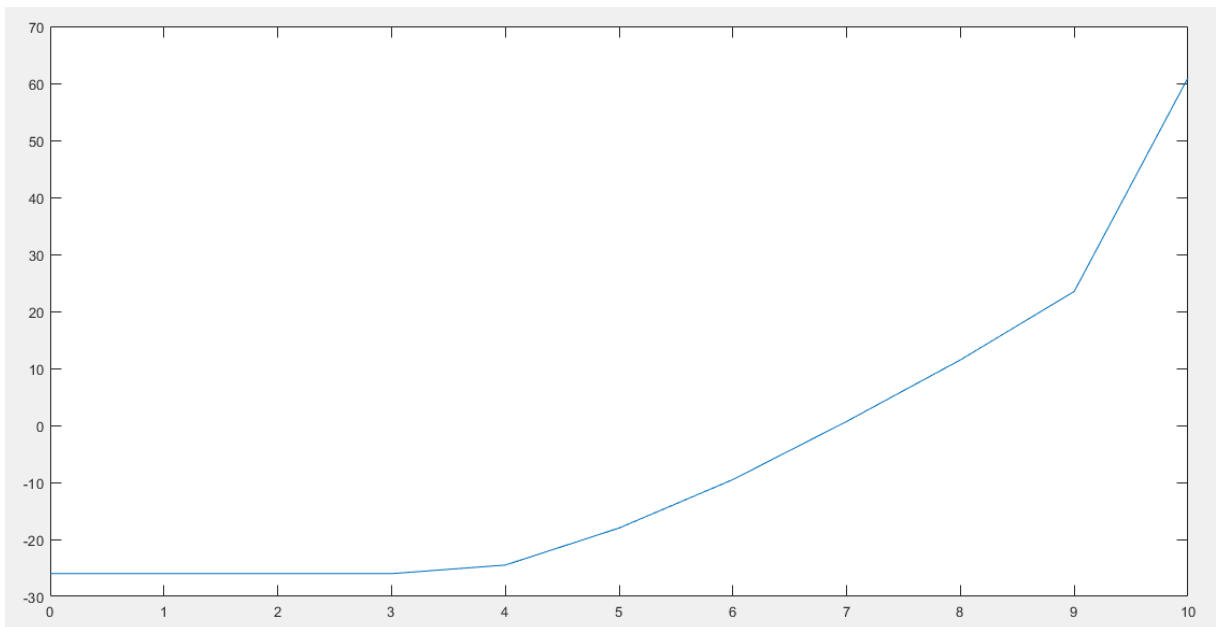


Obrázek 8.1 – Přechodová charakteristika

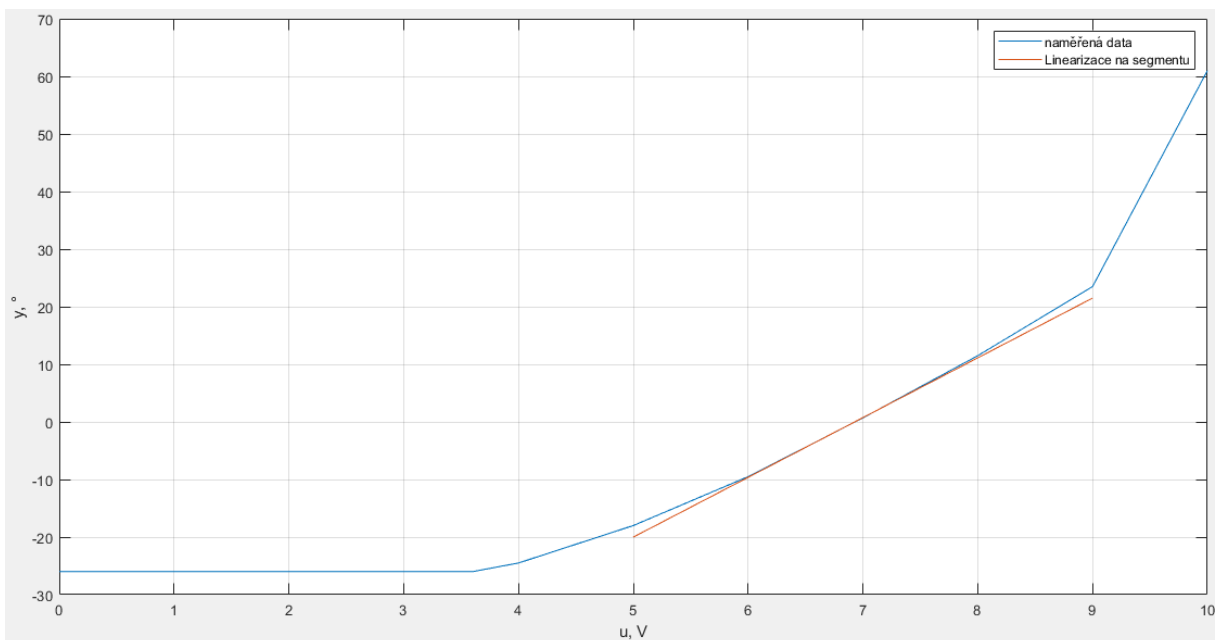
Z grafu je patrné, že zařízení se vyznačuje rychlou odezvou na změnu hodnoty vstupního napětí U , a výrazným překročením regulované hodnoty jeho normální hodnoty s rostoucím napětím, jakož i stálým netlumené oscilace, které přetrvávají a mohou se také časem dočasně zvýšit – to je způsobeno fyzickými vlastnostmi zařízení.

8.3 STATICKÁ CHARAKTERISTIKA

Na základě naměřených dat byla sestrojena statická charakteristika. Systém začne reagovat, když je napětí vyšší než 3,6V, a dochází k významné změně úhlu proti napětí v rozpětí 9-10V, takže bude uvažováno rozpětí 3,6-9V.



Obrázek 8.2 – Statická charakteristika



Obrázek 8.3 – Statická charakteristika systému a přibližná linearizace

Rozdíl mezi lineárním chováním a grafem naměřených dat reálného systému je 2 stupně na začátku a na konci zvolené části statické odezvy při vykreslování tečny, nebo 1 stupeň, je-li přímka vykreslena ve stejné vzdálenosti od začátku, střední, a konec krokové odezvy, takže v tomto rozsahu lze systém považovat za lineární s dalším návrhu regulátoru a hledáním optimálních parametrů pro něj a využitím experimentálních metod k tomu vhodných pro práci s systémy s lineárním chováním, se skutečně naměřenými daty, bez nutnosti jejich následného zpracování a linearizace.

8.4 IDENTIFIKACE

Pokud vezmeme v úvahu statickou odezvu - mezera mezi 6,5V a 7,5V je nejbližší bodu ohybu (inflexní bod) grafu, ale přechodová odezva ukazuje, že průběh v této mezeře výrazně kolísá. Pro další identifikaci bylo rozhodnuto použít dva segmenty: 5-6V, jako nejbližší středu pracovní plochy zařízení a zároveň nejstabilnější na grafu přechodové odezvy, a segment 6-7V jako průměrný interval na nejlineárnější části grafu statické odezvy. Chování systému během odezvy na skokovou změnu napětí v intervalech 4-5V a 5-6V naznačuje, že systém je popsán pomocí aproximační funkce 4-5 řádu.

V ustáleném stavu si systém stále zachovává některé výkyvy v rozsahu 0,5-1,5 stupně (amplituda - 0,25-0,75 stupně), což umožňuje určit přibližný přípustný rozsah regulační chyby a odchylku skutečné hodnoty od nastavenou hodnotu. Při implementaci regulátoru by bylo možné po přechodu systému do stabilního stavu nastavit za úkol snížit konstantní kmitů, ale s vysokou mírou pravděpodobnosti to může vést naopak k destabilizaci systému a zvýšení amplitudy kmitů.

8.4.1 Skript a zpracování dat v MATLABu

Pro vyhledání hodnot funkce byl použit program MATLAB a skript, který hledá minimum pomocí funkce `fminsearch`.

Tento skript byl mírně upraven: časový vektor je generován na základě přijatého datového vektoru a byla přidána funkce použití každé n-té hodnoty v naměřených datových vektorech. Stalo se tak z více důvodů: k práci byl využíván notebook, jehož hardware příliš dlouho zpracovával velké množství řádků zaznamenaných dat, a také, což je hlavní důvod - pravděpodobně kvůli frekvenci procesoru resp. hardware nebo pro nějaký jiný software. Kvůli zacyklení LabVIEW jsou časová data zapisována rychlostí ~1565krát za sekundu, zatímco hardwarové hodiny byly nastaveny na mnohem nižší hodnotu, což způsobilo, že bylo zapsáno více řádků se stejnou hodnotou, což zase způsobilo selhání skriptu MATLAB. Tento

problém nebyl vyřešen zmenšením velikosti vektoru načtením každé n-té hodnoty z něj tak, aby nevznikaly duplicitní řádky, proto bylo použito generování časového vektoru na základě zaznamenaných dat, nicméně řešení se zmenšením délka vektorů byla také využita ke zvýšení rychlosti skriptu a zkrácení čekací doby na nalezení minimálních a požadovaných hodnot.

Generování časového vektoru funguje takto:

1. Ze souboru s uloženými daty se převezme sloupec časových hodnot a na jeho základě se vytvoří datový vektor.
2. Počítá se počet hodnot ve vektoru, první hodnota se bere jako samostatná proměnná n. Dále se vygeneruje jednotkový vektor a vynásobí se n, aby se získal vektor, ve kterém všechny hodnoty odpovídají první hodnotě v časovém vektoru.
3. Od časového vektoru se odečte další vektor. Proto, od jakékoli hodnoty jsou data zapsána, aby skript fungoval, budou vždy začínat od nulové hodnoty.
4. Protože toto řešení způsobilo chybu a také obsahovalo duplicitní časové hodnoty, místo toho se poslední hodnota vektoru vydělí počtem hodnot, čímž se získá krok, ve kterém se má vytvořit časový vektor, a poté se vynásobí vytvořeným vektorem. s hodnotami [0, n-1], čímž se získá nový časový vektor, který se později použije ve skriptu.
5. Volitelné – byly přidány řádky, které přebírají každou hodnotu Tkth z vektorů času, napětí a výstupní hodnoty, aby se snížila velikost vektorů a doba zpracování v počítači. Pro uložení velikosti vektorů a všech naměřených dat stačí nastavit hodnotu Tk=1.

Skript:

```
clear;clc;clf;

data=importdata("Data_DPl.txt");
tm=data(:,2)*0.1;
n=length(tm);
ts=tm(1:1);
Tt=tm-ts;
Um=data(:,4);
Ym=data(:,3);

%Napeti 0-1
Us=Um(1:1);
Ut=Um-Us;
%Uhel od 0
Ys=Ym(1:1);
Yt=Ym-Ys;

Hz=Tt(end)/n; %Nastavení převzorkovací frekvence
```

```
Tg=[0:n-1]*Hz; %Generování časového vektoru

Tk=10; %Zadání velikosti vzorku - přebírání každé Tk-té hodnoty z vektoru
t = Tg(1:Tk:end);
U = Us(1:Tk:end);
Y = Ys(1:Tk:end);
```

Tato část skriptu hledá aproximační model systému na množině naměřených reálných dat. Proces probíhá cyklicky, změnou parametrů se získá model, který se nejvíce blíží chování reálného systému.

```
% parametry diff. rovnice
a1=1; a0=1; b0=1;
Fs=tf(b0,[1 a1 a0]); %prenos tf

Ys=lsim(Fs,U,t,'zoh');
plot(t,U,t,Y);

%optimalizace
p0=[1;1;1];
p=fminsearch(@ (x) fce(x,Y,U,t),p0);
[Er,Yv]=fce(p,Y,U,t);

plot(t,U,t,Y,t,Ys); % zobrazit v grafu s výsledky fminsearch
%plot(t,U,t,Y); % pouze zobrazení napětí a výstupu

xlabel('t, s');
ylabel('u, V; y, °');
grid on;

return

%Funkce aproximace
function [Er,Yv]=fce(p,Ym,U,t)
b0=p(1); a1=p(2); a0=p(3);

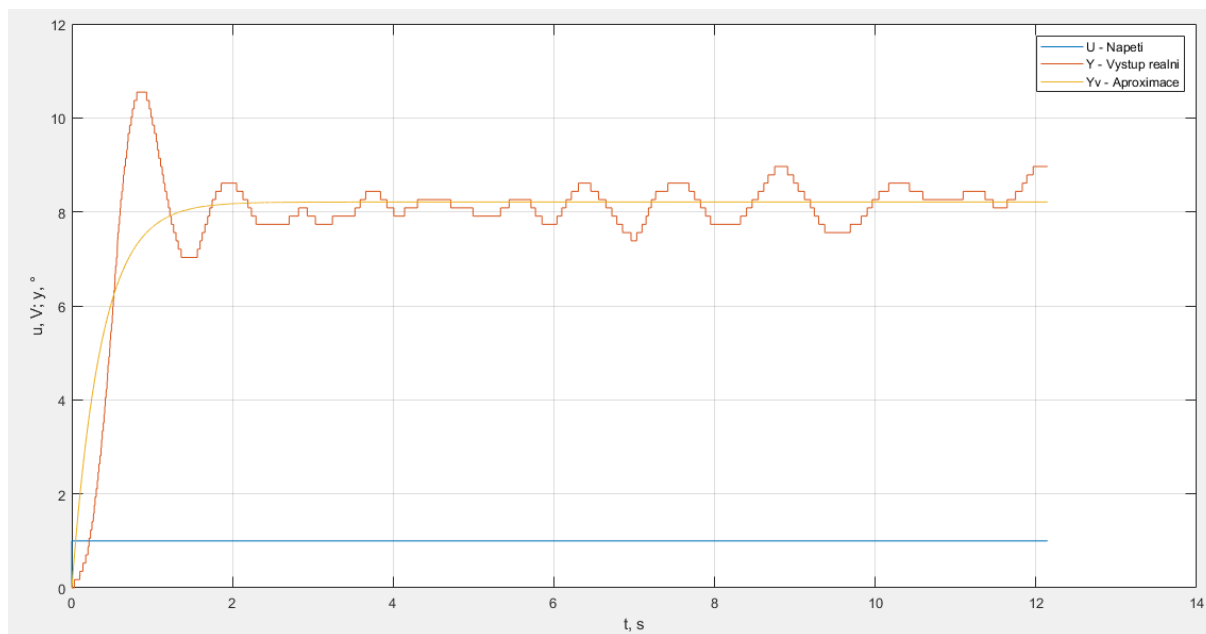
Fs=tf(b0,[1 a1 a0]);
Yv=lsim(Fs,U,t,'zoh');
e=Ym-Yv;
Er=e'*e;
end
```

Funkce *fminsearch* umožňuje najít minimum funkce více proměnných bez omezení, tedy řešení neomezeného optimalizačního problému pomocí simplexní metody.

