

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra kybernetiky

Úlohy automatického řízení pro
fyzikální model úpravný vody

DIPLOMOVÁ PRÁCE

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta aplikovaných věd
Akademický rok: 2019/2020

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Jan HEVERA**
Osobní číslo: **A17N0009P**
Studijní program: **N3902 Inženýrská informatika**
Studijní obor: **Řídicí a rozhodovací systémy**
Téma práce: **Úlohy automatického řízení pro fyzikální model úpravný vody**
Zadávající katedra: **Katedra kybernetiky**

Zásady pro vypracování

- Seznamte se s programovatelnými automaty Simatic S1200 a S1500 firmy Siemens, jejich vývojovým prostředím TIA portal [1] a funkčními bloky pro realizaci automatické regulace.
- Seznamte se se základními regulačními strukturami (jednoduchá regulační smyčka, kaskádní regulace, poměrová regulace, apod.), viz např. [2], [3].
- Prostudujte dostupnou dokumentaci fyzikálního modelu (standu) „úpravný vody“ na KKY, FAV.
- Navrhňte a v nástrojích [1] implementujte vhodné příklady regulace hladiny, tlaku, průtoku a případně též pH a vodivosti. K navrženým příkladům vytvořte též jednoduché operátorské rozhraní (HMI, human-machine interface).

Rozsah diplomové práce: **40-50**
Rozsah grafických prací: **dle potřeby**
Forma zpracování diplomové práce: **tištěná**

Seznam doporučené literatury:

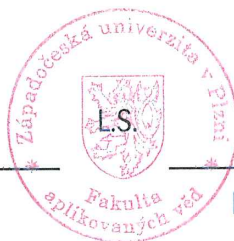
- [1] Siemens: SIMATIC STEP 7 Professional V14 – System manual. 04/2017.
- [2] Åström, K., J., Hägglund, T.: PID Controllers: Theory, Design, and Tuning, 2nd Edition, 1995.
- [3] Dale E. Reed: „The Six“ Fundamental Control Strategies Every Process Control Developer Must Know. ISA.

Vedoucí diplomové práce: **Ing. Pavel Balda, Ph.D.**
Výzkumný program 1

Datum zadání diplomové práce: **1. října 2019**
Termín odevzdání diplomové práce: **25. května 2020**



Doc. Dr. Ing. Vlasta Radová
děkanka





Prof. Ing. Josef Psutka, CSc.
vedoucí katedry

V Plzni dne 1. října 2019

PROHLÁŠENÍ

Předkládám tímto k posouzení a obhajobě diplomovou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni 18. srpna 2020

.....

PODĚKOVÁNÍ

Mé poděkování patří vedoucímu mé diplomové práce panu Ing. Pavlovi Baldovi, Ph.D. za úžasné vedení práce, odborné a užitečné rady, a také ochotu vždy si udělat čas, když jsem potřeboval poradit. Dále bych rád poděkoval mému nadřízenému v práci Ing. Petru Voborníkovi, kolegům Ondrovi Sekyrovi a Ing. Martinu Zemanovi, kteří mi ukázali jak přistupovat k programování a poskytli mi rady ohledně výukového zařízení. V neposlední řadě bych chtěl poděkovat členům své rodiny za jejich podporu během studia.

Anotace

Tato práce se zabývá návrhem řídicího systému fyzikálního modelu úpravní vody. V práci je popsána architektura modelu a uveden je přehled jednotlivých komponent řídicího systému. Následuje popis použitého vývojového softwaru a aplikace implementovaných funkčních bloků. Dále jsou vypsány vytvořené funkční bloky pro účely obsluhy komponent konstrukce a způsob výměny dat mezi PLC a HMI. Poté jsou vyobrazeny jednotlivé obrazovky HMI určené pro obsluhu modelu. Těžištěm práce jsou jednotlivé regulační úlohy navržené pro demonstraci funkce modelu s následnými experimenty uvedenými na konci.

Klíčová slova: PLC, HMI, TIA Portal, čerpadla, senzory, PID regulace

Annotation

This thesis deals with the design of a control system for the physical model of a water treatment plant. The thesis describes the architecture of the model and gives an overview of individual components of the control system. Next follows description of used development software and the application of the implemented function blocks. Furthermore the created function blocks for the purpose of operating the components of physical model are described, followed by listed methods of data exchange between the PLC and the HMI. In addition, the individual HMI screens dedicated for operating the model are described. The focus of the work are individual control tasks designed to demonstrate the function of the model with subsequent experiments which are presented at the end of this thesis.