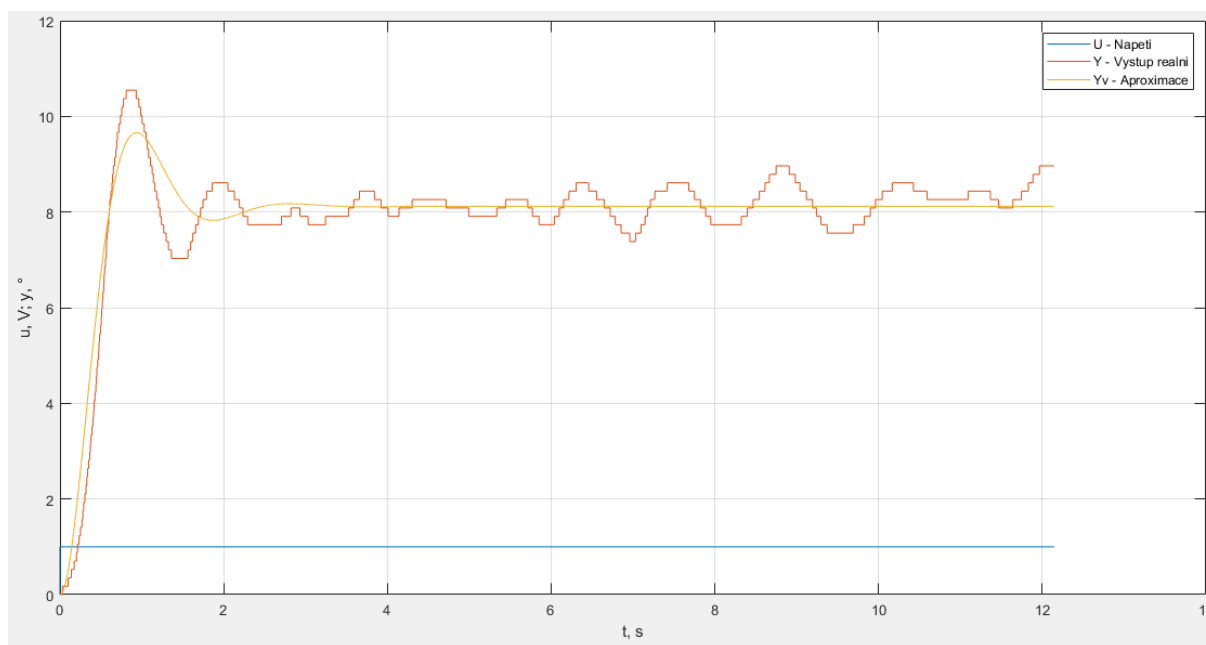
8.4.2 Provádění aproximace

Pomocí funkce *fminsearch* v MATLABu byla provedena řada cyklů aproximace naměřených dat, abychom získali funkci aproximovaného modelu, který by co nejvíce odpovídal reálnému systému a popisoval jeho chování. Protože z přechodové odezvy lze vizuálně pochopit, že systém je popsán funkcí vyššího řádu, byla aproximace provedena pomocí modelů počínaje 2.

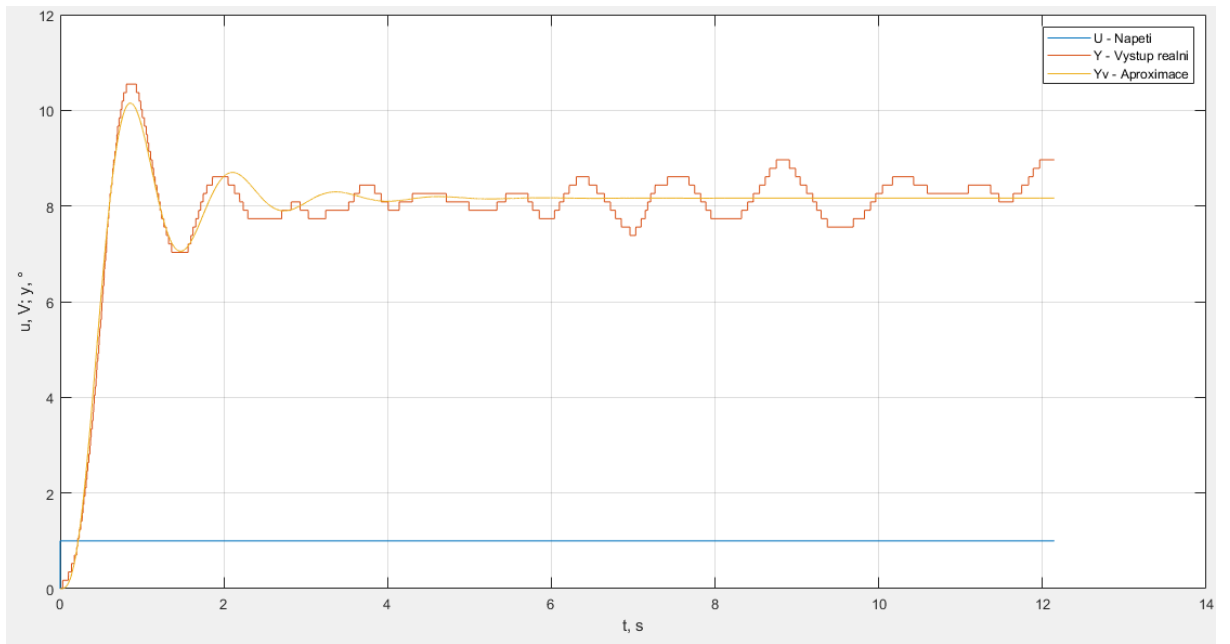
řádem, ale byl také aplikován výsledek aproximace modelem 1. řádu. Výsledné grafy, parametry funkcí a odpovídající střední kvadratická chyba pro různé aproximace řádu jsou uvedeny níže:



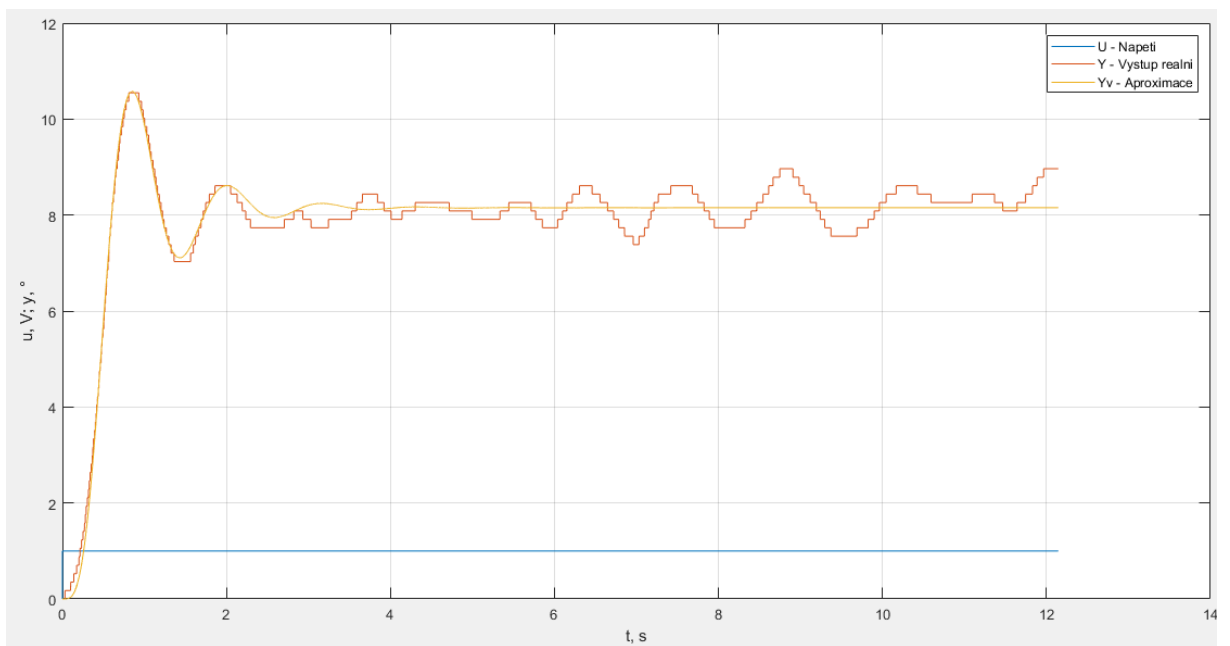
Obrázek 8.4 – Aproximace modelem 1. řádu



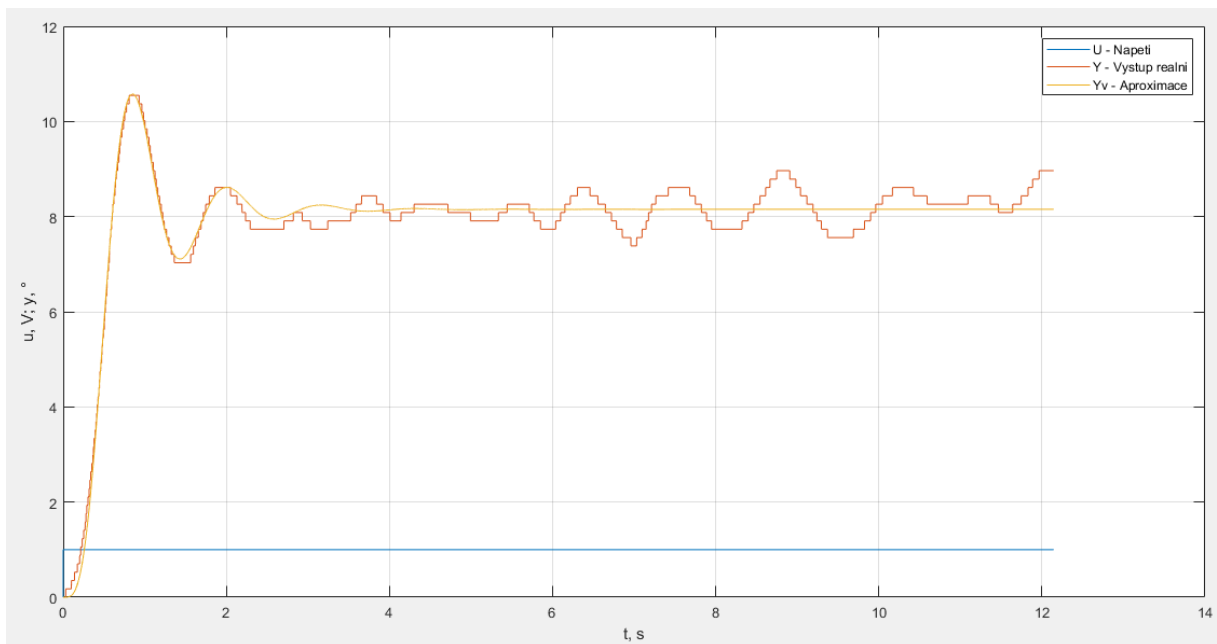
Obrázek 8.5 – Aproximace modelem 2. řádu



Obrázek 8.6 – Aproximace modelem 3. řádu



Obrázek 8.7 – Aproximace modelem 4. řádu



Obrázek 8.8 – Aproximace modelem 5. řádu

Aproximační modely a jejich chyba pro modely různých řádů jsou uvedeny v tabulce 8.1.

Tabulka 8.1 – Výsledky aproximace

Řádek	Model	Chyba aproximace E_r
1	$G_s(s) = \frac{22.37}{s + 2.73}$	11052.8296
2	$G_s(s) = \frac{117.42}{s^2 + 3.56s + 14.46}$	4705.8062
3	$G_s(s) = \frac{830.19}{s^3 + 6.28s^2 + 34.34s + 112.67}$	2209.8503
4	$G_s(s) = \frac{12268}{s^4 + 14.63s^3 + 112.58s^2 + 508.36s + 1505}$	1985.8891
5	$G_s(s) = \frac{6.3770 \cdot 10^{13}}{s^5 + 5.1979 \cdot 10^9 s^4 + 7.6041 \cdot 10^{10} s^3 + 6.3770 \cdot 10^{13}}{+5.8518 \cdot 10^{11} s^2 + 2.6424 \cdot 10^{12} s + 7.8220 \cdot 10^{12}}$	1985.8891

Aproximace 1. a 2. řádu má vizuálně zřejmý nesoulad s reálnými údaji a má také největší aproximační chybu. Funkce 3. řádku má již poměrně blízký popis reálného systému a nepřilíš

výraznou chybu $E_r = 2209,85$ ve srovnání s chybou při aproximaci funkce vyššího řádu. Systém 4. řady má nejlepší aproximaci jak graficky, celkem přesně opakuje graf naměřených dat, tak má menší chybu $E_r=1985,89$. Systém 5. řádku jako výsledek aproximace poskytl výsledek shodný s funkcí 4. řádu, a to jak z hlediska grafu funkce, tak z hlediska hodnoty chyby po konečném posouzení, $E_r=1985,89$. Takže na základě této řady aproximačních cyklů bylo rozhodnuto použít získaná data odpovídající výsledku pro funkci 4. řádu, protože popisuje dynamiku systému přesněji než funkce nižšího řádu a je také matematicky jednodušší a obsahuje menší číselné údaje pro zpracování a matematické výpočty, jako systém vyššího 5. řádu.

Specifičnost soustavy je následující: určitá obtížnost nastavení parametrů je způsobena tím, že systém po celou dobu svého provozu osciluje, díky čemuž použití většiny způsobů nastavení regulátoru nezaručuje stabilní provoz ani nejlepší výsledky.

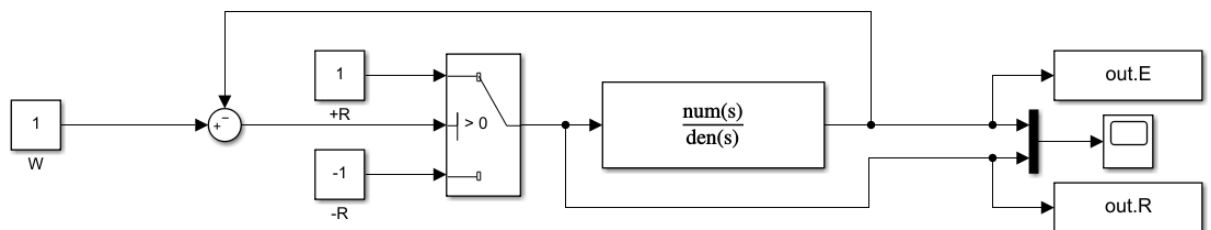
9 VÝPOČET PARAMETRU PID

K výpočtu parametrů byla použita řada metod a realizace matematických výpočtů byla provedena v prostředí MATLAB. Pro prostředí Simulink je v některých případech navíc využíváno pro simulaci chování jednoduchých systémů v softwarovém virtuálním prostředí.

9.1 METODA ZN URČENÍ POMOCÍ RELÉ VE ZPĚTNÉ VAZBĚ

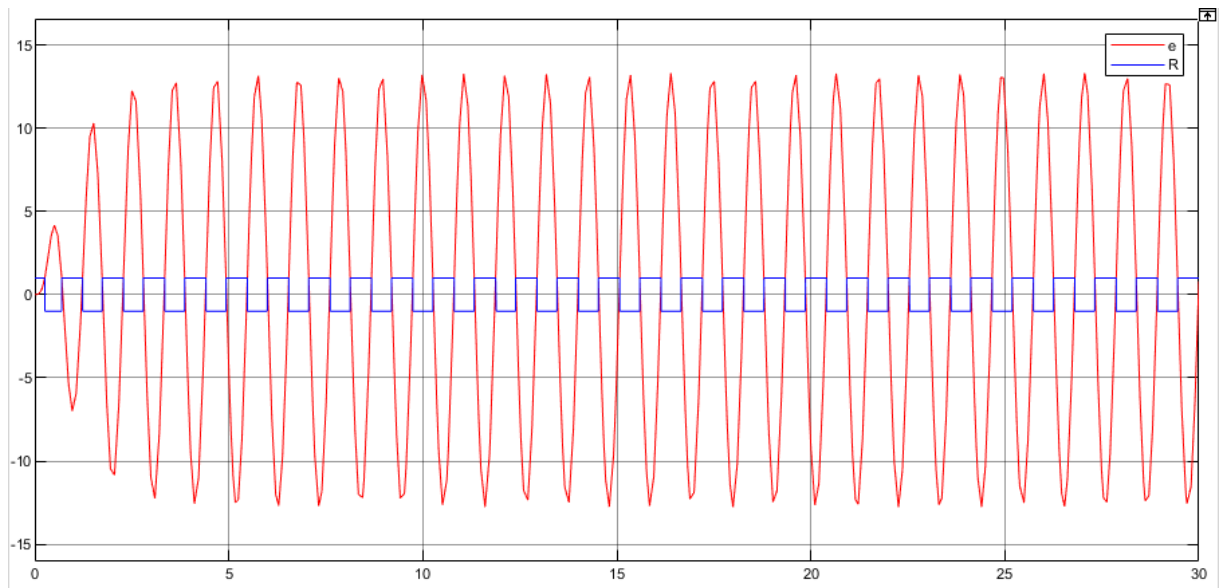
Nejprve byl získán přibližný model systému pomocí MATLABu. Dále bylo v prostředí Simulink sestaveno schéma relé ve zpětné vazbě propojené s Přenosovou funkcí s matematickým modelem systému získaným pomocí aproximace a popisujícím teoretické chování reálného systému. Protože je systém oscilační, byla k výstupní hodnotě systému $y(t)$ přidána porucha $d(t)$, což je oscilační sinusový signál s amplitudou 1,5 a simulující konstantní kmity reálného systému.