Key words: PLC, HMI, TIA Portal, water pumps, sensors, PID control

Obsah

Úvod	1
0.1 Výchozí stav	1
0.2 Cíl práce	2
1 Konstrukce zařízení	3
1.1 Nádrže	4
1.2 Čerpadla	5
1.3 Senzory	5
1.3.1 Měření průtoku	6
1.3.2 Měření tlaku	7
1.3.3 Měření vodivosti	8
1.3.4 Měření pH	9
2 Komponenty řídicího systému	10
2.1 Programovatelné automaty	11
2.1.1 Siemens S7-1200	11
2.1.2 Siemens S7-1500	12
2.2 Frekvenční měnič	12
2.3 Operátorské panely	13
2.3.1 KTP700 Basic	14
2.3.2 TP700 Comfort	14
3 Vývojový software	15
3.1 Funkční bloky PUT a GET	16
3.2 Funkční blok SinaSpeed	17
3.3 Funkční blok USS_DRV	18

3.4	Funkční blok PID_Compact	19
3.4.1	Pre-tuning	21
3.4.2	Fine tuning	21
4	Funkční bloky pro manipulaci	22
4.1	Funkční blok pro chyby	22
4.1.1	Datový blok pro chyby	23
4.2	Funkční blok pro měření	24
4.2.1	Datový blok pro měření	25
4.3	Funkční blok pro čerpadlo	27
4.3.1	Datový blok pro čerpadlo	28
4.4	Funkční blok pro regulátor	30
4.5	Funkční blok pro generátor signálu	31
4.5.1	Datový blok pro generátor signálu	31
5	Data a datové rozhraní	33
5.1	Datový blok pro všechna data PLC	34
5.1.1	Datová struktura General	34
5.2	Datový blok pro HMI	36
5.3	Funkční blok pro předávání dat mezi PLC a HMI	36
5.4	Předávání dat mezi dvěma PLC	38
6	Prostředí operátorského panelu	40
6.1	Hlavní obrazovka	41
6.2	Pop-up okno pro měření	44
6.3	Pop-up okno pro čerpadlo	45
6.4	Obrazovka pro úlohu regulace hladiny	46
6.5	Obrazovka pro úlohu regulace průtoku	47
6.6	Obrazovka pro úlohu regulace tlaku	48
6.7	Obrazovka pro regulátor	48
6.8	Obrazovka pro generátor signálu	49
7	Regulační úlohy	51
7.1	Regulace hladiny	52

7.1.1	Průběh Pre-tuningu	53
7.1.2	Průběh Fine tuningu	54
7.1.3	Stanovené konstanty	54
7.1.4	Skoková chyba regulace	55
7.1.5	Obdélníková chyba regulace	56
7.1.6	Sinusová chyba regulace	57
7.2	Regulace průtoku	57
7.2.1	Průběh Pre-tuningu	58
7.2.2	Průběh Fine tuningu	58
7.2.3	Stanovené konstanty	59
7.2.4	Regulace na skokovou změnu požadované hodnoty	60
7.3	Regulace tlaku	60
7.3.1	Průběh Pre-tuningu	61
7.3.2	Průběh Fine tuningu	62
7.3.3	Stanovené konstanty	62
7.3.4	Regulace na skokovou změnu požadované hodnoty s parametry z Fine tuningu	63
7.3.5	Regulace na skokovou změnu požadované hodnoty s uprave- nými parametry z Fine tuningu	63
8	Závěr	65

Seznam obrázků

1	Fyzikální model úpravny vody	2
1.1	3D model konstrukce výukového zařízení [19]	3
1.2	Polyethylenová nádrž [9]	5
1.3	Čerpadlo značky Gundfos [4]	5
1.4	Přehled instalovaných senzorů	6
1.5	Princip Faradayova indukčního zákona [13]	6
1.6	Princip tlakového senzoru [1]	7
1.7	Princip dvouelektrodového měření vodivosti [5]	8
1.8	Obecné schéma pH sondy [3]	9
2.1	Diagram propojení komponent řídicího systému s komponentami konstrukce zařízení	10
2.2	SIMATIC S7-1200 CPU 1214C [16]	11
2.3	SIMATIC S7-1500 CPU 1512C [17]	12
2.4	Sinamics G120C [18]	13
2.5	SIMATIC HMI KTP700 Basic [14]	14
2.6	SIMATIC HMI TP700 Comfort [15]	14
3.1	Funkční blok PUT	16
3.2	Funkční blok GET	17
3.3	Funkční blok SinaSpeed	17
3.4	Funkční blok USS_DRV	18
3.5	Funkční blok PID_Compact	20
4.1	Funkční blok pro manipulaci s chybami FB_Faultx8	23
4.2	Data bloku FB_Faultx8	24

4.3	Funkční blok pro manipulaci s měřením FB_Measurement	25
4.4	Data bloku FB_Measurement	26
4.5	Funkční blok pro manipulaci s čerpadlem FB_Pump	27
4.6	Data bloku FB_Pump	29
4.7	Funkční blok pro regulátor FB_PIDControl	30
4.8	Funkční blok pro generování signálu FB_SignalGenerator	31
4.9	Data bloku FB_SignalGenerator	32
5.1	UDT pro měření	33
5.2	Datový blok DB_GlobalPLC	34
5.3	Datová struktura řídicích veličin UDT_General	35
5.4	Datový blok DB_HMI	36
5.5	Funkční blok FB_MeasurementHandler	37
5.6	Princip předávání dat	38
5.7	Data odesílána z PLC S7-1500 do S7-1200	39
6.1	Struktura obrazovek	40
6.2	Hlavní obrazovka na operátorském panelu TP700C	41
6.3	Hlavní obrazovka na operátorském panelu KTP700B	43
6.4	Pop-up okno pro měření	44
6.5	Pop-up okno pro čerpadlo	45
6.6	Obrazovka pro úlohu regulace hladiny	46
6.7	Obrazovka pro úlohu regulace průtoku	47
6.8	Obrazovka pro úlohu regulace tlaku	48
6.9	Výchozí obrazovka pro nastavení regulátoru	49
6.10	Výchozí obrazovka pro nastavení generátoru signálu	50
7.1	Regulační smyčka [6]	51
7.2	Průběh Pre-tuningu u úlohy regulace hladiny	53
7.3	Průběh Fine tuningu u úlohy regulace hladiny	54
7.4	Průběh regulace na skokový signál chyby	55
7.5	Průběh regulace na obdélníkový signál chyby	56
7.6	Průběh regulace na sinusový signál chyby	57

7.7	Průběh Pre-tuningu u úlohy regulace průtoku	58
7.8	Průběh Fine tuningu u úlohy regulace průtoku	59
7.9	Průběh regulace se skokovou změnou požadovaného průtoku	60
7.10	Průběh Pre-tuningu u úlohy regulace tlaku	61
7.11	Průběh Fine tuningu u úlohy regulace tlaku	62
7.12	Průběh regulace se skokovou změnou požadovaného tlaku (parametry z Fine tuningu)	63
7.13	Průběh regulace se skokovou změnou požadovaného tlaku (upravené parametry z Fine tuningu)	64

Úvod

Úpravna vody je podle určené aplikace komplex procesů, které svým chováním mění parametry vody na požadované hodnoty. Pro měření parametrů může být použita celá řada senzorů od měření pH, vodivosti až po měření hodnoty chemických látek obsažených ve vodě, jako je například měření kyseliny peroctové, peroxidu vodíku, atd. Úprava vody je nezbytnou součástí mnoha průmyslových aplikací. Vodu s požadovanými parametry vyžadují například elektrárny, veřejné a soukromé bazény, čističky vod, potravinové průmysly, chemické průmysly, atd. Úpravou vody se rozumí procesy jako je čiření a sedimentace, filtrace, odstranění organických látek, odstranění minerálů a v neposlední řadě odstranění mikroorganismů. V posledních letech se v některých zemích zvyšuje trend výstavby vodáren na přeměnu mořské vody na pitnou, z důvodu jejího nedostatku. Vodárny bývají automatizované a k jejich řízení se můžou využívat programovatelné logické automaty od celé řady výrobců.

0.1 Výchozí stav

První část práce se zabývá popisem fyzikálního modelu úpravny vody. Model byl dodán na Západočeskou univerzitu v Plzni pro účely výuky od firmy ProMinent Systems spol. s r. o. Obsahuje dvě vodní nádrže, které bylo nutné před začátkem práce napustit čistou vodou pomocí vstupního ventilu. Je nutné podotknout, že se k modelu přistupuje jako k izolované soustavě, to tedy znamená, že po napuštění už není dodávána čerstvá voda a také není voda odčerpávána z modelu na odpad. Probíhá zde cirkulace vody mezi dvěma nádržemi. Pro cirkulaci vody jsou používána odstředivá čerpadla ovládaná frekvenčními měniči, které bylo nutné pro správnou funkčnost nastavit. Po zprovoznění všech nezbytných komponent potřebných pro úlohy regulace bylo možné napsat řídicí program ve vývojem prostředí TIA Portal pro logické automaty od firmy Siemens.