Dále byly pomocí simulace v Simulinku získány parametry potřebné pro další výpočty: maximální amplituda e_{\max} , což je v podstatě amplituda oscilací systému na hranici stability, a kritická perioda T_k .



Obrázek 9.1 – Relé ve zpětné vazbě

Výsledný graf oscilací a změn napětí je zobrazen pomocí bloku Scope a bloky Out.E a Out.R, které se v Simulinku nazývají To Workspace, zaznamenávají číselná data, která lze číst nebo používat v MATLABu.



Obrázek 9.2 – Kritické kmity s relé ve zpětné vazbě

Pomocí výsledného e_{\max} se pak vypočítá výsledné kritické zesílení r_{0k} .

Pro výpočet parametrů r_{0k} a byl použit skript v prostředí MATLAB.

Použité parametry:

π – π konstanta rovna 3,1416, automaticky nastavená v MATLABu.

R je rozsah kmitání pro zpětnovazební relé. Nastavuje se ručně na zpětnovazebním relé v Symlinku.

e_{\max} je maximální amplituda mezi minimální a maximální hodnotou konstantního kolísání výkonu operačního systému na hranici stability. Vypočítá se jako je největší hodnota amplitudy, které je dosaženo na začátku kmitů konstantní amplitudy.

T_k je kritická perioda odpovídající době, která uplynula mezi oscilacemi. Nejvhodněji se bere jako poslední dvě po sobě jdoucí kladné nebo záporné hodnoty a hodnota frekvence bude rozdílem v časovém vektoru pro tyto hodnoty.

Hodnoty e_{\max} a T_k jsou vypočítány z jejich hodnotových vektorů získaných simulací chování zpětnovazebního relé v Symlinku.

Dále po získání parametrů r_{0k} a T_k byly vypočteny hodnoty pro ladění PID regulátorů podle vzorců použitých v této metodě.

```
R=1;
Tk=Tkvyp;
emax = max(out.E);
R0k=(4*R) / (pi*emax);
```

```

r0p=0.5*R0k;
Pzn=r0p

r0pi=0.45*R0k;
Tipi=0.85*Tk;
PIzn=[r0pi Tipi]

r0pid=0.6*R0k;
Tipid=0.5*Tk;
Tdpid=0.125*Tk;
PIDzn=[r0pid Tipid Tdpid]

```

Ve skriptu jsou data vypočtena pro všechny typy regulátorů, pro které je metoda ZN použitelná, dle tabulek. Jedná se o regulátory P, PI, PID, jejich hodnoty jsou uloženy v poli a ty se zase zobrazují pro každou variantu regulátoru, jeden řádek pro každý typ. Poté můžete použít libovolný z vypočtených parametrů v závislosti na zvoleném typu regulátoru.

Vypočtené parametry jsou přiloženy v tabulce.

Tabulka 9.1 – Vypočítané parametry regulátoru metodou ZN

	r_0	T_i	T_d
P	0,0478		
PI	0,0430	0,9063	
PID	0,0574	0,5331	0,1333

Při použití metody ZN pomocí relé ve zpětné vazbě je třeba vzít v úvahu, že získané parametry pro ladění PID regulátoru nemusí být dostatečně vhodné pro aplikaci regulace reálným systémem, protože metoda je poměrně agresivní a může vést k nestabilnímu provozu systému.

9.2 EXPERIMENTÁLNÍ TYREUSOVA-LUYBENOVA METODA KRITICKÝCH PARAMETRŮ

Metoda Tyreusova-Luybenova využívá kritické parametry získané dříve při zjišťování parametrů pomocí metody ZN a lze je pak snadno vypočítat podobným způsobem jako metoda ZN pomocí tabulky 4.2. Tento výpočet se provádí nejjednodušší matematickou operací vynásobením kritického zesílení a periody potřebnými konstantami uvedenými v tabulce, a skript, který tyto operace provádí, je uveden níže

```

%Add Experimentalni Tyreusova-Luybenova metoda kritickych parametru
r0tl=0.45*R0k;
Titl=2.2*Tk;
Tdtl=0.16*Tk;

```

```
PIDtl=[r0tl Titl Tdtl]
```

9.3 METODA COHEN-COON

Tato metoda využívá výsledku simulace chování modelu Y_s , získaného pomocí skriptu, který aproximuje chování reálného systému na základě dat přijatých modelem 4. řádu a provádí vlastní aproximaci soustavy přenosem prvního řádu s dopravním zpožděním. Na základě hodnot získaných jako výsledek aproximace, popisující chování systému: zesílení Z , dopravní zpoždění D , doba nabehu T , jsou vypočteny hodnoty τ , α , a jejich čísla jsou již použita podle tabulky. pro výpočet parametrů PID regulátoru.

```
yp=Ys; %vysledek simulace je prevzat z hlavniho skriptu aproximace systemem
4. Radu

%% b) Aproximacni soustava 1. radu
p0=[0.7;0.6;2.6];
p=fminsearch(@(x) ch(x, yp, up, tp), p0);
[chyba, yy]=ch(p, yp, up, tp);
Km=p(1); Tm=p(2); tau=p(3); %nashazime charakteristicka cisla alpha, tau -
Z, D, T z CV9 8/11
Fa=tf(Km, [Tm 1]); Fa.IODelay=tau;

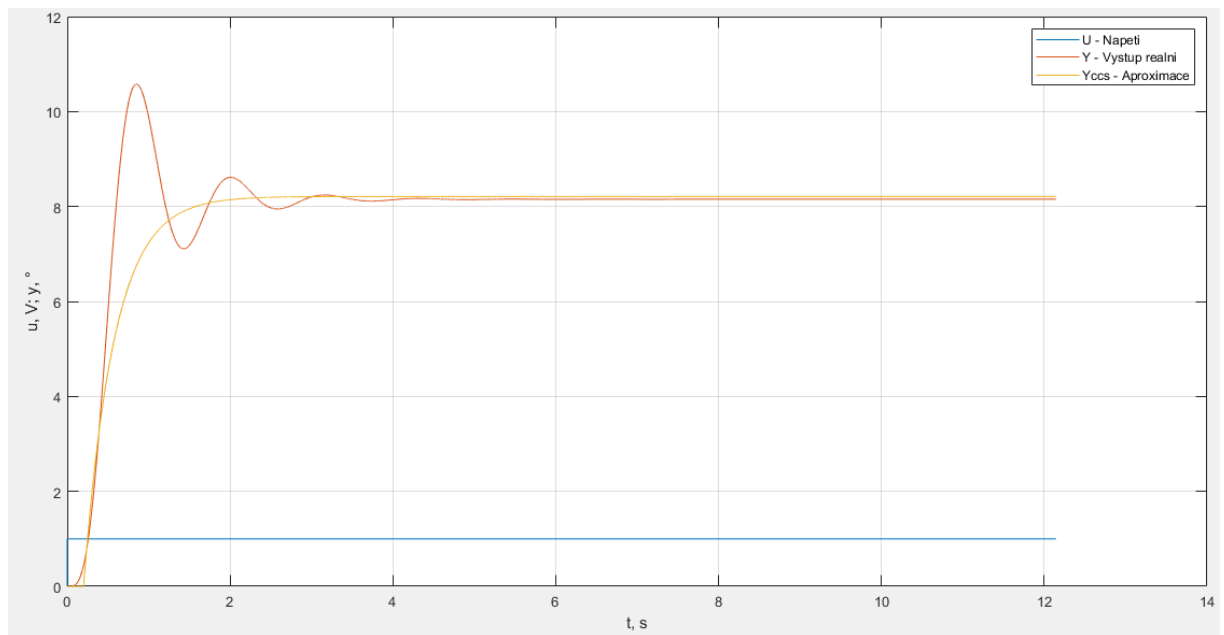
Yccs=lsim(Fa, up, tp, 'zoh');

%% c) Cohen-Coon - parametry PID
%charakteristicka cisla alpha, tau
a=Km*tau/Tm; Tau=tau/(tau+Tm);
%Vypocet P-I-D slozek podle tabulky pro PID-regulator
r0=(1.35/a)*(1+(0.18*Tau)/(1-Tau));
Ti=((2.5-2*Tau)/(1-0.39*Tau))*tau;
Td=((0.37-0.37*Tau)/(1-0.81*Tau))*tau;

plot(tp, up, tp, yp, tp, Yccs, tp, Ys);

%% Funkce - parametry aproximace
function[chyba, yx]=ch(p, yp, up, tp)
    Km=p(1); Tm=p(2); tau=p(3);
    if tau<=0
        tau=0;
    end
    Fs=tf(Km, [Tm 1]); Fs.IODelay=tau;
    yx=lsim(Fs, up, tp);
    chyba=(yx-yp) '*(yx-yp);
end
```

Na závěr skript nasimuluje odezvu systému získanou jako výsledek aproximace na parametry naměřené na reálném systému, kterými jsou jednotkový skok, a na grafu zobrazí simulaci chování aproximovaného systému.



Obrázek 9.3 – Aproximace modelem 1. řádu s dopravním zpožděním

9.4 METODA POŽADOVANÉHO MODELU

Skript pro tuto metodu je popsán níže a v podstatě používá stejné funkce MATLABu jako v předchozích metodách.

```
%nastavuje se z tabulky
%alp=0.3679; % relativni prekmit 0%
%alp=0.5944; % relativni prekmit 10%
alp=0.6881; % relativni prekmit 20%
%% ) Aproximacni soustava 1. radu
p0=[8.13;0.17;0.26;-14.11];
p=fminsearch(@ (x) ch(x, yp, up, tp), p0);
[chyba, yy]=ch(p, yp, up, tp);

Z=p(1); T=p(2); E=p(3); D=p(3);
Ga=tf(Z, [T^2 2*E*T 1]); Ga.IODelay=D;
Ysm=lsim(Ga, U, t);

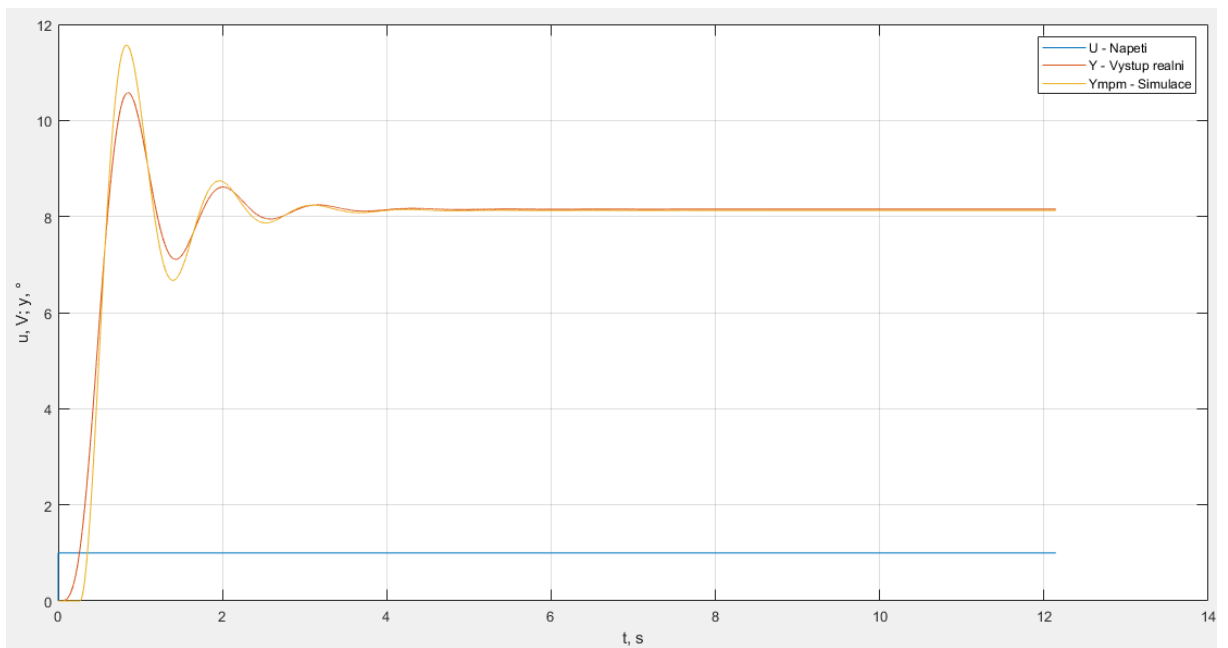
%Vypocet PID
r0=(alp*2*E*T) / (D*Z);
Ti=2*E*T;
Td=T / (2*E);
PIDmpm=[r0, Ti, Td]

plot(t, U, t, Ys, t, Ysm);

function [chyba, yx]=ch(p, yp, up, tp)
    Z=p(1); T=p(2); E=p(3); D=p(3);
    if D<=0
        D=0;
    end
    Gs=tf(Z, [T^2 2*E*T 1]); Gs.IODelay=D;
    yx=lsim(Gs, up, tp);
    chyba=(yx-yp) '*(yx-yp);
end
```


Skript této metody funguje stejně jako skript pro Cohen-Coonovu metodu: vyhledávají se optimální parametry pro funkci, která by měla popisovat chování reálného systému co nejlíže tomu skutečnému na základě naměřená data s tím rozdílem, že obě tyto metody využívají různé aproximační modely a různé vzorce pro výpočet výsledných parametrů PID regulátoru. Charakteristickým rysem této metody je možnost nastavení parametru α , který určuje požadovaný překmit pro finální ovládání regulátoru systému.

Výsledek simulace na základě nalezeného modelu získané metodou požadovaného modelu si můžete prohlédnout níže.



Obrázek 9.4 – Aproximační model (tlumená kmitavá soustava s dopravním zpožděním D)

9.5 VÝSLEDNÉ PARAMETRY

Výsledky získané pomocí různých metod ladění parametrů PID regulátoru jsou uvedeny níže. Dále jsou jim prezentovány výsledky matematické analýzy zjištěných parametrů pomocí integrálních kritérií kvality. Pro metodu požadovaného modelu byly provedeny tři výpočty s různými ručně nastavenými parametry α^* , odpovídajícími relativnímu překmitu $\kappa = 0\%$, 10% a 20% .

Tabulka 9.2 – Vypočítané parametry regulátoru

Metoda	P	I	D
Ziegler-Nichols rele	0,0574	0,5331	0,1333
Tyreusova-Luybenova	0,0430	2,3456	0,1706
Cohen-Coon	0,3384	0,4176	0,0672
Požadovaného modelu 0%	0,0157	0,0918	0,3293
Požadovaného modelu 10%	0,0254	0,0918	0,3293
Požadovaného modelu 20%	0,0294	0,0918	0,3293

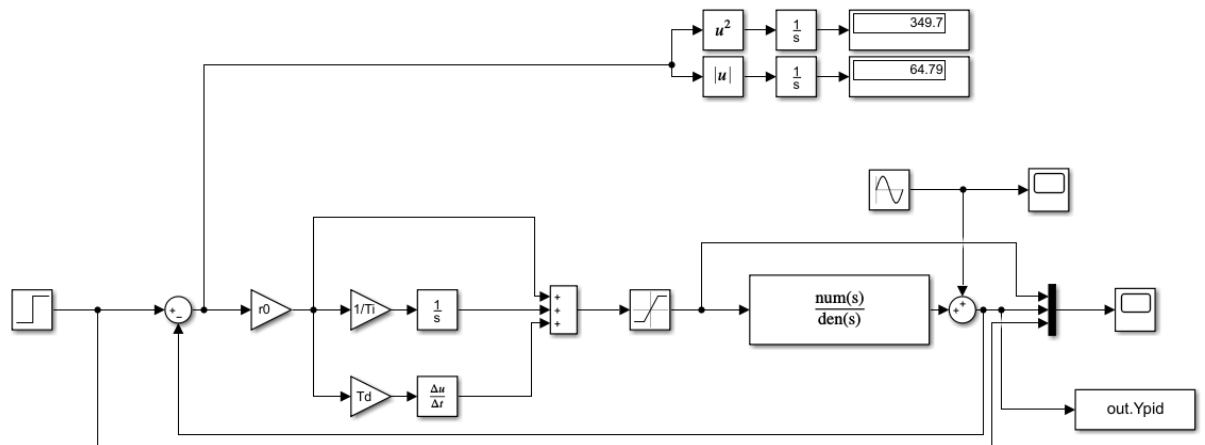
Nalezené parametry regulátoru byly nastaveny pro PID regulátor v Simulinku pro simulaci očekávaného chování systému a také pro získání hodnot integrálních kritérií kvality pro teoretické posouzení kvality regulace nalezenými parametry systému pomocí matematických metod. Na základě hodnot vyhodnocení pomocí integrálních kritérií a také grafů získaných simulací lze učinit předpoklad, který z parametrů získaných různými metodami by byl nejlepší. Simulace chování modelu může také poskytnout představu o očekávaném chování reálného systému a nezávisle zvolit nejvhodnější parametry z hlediska výkonu nebo stability systému.

9.6 PROVÁDĚNÍ SIMULACÍ

Pro simulaci v prostředí Simulink bylo vytvořeno URO s matematickým modelem systému, získaným aproximací, řízeným PID regulátorem ve schématu standardního zapojení. PID regulátor je zapojen do série v obvodu za modelem systému před ním, který má zpětnou vazbu. Ke změně požadované hodnoty $w(t)$ dochází náhle. Regulátor pracuje tak, že přijme vstupní hodnotu požadované hodnoty $w(t)$, od které se odečte aktuální výstupní hodnota systému $y(t)$, podle toho se vypočítá generované napětí. K výstupu systému je přidán sinusový signál pro simulaci oscilační povahy systému. Signál má tyto parametry: amplituda 0,8, frekvence 6 radiánů/sekundu, což odpovídá kolísání přechodové odezvy systému. Na

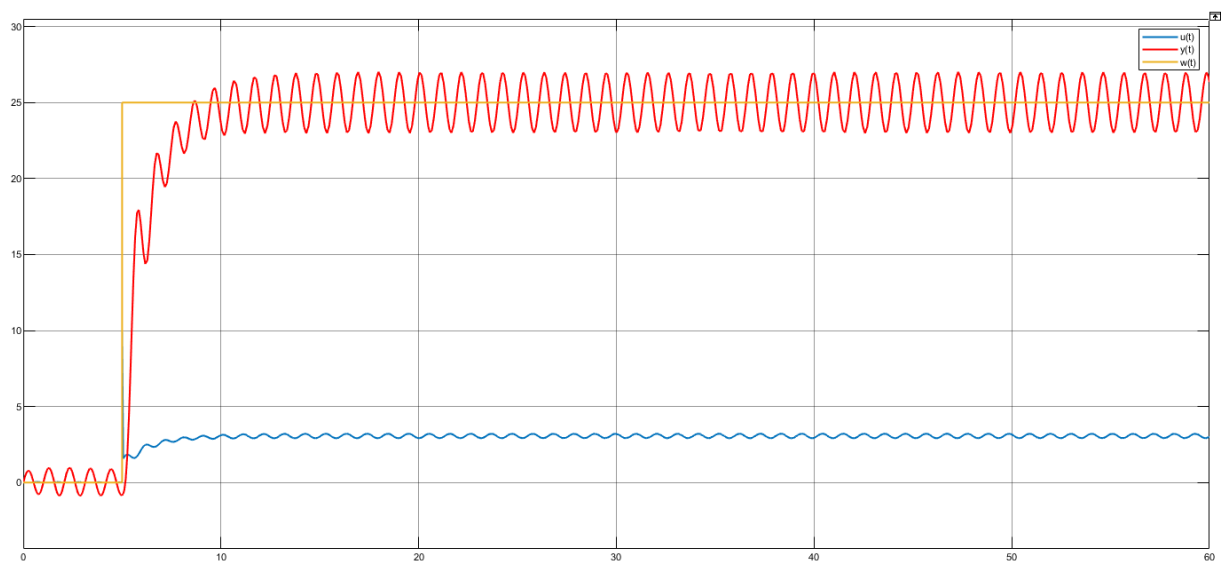
obrazovce jsou zobrazeny grafy výstupní hodnoty systému $y(t)$, vypočtené regulátorem napětí $u(t)$, a také požadovaná hodnota $w(t)$. Systém simuluje chování při zvyšování požadované hodnoty z počáteční nuly na 25 stupňů, což bude odpovídat změně z počáteční polohy na průměrnou hodnotu v lineárním rozsahu reálného systému při asi 0 stupních.

Graf změny $y(t)$ je uložen jako vektor a jeho data je možné prohlížet v MATLABu a byl doplněn i výpočet kvality regulace pomocí integrálních kritérií.

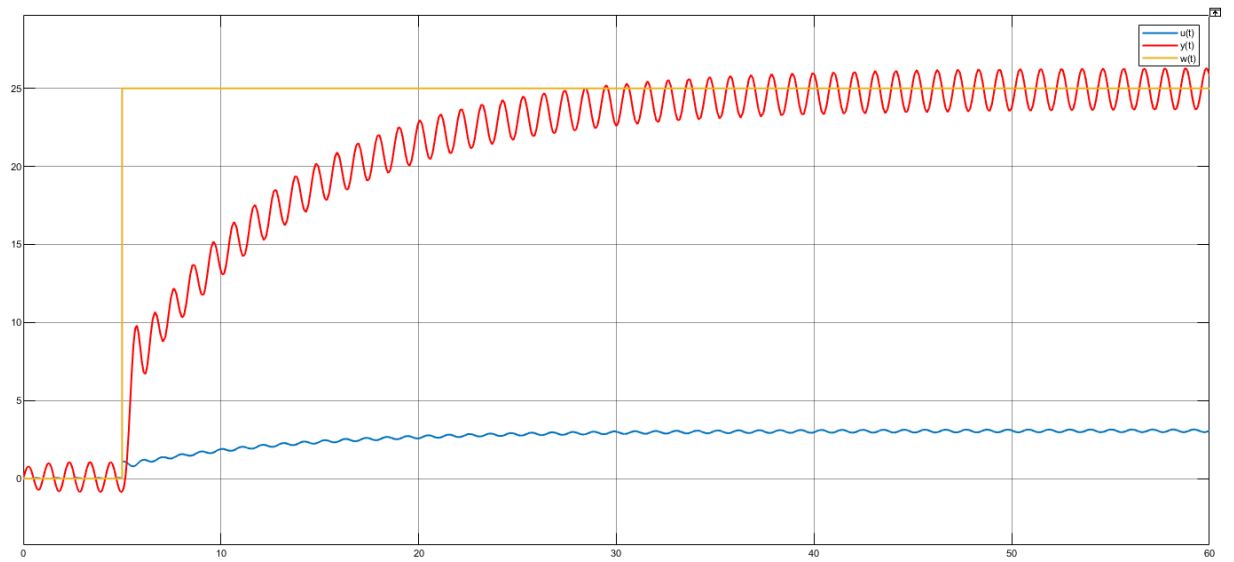


Obrázek 9.5 – Simulační obvod v Simulinku

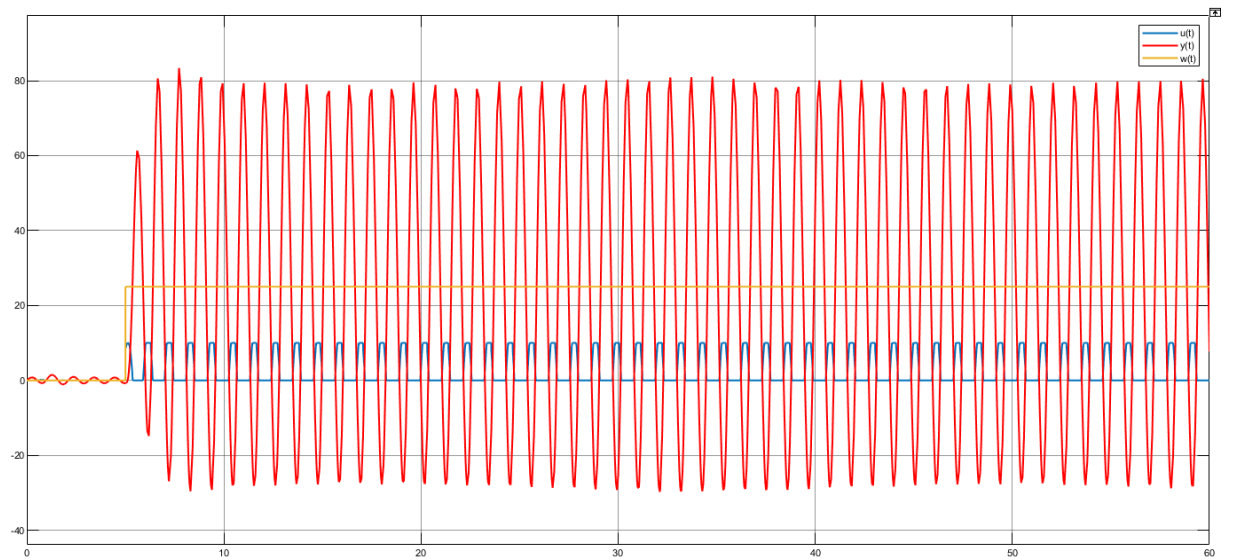
Vzhledem k dlouhé odezvě způsobené vysokým I členem v metodě TL běží simulace 60 sekund.



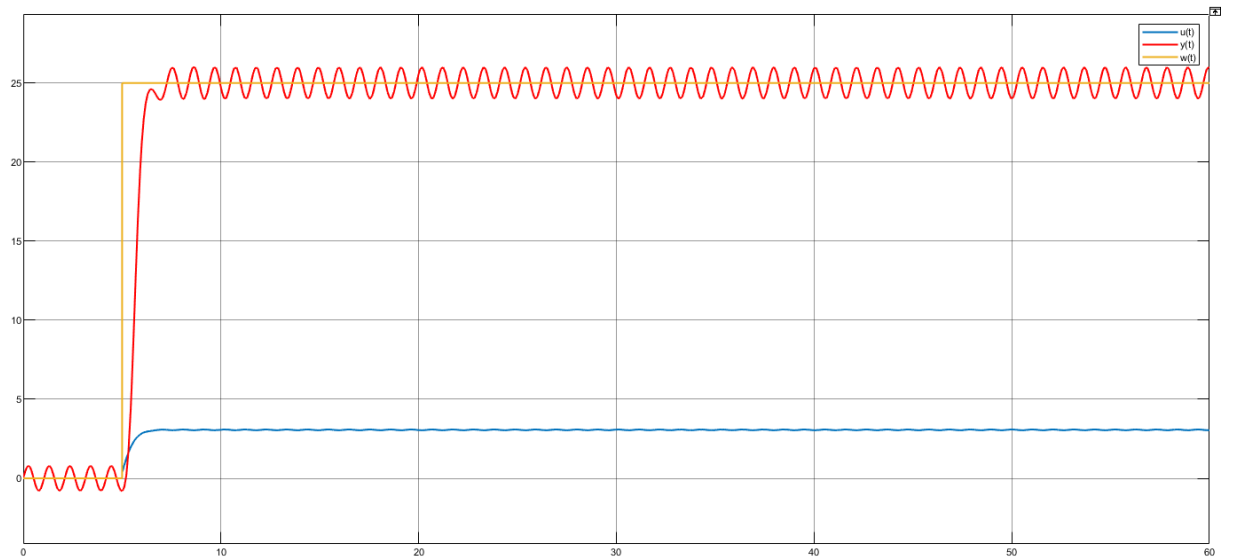
Obrázek 9.6 – Simulace podle metody ZN



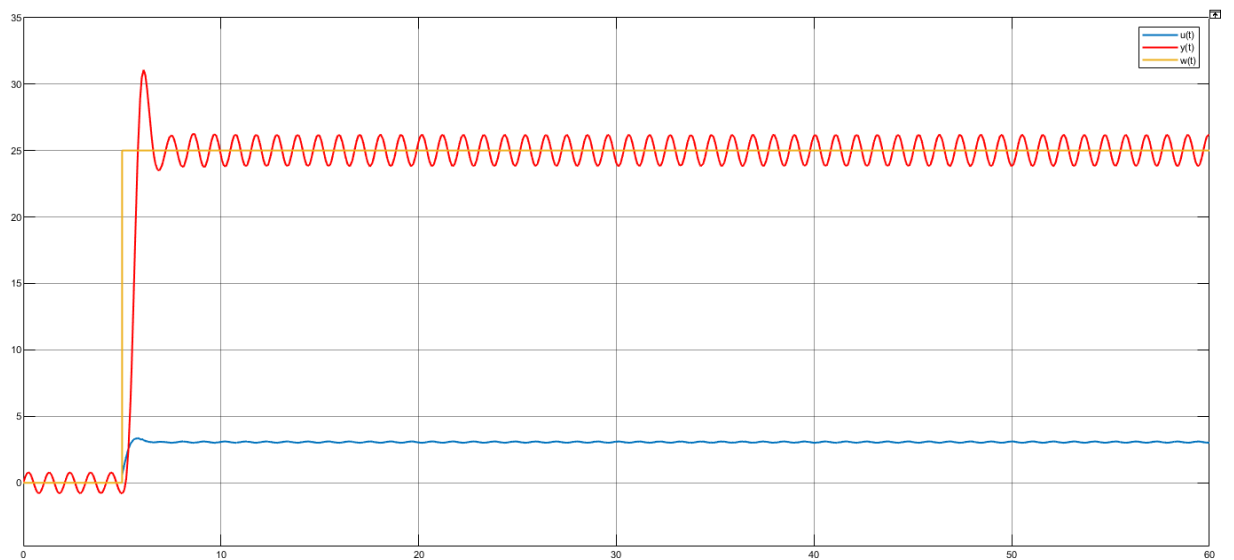
Obrázek 9.7 – Simulace podle metody TL



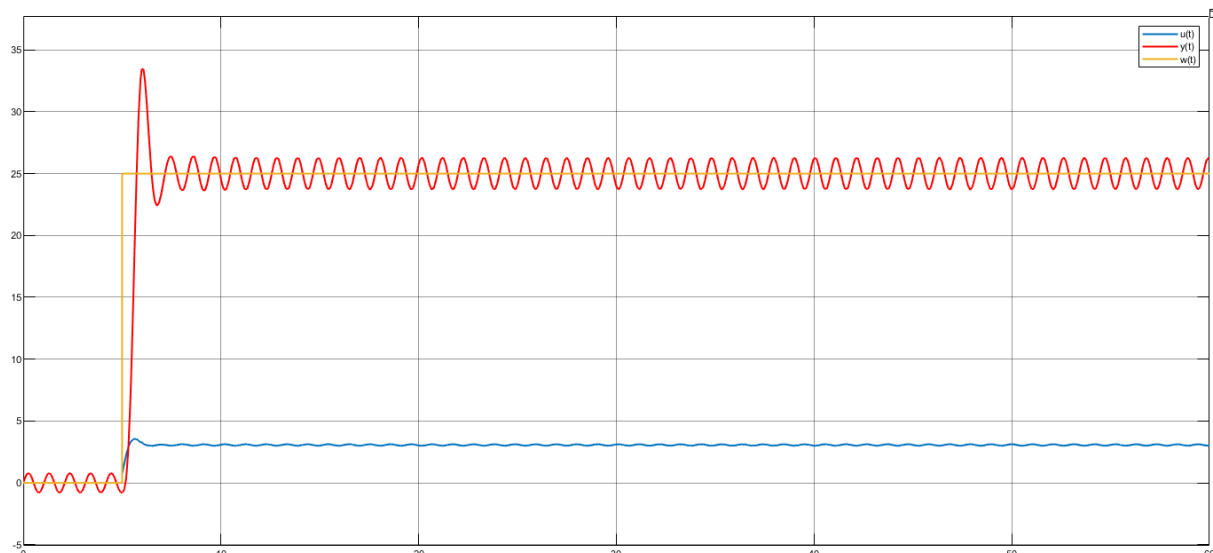
Obrázek 9.8 – Simulace podle metody CC



Obrázek 9.9 – Simulace podle metody MPM překmit 0%



Obrázek 9.10 – Simulace podle metody MPM překmit 10%



Obrázek 9.11 – Simulace podle metody MPM překmit 20%

Tabulka 9.3 – Vypočítané parametry regulátoru a integrální kritéria kvality

Metoda	P	I	D	IAE	ISE
Ziegler-Nichols rele	0,0574	0,5331	0,1333	95	470,2
Tyreusova-Luybenova	0,0430	2,3456	0,1706	188,6	1713
Cohen-Coon	0,3384	0,4176	0,0672	1888	80032
Požadovaného modelu 0%	0,0157	0,0918	0,3293	54,7	376
Požadovaného modelu 10%	0,0254	0,0918	0,3293	60,3	341,4
Požadovaného modelu 20%	0,0294	0,0918	0,3293	64,8	349,7

V souladu s hodnocením výsledků podle integrálních kvalitativních kritérií vykazala teoreticky nejlepší výsledky metoda požadovaného modelu s překmitem 10 %, stejně jako v případě uvažování grafů. Nejhorší výsledky vykazovala metoda CC z důvodu nestabilního chování systému, a metoda TL z důvodu velmi nízké rychlosti regulace.

Všechny simulační grafy obsahují kmitání a u všech metod kromě metody CC vypadají velmi podobně (kromě metody ZN, u které jsou o něco vyšší), což je způsobeno simulací kmitání soustavy, a ve své podstatě se reálný systém chová podobně, takže z tohoto pohledu simulace v mnoha ohledech odpovídá reálnému systému.

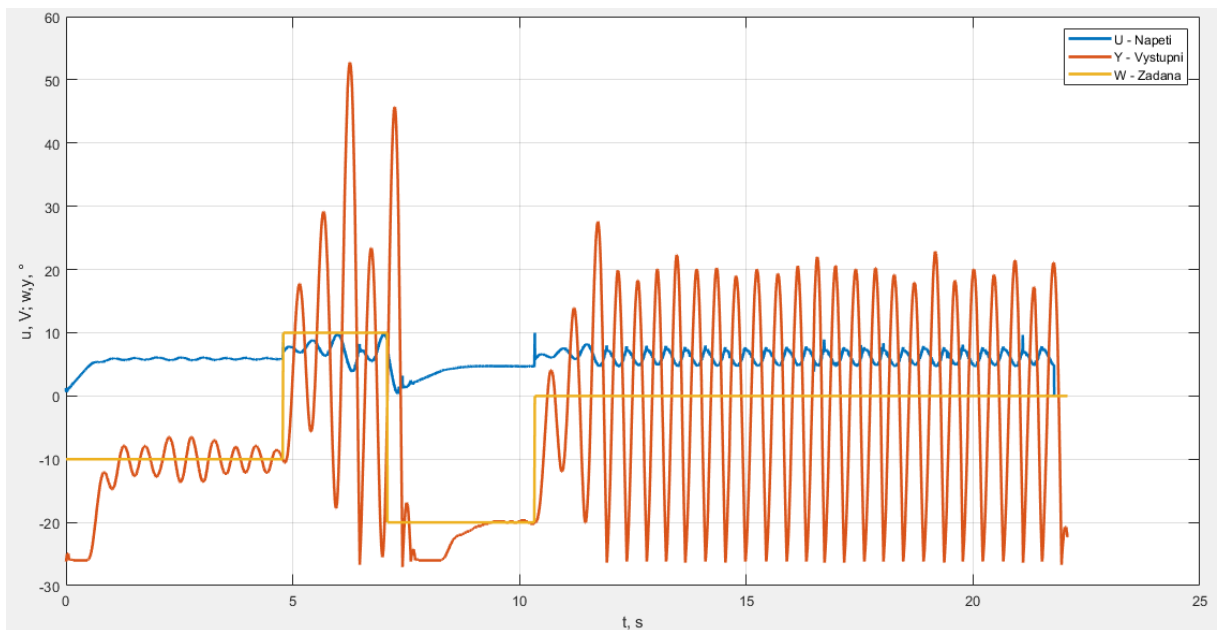
Protože je ale reálný systém oscilační, je třeba počítat s tím, že nalezené parametry nemusí vyhovovat jeho regulaci a jeho chování se může lišit od očekávaného.

10 REGULACE REÁLNÉHO SYSTÉMU

10.1 ÚPRAVA PODLE VYPOČTENÝCH PARAMETRŮ

10.1.1 Metoda ZN

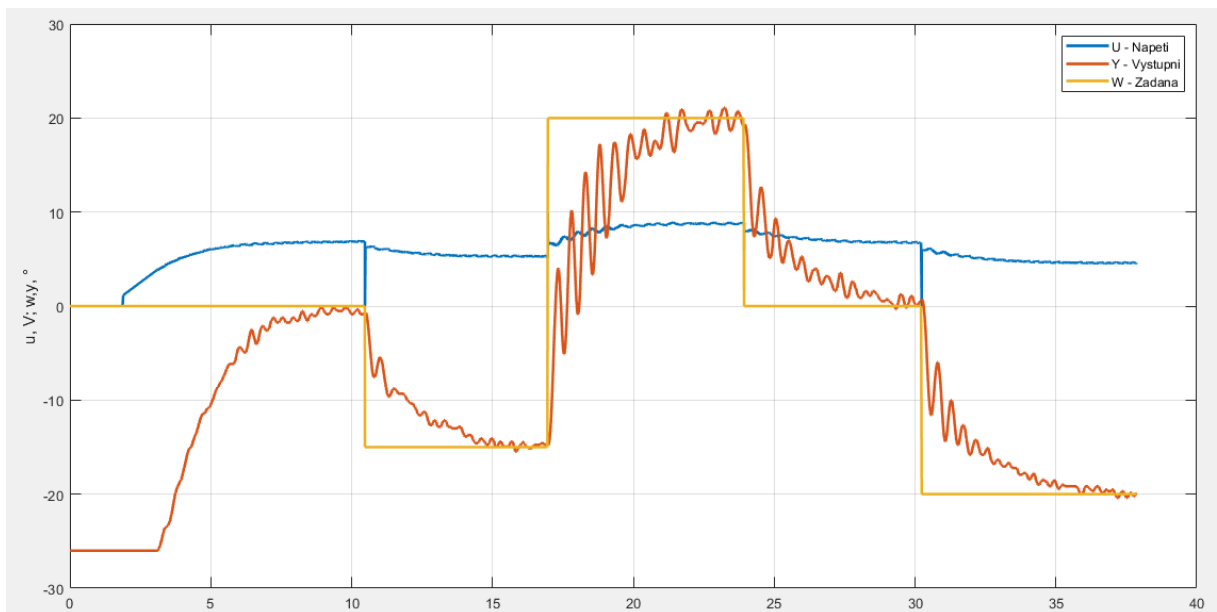
System je možné regulovat v případě $w(t) = -10$ stupňů nebo méně, ale se zvýšením zadane je system nestabilní.



Obrázek 10.1 – Chování reálného systému s laděním regulátoru metodou ZN

10.1.2 Metoda TL

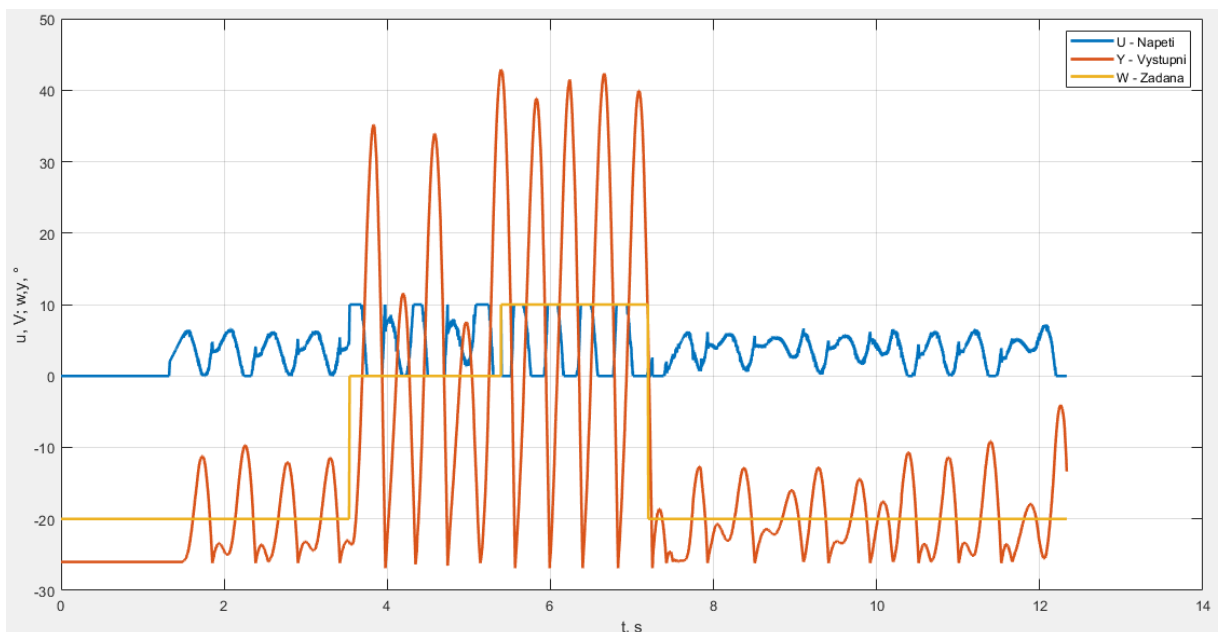
System pracuje poměrně pomalu a vyznačuje se také silným kmitáním při změně $w(t)$ v procesu nastavování výstupní hodnoty $y(t)$ na požadovanou, nicméně regulátor plní svou funkci a zajišťuje stabilní provoz regulátoru. system po dosažení výstupni hodnoty $y(t)$ do požadované.



Obrázek 10.2 – Chování reálného systému s laděním regulátoru metodou TL

10.1.3 Metoda CC

Tato metoda ukázala nejvíce nestabilní chování systému pro jakékoli parametry a změny.



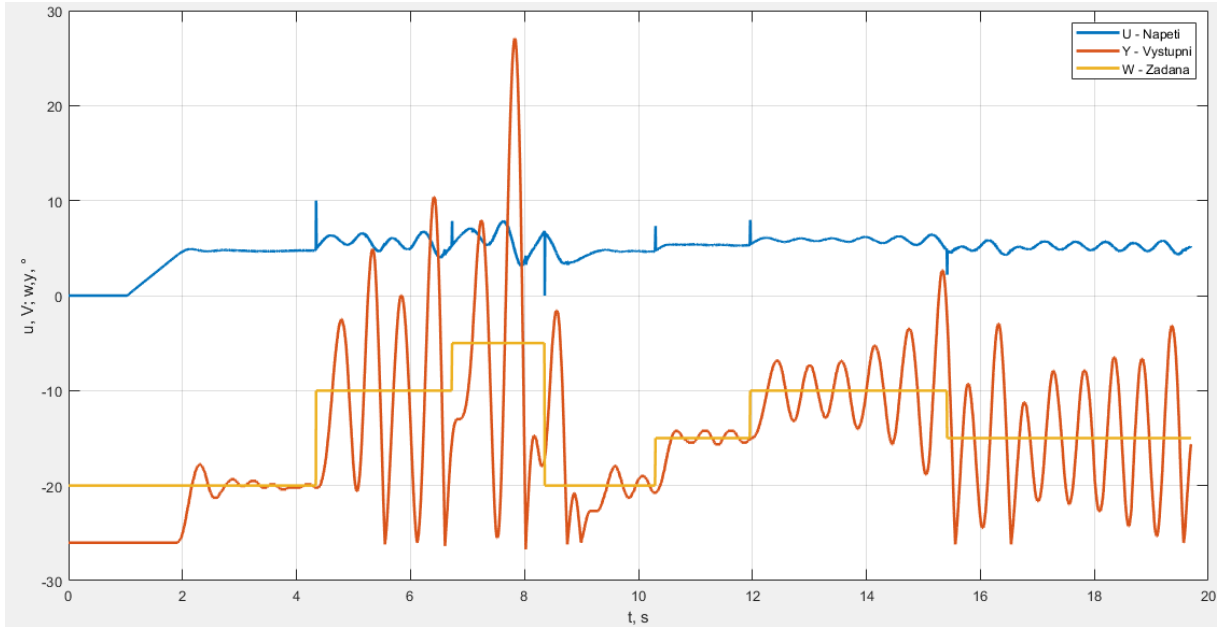
Obrázek 10.3 – Chování reálného systému s laděním regulátoru metodou CC

10.1.4 Metoda požadovaného modelu

V důsledku odlišného parametru κ a podle toho α^* má MPM různé hodnoty P složky pro požadovaného překmitu 0, 10, 20%. V případě nenulového překmitu se systém chová nestabilně, při použití MPM $\kappa=0\%$ v důsledku ladění regulátor provádí regulaci při $w(t)$ až -

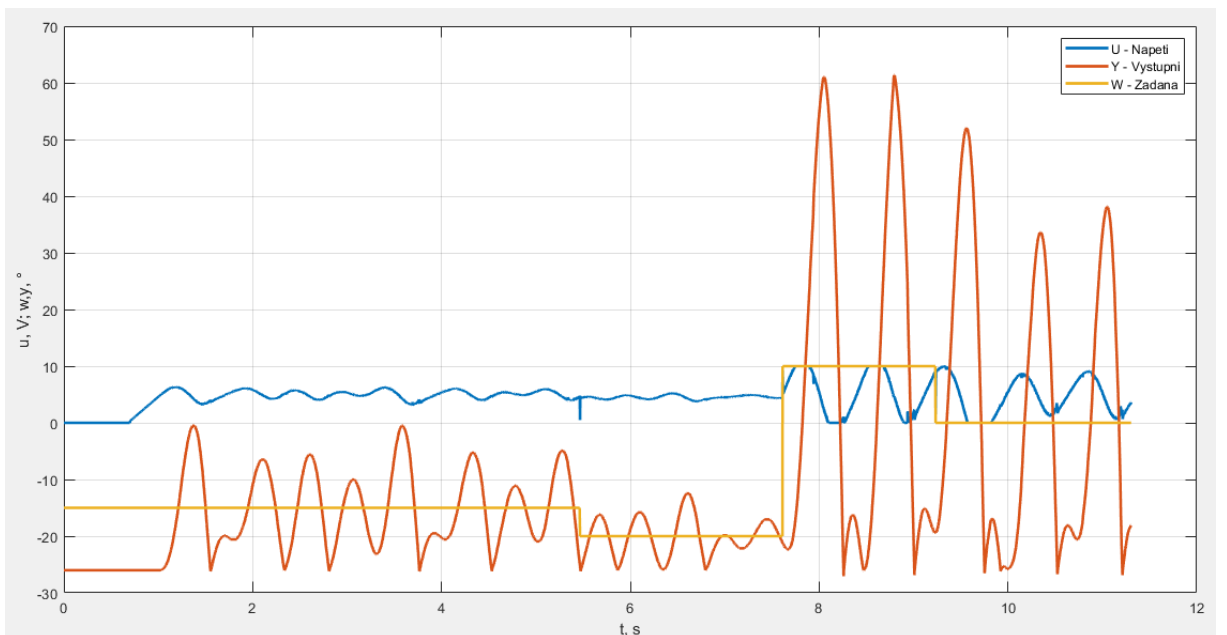
15 stupňů, při vyšších hodnotách přesahuje oblast stability. Zvýšení P složky, počítané s vyšším požadovaným překmitem, způsobuje nestabilní provoz systému v celém rozsahu možných hodnot pro nastavení.

MPM $\kappa=0\%$



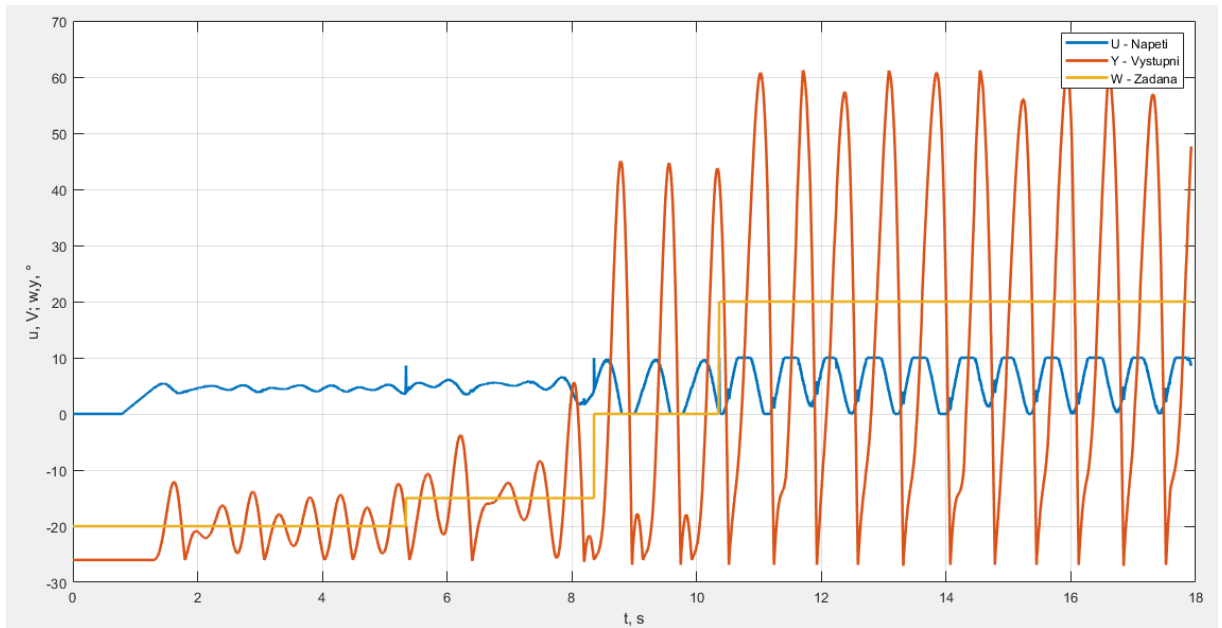
Obrázek 10.4 – Chování reálného systému s laděním regulátoru MPM (překmit 0%)

MPM $\kappa=10\%$



Obrázek 10.5 – Chování reálného systému s laděním regulátoru MPM (překmit 10%)

MPM $\kappa=20\%$



Obrázek 10.6 – Chování reálného systému s laděním regulátoru MPM (překmit 20%)

Ze všech metod má nejlepší výsledky regulace ukázal použití parametrů, získané metodou Tyreusova-Luybenova: systém se chová poměrně pomalu, ale nejstabilněji, navíc v celém rozsahu, stejně jako s prudkými změnami $w(t)$, na rozdíl od výsledků získaných jinými metodami.

Dalším vhodným postupem je využít zjištěné parametry a provést jejich dodatečnou ruční úpravu, za účelem zvýšení výkonu systému a případně dalších změn v dynamice systému.

10.2 RUČNÍ POKROČILÉ NASTAVENÍ PARAMETRŮ

10.2.1 Obecný popis nastavení

Ze všech složek PID regulátoru se výsledky získané metodou Tyreusova-Luybenova liší od zbytku výrazně vyšší hodnotou I složky, což má za následek stabilní, ale extrémně pomalé chování systému, takže první krok ručního ladění bylo snížit hodnotu I složky ke zvýšení rychlosti systémů.

Pokles integrační části, který probíhal postupně, na hodnotu 0,9, výrazně zvýšil rychlost, s jakou systém dosahuje požadovaných parametrů, ale také vede k nestabilnímu provozu systému v případě, že výstupní hodnota $y(t)$ přesahuje mez linearitu (více než 20 stupňů), což je možné i v případě, že požadovaná hodnota $w(t)$ tuto hodnotu nepřekročí, a to z důvodu

oscilačního charakteru systému a překmitu. Rovněž k nestabilnímu chování při této hodnotě složky I dochází při prudkém poklesu požadované hodnoty $w(t)$ o 10 stupňů a více (při změně požadované hodnoty o 5 stupňů a v rozmezí od -26 do +20 stupňů $w(t)$, systém funguje stabilně). Další pokles I složky zaručeně povede k nestabilnímu chování systému, proto byla v této fázi nastavování regulátoru a zkoumání vlivu parametrů jeho regulátoru na dynamiku systému brána jeho použitá hodnota 0,9.

V souvislosti se situacemi nestabilního chování došlo také ke změně derivační složky D, která se postupně zvyšovala na vyšší hodnoty. Byly provedeny experimentální úpravy s různými parametry derivační složky D. Obecně se její postupné zvyšování pozitivně projevilo na stabilitě soustavy a, což může být v některých případech důležité, s eliminací oscilačního charakteru soustavy: s jeho nárůstem probíhala změna úhlu natočení $y(t)$ plynuleji a bez kolísání nebo dočasného přerušování změny aktuální výstupní hodnoty, a to až do určitého okamžiku – po kterém došlo k dalšímu nárůstu hodnoty D složky vedlo ke stále výraznějšímu předkmitu a překmitu, v důsledku čehož po změně požadované hodnoty $w(t)$ se úhel natočení nejprve po určité době změnil v opačném směru, než ve kterém se měl změnit, a po dosažení požadované hodnoty se jeho hodnota zvýšila nebo snížila. dále než je požadováno, načež ještě poměrně dlouhou dobu $y(t)$ dosahovalo nastavené hodnoty $w(t)$. V souladu s tím byla stanovena určitá horní hranice, nad kterou nemá smysl zvyšovat složku D, jejíž hodnoty a popis odpovídajícího charakteru chování systému budou uvedeny dále (napsáno níže) a výsledky ručního ladění a experimentů budou seskupeny do jedné samostatné části kapitoly.

Druhou složkou vlivu změny složky D na chování systému je změna jeho kmitání: s jeho nárůstem charakter změny $y(t)$, totiž změna aktuálního úhlu natočení a polohy ventilátoru, stává se hladším a zbavuje se oscilací v procesu regulace a dosazení do nové zadané hodnoty $w(t)$.

Vzhledem k tomu, že začátek rozsahu celkem plynulé změny $y(t)$ je nižší než práh, nad kterým dochází v systému k výraznému překmitu, je možné zvolit D složku tak, aby se systém při změně $w(t)$ choval co nejplynuleji a přitom bez překmitu, nebo zvolit alternativní možnost ladění s výraznější plynulostí ovládnutí, ale s malým překmitem. Obecně platí, že volba parametrů, které mají být upraveny pro regulaci, je kompromisem mezi různými aspekty chování systému, mezi nimiž lze upřednostnit jeden z těchto aspektů, to znamená udělat jej co nejvyšší kvality a zároveň zhoršit chování z hlediska jiných vlastností dynamiky systému, nebo zvolit mezivolbu takové parametry, za kterých budou všechny prvky dynamiky

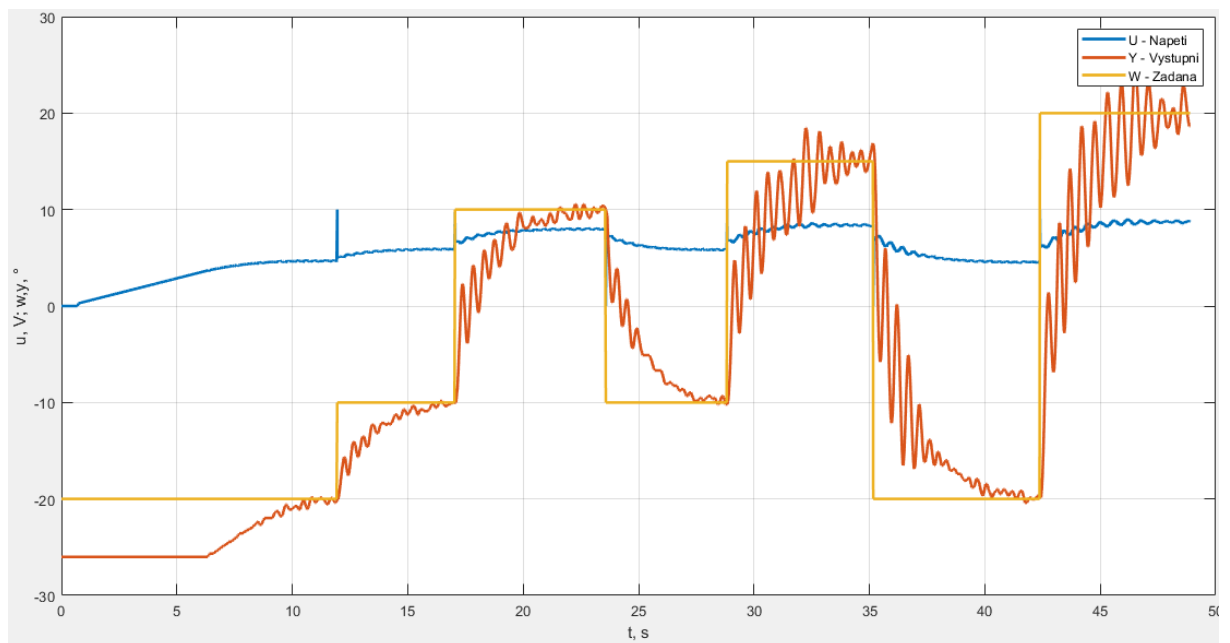
systemu jako celku fungovat na normální úrovni, aniž by se některé zlepšovaly na úkor zhoršování jiných.

Protože zvýšení hodnoty složky D v důsledku mělo negativní dopad na výkon systému a dosažení požadované hodnoty $w(t)$ trvalo výrazně dlouhou dobu, bylo rozhodnuto vyzkoušet ke snížení parametru I složky regulátoru. Při jejím snížení na hodnotu 0,3 systém vykazoval určitý překmit spojený s oscilační povahou systému, ale lze jej považovat za nevýznamný ve srovnání s předchozími výsledky, neboť i přes to, že hodnota překmitu byla větší, rychlost, s jakou zařízení dosáhlo požadované hodnoty, výrazně vzrostla.

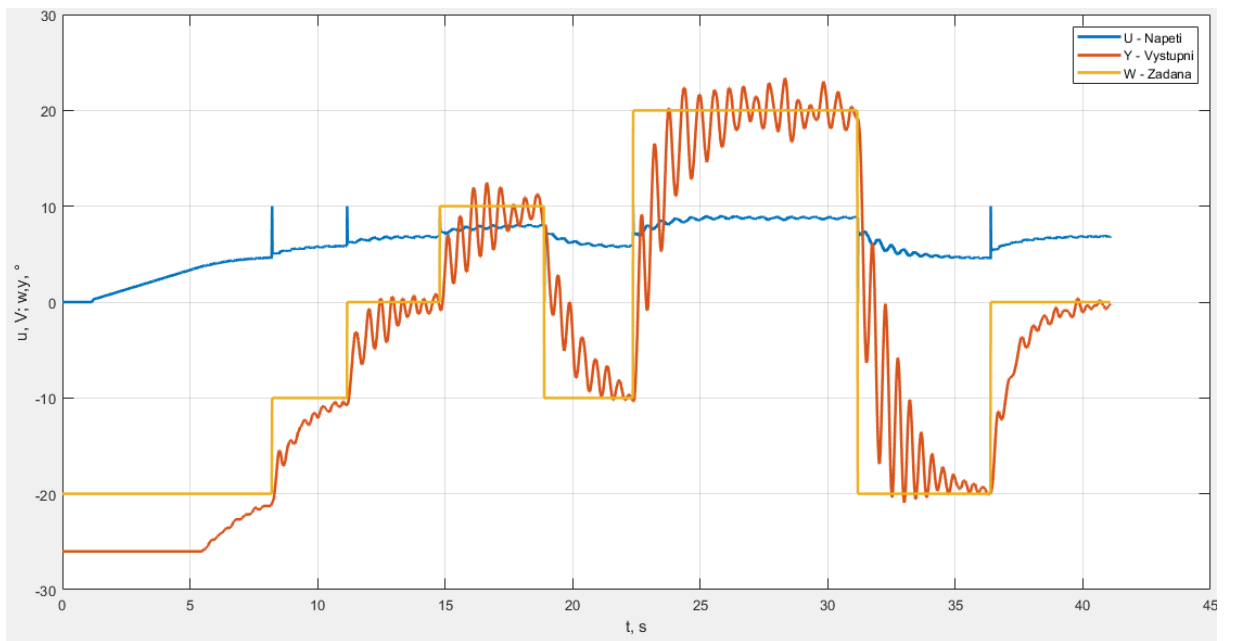
10.2.2 Parametry regulátoru a odpovídající dynamika

Následují přesnější hodnoty komponent PID regulátoru, odpovídající grafy chování systému, rozsahy hodnot a vlastnosti dynamiky chování systému odpovídající každé z nich, které byly získány v proces ručního ladění regulátoru a studium chování systému. To slouží jak k ilustraci procesu a sledu experimentálního nastavování regulátoru, tak k pochopení změn dynamiky systému potenciálním uživatelem a také umožňuje zvolit pro regulaci laboratorním zařízením ty parametry, se kterými se dynamika největší míra odpovídá požadovanému chování systému při jeho regulaci.

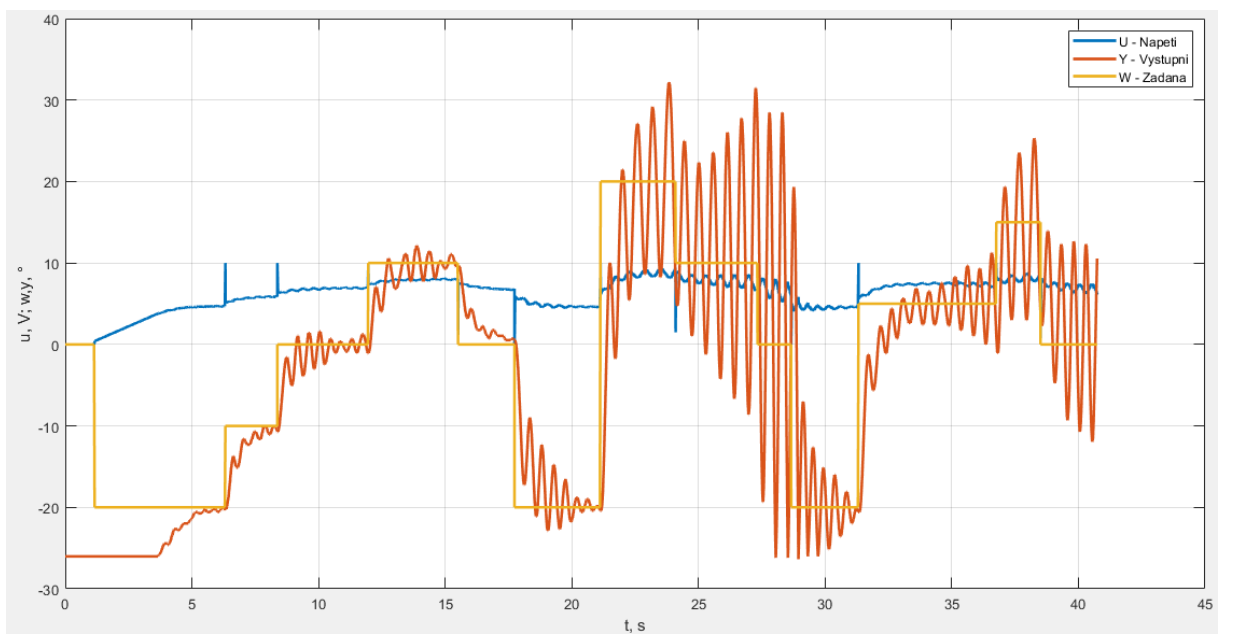
Nejprve byly odebrány parametry získané metodou TL a změněna I složka.



Obrázek 10.7 – Chování reálné soustavy při ručním ladění, I složka = 2

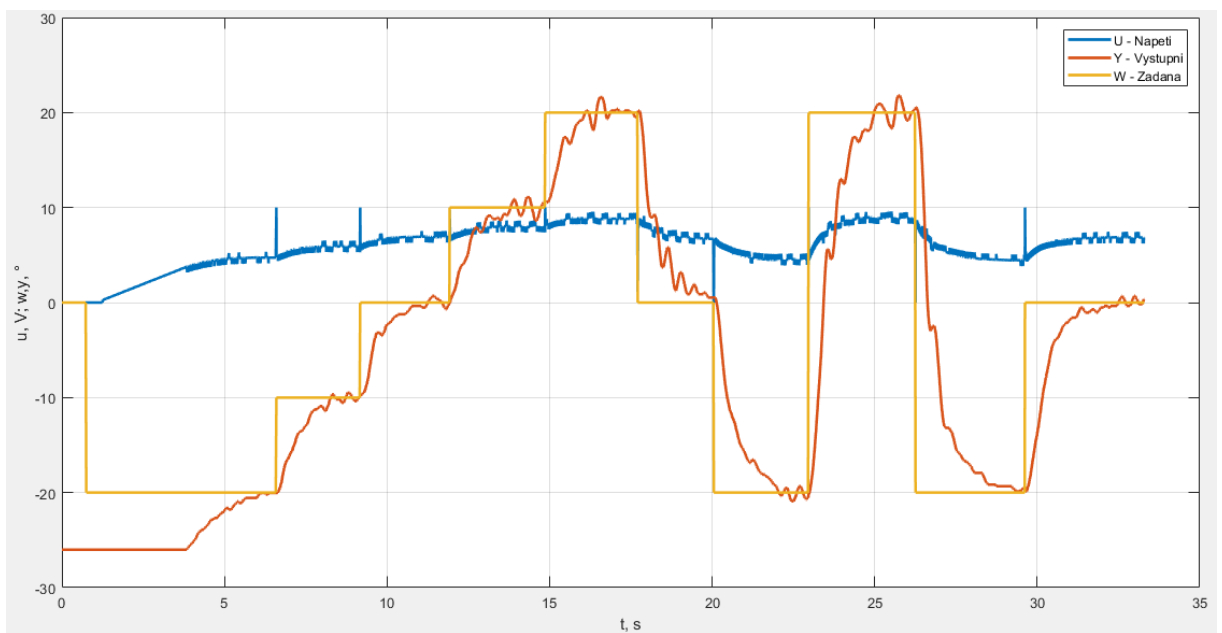


Obrázek 10.8 – Chování reálné soustavy při ručním ladění, I složka = 1,5

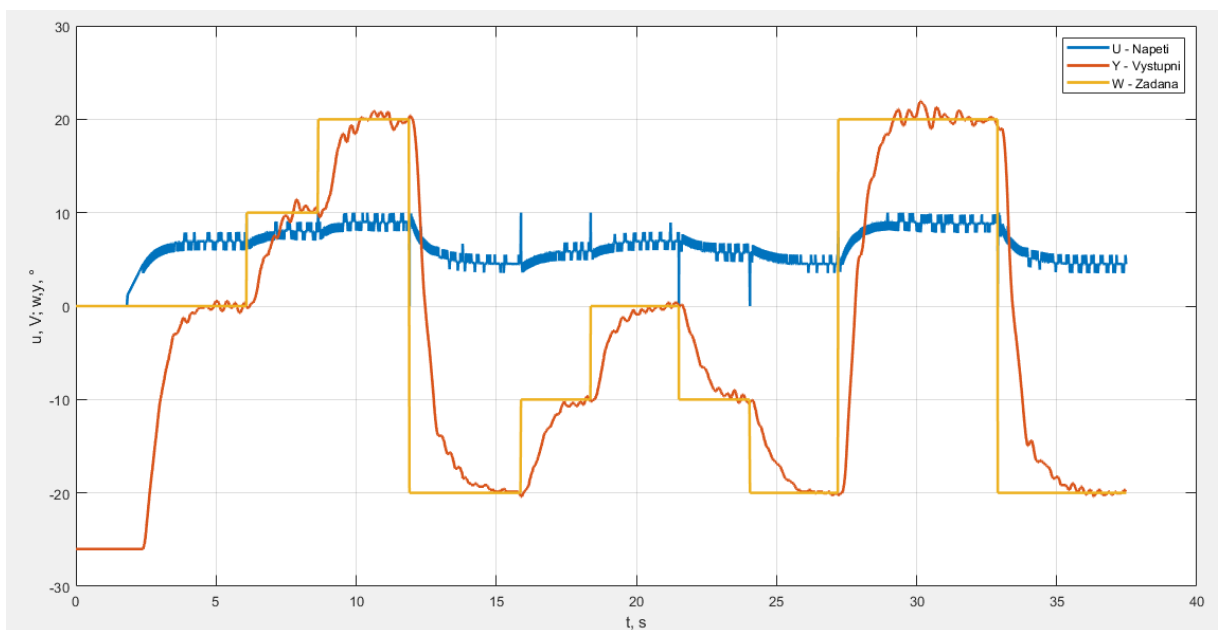


Obrázek 10.9 – Chování reálné soustavy při ručním ladění, I složka = 0,9

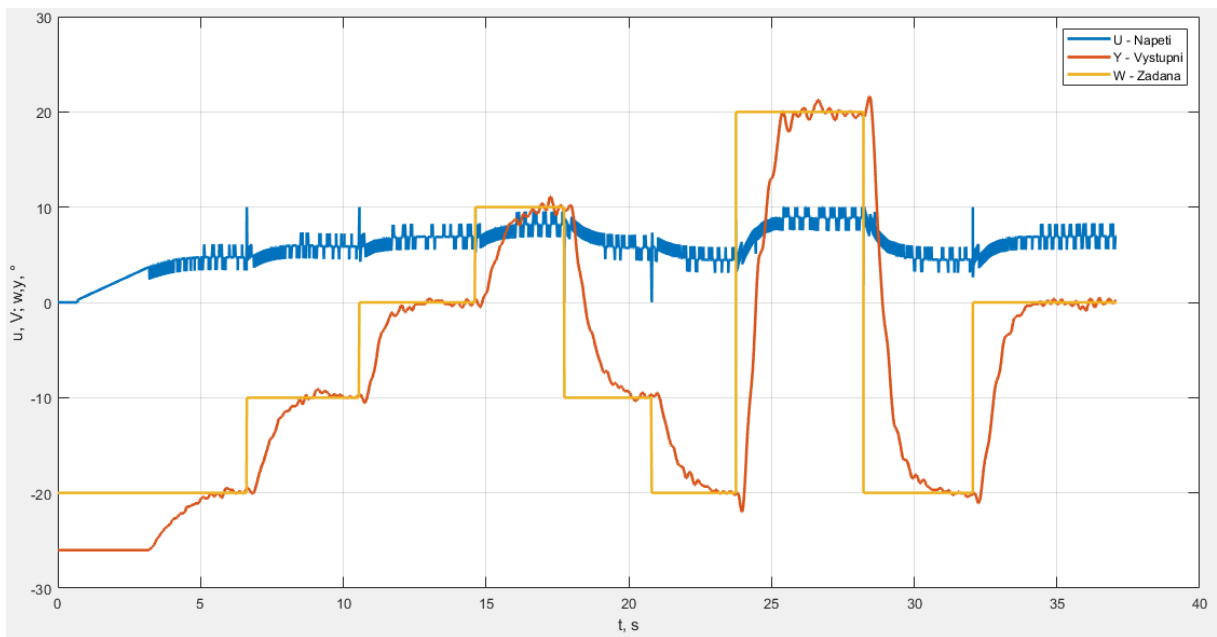
Dále, při $I = 0,9$ se provádí zvýšení D složky.



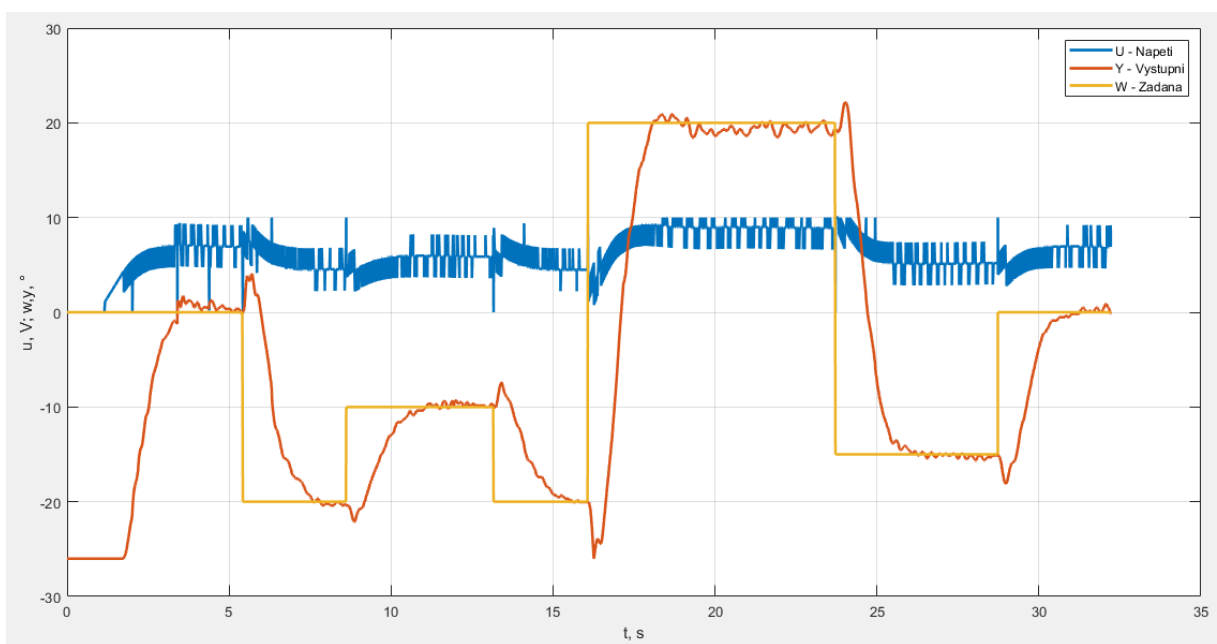
Obrázek 10.10 – Chování reálné soustavy při ručním ladění, D složka = 0,8



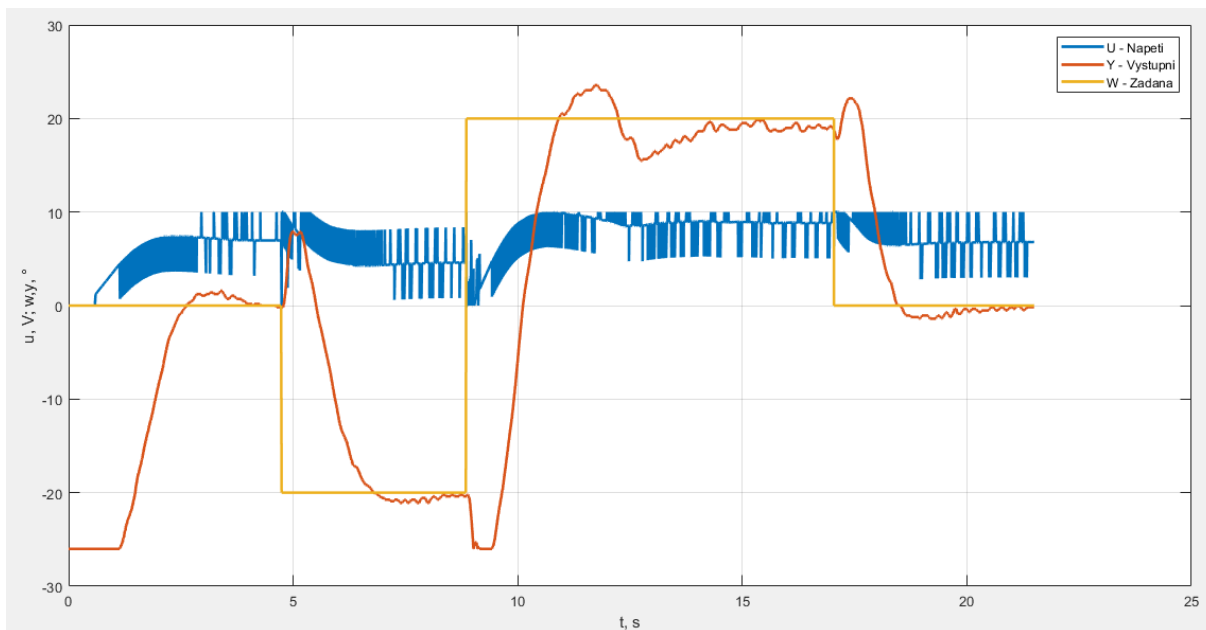
Obrázek 10.11 – Chování reálné soustavy při ručním ladění, D složka = 1,3



Obrázek 10.12 – Chování reálné soustavy při ručním ladění, D složka = 1,8

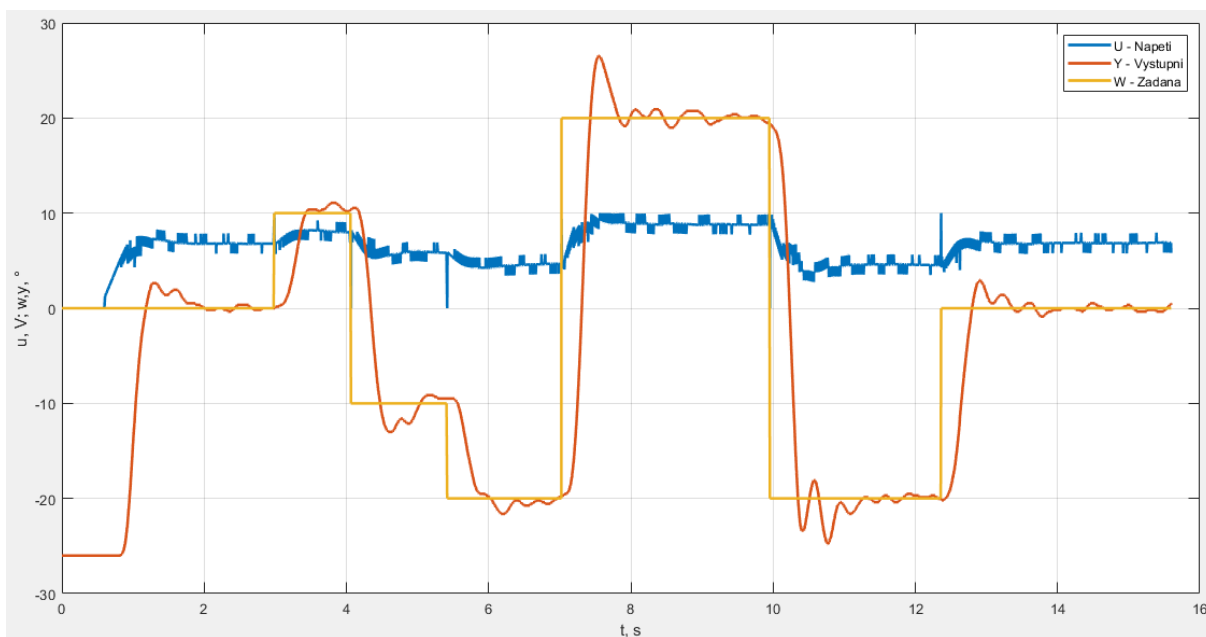


Obrázek 10.13 – Chování reálné soustavy při ručním ladění, D složka = 3

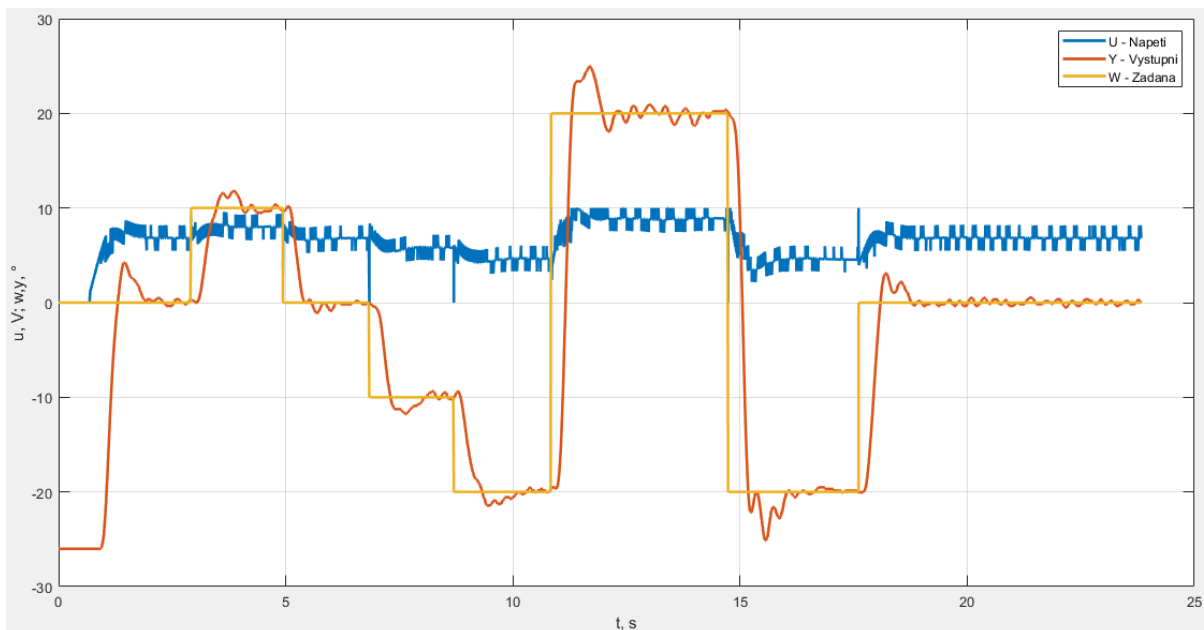


Obrázek 10.14 – Chování reálné soustavy při ručním ladění, D složka = 5

Další nastavení I a úprava D komponenty pro zvýšení rychlosti regulace.



Obrázek 10.15 – Chování reálné soustavy při ručním ladění, verze 1: I=0,3 D=1,4



Obrázek 10.16 – Chování reálné soustavy při ručním ladění, verze 2: $I=0,3$ $D=1,8$

Změna hodnoty P složky jako celku poněkud změnila dynamiku systému, změnila rychlost odezvy a stabilitu, ale v podstatě to vedlo především k nutnosti překonfigurovat I a D složek pro stabilizaci chodu regulátoru s novým parametrem P , a nepřinesla znatelné pozitivní změny v regulačním procesu.

Tabulka 10.1 – Ruční doladění, I složka

P	I	D	Dynamika systému
0,0430	Méně 0,9	0,1706	System je nestabilní
0,0430	0,9-2,3	0,1706	Středně stabilní chování a reakční rychlost s výrazným kolísáním
0,0430	Více 2,3	0,1706	Stabilní chování s velmi nízkou fluktuací a velmi pomalou odezvou

Snížení I složek zvyšuje výkon systému, ale snižuje stabilitu a zvyšuje kmitavost. Zesílení zpomaluje věci, ale snižuje jitter.

Tabulka 10.2 – Ruční doladění, D složka, dopad na překmit

P	I	D	Dynamika systému
0,0430	0,9	1,4 a méně	Bez překmitu a předkmitu
0,0430	0,9	Více 1,4	Vzniká předkmit
0,0430	0,9	Více 3	Vzniká překmit

Malá D složka dává výrazně oscilační charakter změny výstupní hodnoty $y(t)$, když se mění požadovaná, její zvýšení dělá změnu plynulou, také stabilizuje systém, ale od určitého okamžiku vede k předregulaci a přeregulování.

Tabulka 10.3 – Ruční doladění, D složka, vliv na hladkost změny $y(t)$

P	I	D	Dynamika systému
0,0430	0,9	Méně 0,8	Má znatelné výkyvy při změně $w(t)$
0,0430	0,9	0,8-1,8	Má mírné, postupně se snižující výkyvy
0,0430	0,9	Více 1,8	Změna $y(t)$ probíhá hladce, vlastně bez výkyvů

Zvýšení D nad 0,8 dává relativně plynulé, bez kolísání, ovládání, při $D=1,8$ je ovládání velmi plynulé a zároveň bez výrazné předregulace.

Doporučené parametry ladění pro PID regulátor v závislosti na požadované dynamice, včetně těch, které byly získány během konečného ručního ladění.

Tabulka 10.4 – Konečné doporučené parametry

P	I	D	Dynamika systému
0,0430	2,3	0,17	TL metoda, ukazuje pomalé řízení s nízkými oscilacemi, když je $w(t)$ menší než 0, a vysokými oscilacemi, když je $w(t)$ větší než 0
0,0430	0,9-2	0,17	Ukázka zvyšování rychlosti a kmitání systému, až do překročení meze stability
0,0430	0,9	0,17-1,3	Ukázka stabilizace systému a snížení oscilací, snížení rychlosti regulace a zlepšení stability
0,0430	0,9	Více 1,8	Demonstrace předregulace, pomalá regulace, stabilita
0,0430	0,9	Více 1,8	Ukázka překmitu, pomalé regulace, stabilita
0,0430	0,3-0,9	1,8	Ukázka výrazného zvýšení výkonu systému zvýšením projevu jeho oscilačního charakteru
0,0430	0,3	1,4-1,8	Rychlý, stabilní systém s malými překmity

Řadu těchto parametrů lze teoreticky využít i pro řízení reálného systému, zejména pro případy, kdy je požadována vysoká plynulost změny výstupní hodnoty $y(t)$ a není vyžadována rychlá regulace, a poslední možnost nastavení parametrů regulátoru poskytuje velmi vysokou rychlost regulace a vysokou stabilitu, což je dobré z hlediska regulace i reálného zařízení, při absenci specifických požadavků. Ale protože se však pracovalo na ovládní laboratorního výukového zařízení a pro výukové účely, byla navržena i řada dalších možností ladění komponent PID regulátoru a bylo popsáno odpovídající chování systému v rámci studia dynamiky PID regulátoru, a odpovídající chování systému je popsáno v rámci výzkumu dynamiky laboratorního zařízení i pro možnost jejich využití k demonstraci různých možných variant dynamiky téhož systému a vlivu parametrů regulátoru. Na něm, které lze použít k demonstraci pro vzdělávací účely.

Obecně lze konstatovat, že může mít smysl volit parametry pro regulaci systému v závislosti na požadavcích uživatele, s ohledem na požadovaný překmit, předregulaci, plynulost regulace při změně $w(t)$, v určitých mezích stability systému. Na základě výše uvedených informací je možné nastavit regulátor tak, aby řídil chování systému nejlépe, jak uživatel požaduje, a umožnil mu tak volbu její požadované dynamiky.

ZÁVĚR

Vzhledem k technické implementaci laboratorního zařízení Quanser QNET Physics and Dynamics Board, vyžadované podmínkami zadání práce, a která regulaci technicky neumožňuje, bylo použito jiné laboratorní vybavení.

V průběhu plnění úkolu diplomové práce bylo realizováno řízení laboratorního systému QNET 2.0 Vertical Take-off and Landing (VTOL) Board pomocí PID regulátoru pomocí programu vytvořeného v programovacím prostředí LabVIEW. Na základě matematické analýzy systému, použití různých metod ladění PID regulátoru byly nalezeny parametry vhodné pro realizaci regulace laboratorním systémem. Pro stabilizaci a urychlení dosazení parametrů na požadované hodnoty byla provedena dodatečná ruční úprava a úprava parametrů. To bylo také spojeno se studiem dynamiky systému.

V důsledku toho byly intervaly parametrů regulátoru stanoveny ve stabilní oblasti laboratorního vybavení s různou dynamikou chování systému, což umožňuje vybrat požadovanou možnost regulace a dává představu jak o chování použitého laboratorního zařízení s různým nastavením regulátoru, tak o vlivu různých komponent PID regulátoru na dynamiku chování systému.

POUZITÁ LITERATURA

BALÁTĚ, J., 2004. *Automatické řízení*. Praha: BEN – technická literatura. [cit. 2023-05-10]. ISBN 80-7300-148-9. Dostupné z: http://utb.tsx.cz/Automaticke_rizeni.pdf

Data v LabVIEW [online]. In StudFile.net [cit. 2023-05-10]. Dostupné z: <https://studfile.net/preview/5759812/>

DUŠEK, František, 2023. *Spojité řízení*. [Studijní text.]

HALVORSEN, HANS-PETTER, 2017. *Control and Simulation in LabVIEW* [online]. [cit. 2023-04-24]. Dostupné z: <https://www.halvorsen.blog/documents/tutorials/resources/Control%20and%20Simulation%20in%20LabVIEW.pdf>

HONC, Daniel, 2018. *Automatizace*. [Studijní text.]

KUPKA, Libor, 2019. *Teorie automatického řízení 2*. [Studijní text.]

KUPKA, Libor, 2023. *Diskrétní řízení*. [Studijní text.]

LabVIEW – grafické programování: typy dat [online]. [cit. 2023-05-10]. Dostupné z: <http://LabVIEW-ifit.narod.ru/page27.html>

NATIONAL INSTRUMENTS, 2014. *Quanser QNET VTOL Board 2.0 for NI ELVIS II/II+* [online]. [cit. 2023-04-19]. Dostupné z: <https://www.ni.com/cs-cz/support/model.quanser-qnet-vtol-board-2-0-for-ni-elvis-ii-ii-.html>

NATIONAL INSTRUMENTS, 2023a. *Control & Simulation Loop* [online]. [cit. 2023-04-19]. Dostupné z: https://www.ni.com/docs/en-US/bundle/labview-control-design-and-simulation-module/page/lvsim/sim_simulation_loop.html

NATIONAL INSTRUMENTS, 2023b. *Fuzzy Logic* [online]. [cit. 2023-04-19]. Dostupné z: https://www.ni.com/docs/en-US/bundle/labview/page/lvpidmain/fuzzy_main.html

NATIONAL INSTRUMENTS, 2023c. *Fuzzy Logic VIs* [online]. [cit. 2023-04-19]. Dostupné z: <https://www.ni.com/docs/en-US/bundle/labview/page/lvpid/fuzzylogicitle.html>

NATIONAL INSTRUMENTS, 2023d. *LabVIEW Control Design and Simulation Module* [online]. [cit. 2023-04-23]. Dostupné z: https://zone.ni.com/reference/en-XX/help/371894J-01/lvsimconcepts/sim_c_ctrl_dsgn/

NATIONAL INSTRUMENTS. *NI ELVIS II+* [online]. [cit. 2023-04-19]. Dostupné z: <https://www.ni.com/cs-cz/support/model.ni-elvis-ii-.html>

O'DWYER, Aidan 2003. *Handbook of PI and PID Controller Tuning Rules*. 1st ed. Imperial College Press. ISBN: 978-1-86094-350-0.

PECHOUŠEK, J. 2004. *Základy programování v prostředí LabVIEW*. Olomouc: Univerzita Palackého v Olomouci. 87 s. ISBN 80-244-0800-7.

Popis funkcí PID regulátoru [online]. [cit. 2023-05-01]. Dostupné z:

<https://plcontroller.ru/wp-content/uploads/2020/06/%D0%9E%D0%BF%D0%B8%D1%81%D0%B0%D0%BD%D0%B8%D0%B5-%D1%84%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8-%D0%9F%D0%98%D0%94-%D1%80%D0%B5%D0%B3%D1%83%D0%BB%D1%8F%D1%82%D0%BE%D1%80-1.pdf>

QUANSER, 2014. *QNET 2.0 VTOL Board* [online]. [cit. 2023-04-19].

Dostupné z: <https://www.quanser.com/products/qnet-2-0-vtol-board/>

QUANSER, 2015. *QNET Physics and Actuators - User Manual* [online]. [cit. 2023-04-19].

Dostupné z: <https://www.quanser.com/products/qnet-physics-dynamics-board/>

VÍTEČKOVÁ, M. a ANTONÍN Viteček, 2011. *Vybrané metody seřizování regulátorů* [online]. Nakladatelství VŠB – TU Ostrava. [cit. 2023-05-10]. ISBN 978-80-248-2503-8.

Dostupné z: <http://books.fs.vsb.cz/ZRMS/vybrane-metody-serizovani-regulatoru.pdf>

VLACH, J.; HAVLÍČEK, J.; VLACH, M. 2008. *Začínáme s LabVIEW*. Praha: BEN - technická literatura. 247 s. ISBN 978-80-7300-245-9.

VŠB – TU Ostrava. *Kvalita regulačního pochodu diskrétních regulačních obvodů* [online].

[cit. 2023-05-08]. Dostupné z: <http://books.fs.vsb.cz/cislicovaregulace/data/kapitola7.html>

PŘÍLOHY

A - CD

Příloha k diplomové práci
Řízení modulárního mechanického systému
Filipp Poplavko

CD

Obsah

1. Text diplomové práce ve formátu PDF.
2. Zdrojový kód aplikace a v něm použité subVI.
3. Skripty MATLABu pro aproximaci, výpočet parametrů PID regulátoru metodami použitými v diplomové práci, a k zobrazení grafů naměřených dat.
4. Simulink schéma s relé ve zpětné vazbě a simulace chování modelu.
5. Textové soubory s uloženými údaji o chování reálného systému.
6. Návod k obsahu složek se soubory.