Obrázek 1: Fyzikální model úpravy vody

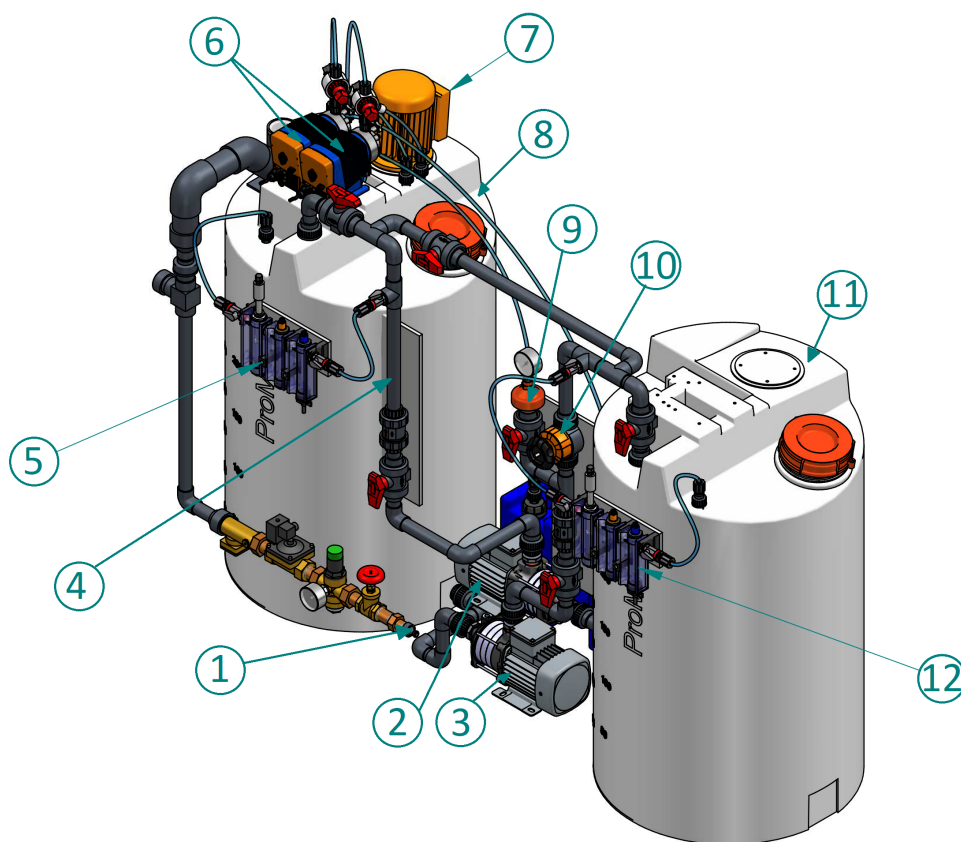
0.2 Cíl práce

Cílem této práce je návrh řídicího systému uvedeného fyzikálního modelu, díky němuž bude možné demonstrovat jeho chod pro výukové účely. Aby bylo možné navrhnout funkční řídicí systém je nutné se seznámit s celou konstrukcí modelu a jednotlivými komponentami řídicího systému. Dalším krokem je návrh funkčních bloků, které budou jednotlivé komponenty konstrukce ovládat. Pro přehled o aktuálním stavu modelu musí být vytvořeny ovládací obrazovky pro operátorské panely. Všechny tyto úkony vedou k poslední stěžejní části této práce a tou je návrh regulačních úloh, na kterých bude možné předvádět chod modelu úpravy vody.

Kapitola 1

Konstrukce zařízení

Tato kapitola se zabývá konstrukcí dodaného výukového zařízení a popisem jeho jednotlivých komponent. Při návrhu regulačních úloh bylo nutné zohlednit všechny dostupné komponenty a jejich potenciální využití. Jednotlivé komponenty jsou propojeny s PLC umístěnými v rozvaděči (viz kapitola 2). Konstrukce každého zařízení se odvíjí od požadované funkce, její návrh musí zajistit splnění technologického cíle. Dále se v mnoha případech musí klást větší důraz na bezpečnost provozovaného zařízení a tedy i začlenění nezbytných komponent, které ji zajistí. Konstrukce fyzikálního modelu úpravy vody je vykreslena na obr. 1.1.



Obrázek 1.1: 3D model konstrukce výukového zařízení [19]

Z obrázku lze vidět, že konstrukci dominují dvě plastové nádrže. Mezi každou nádrží je umístěno čerpadlo, které zajišťuje možnost čerpání vody z jedné nádrže do druhé. Na každé nádrži jsou umístěny senzory monitorující aktuální vodivost a pH vody dané nádrže. V jedné nádrži je také možné provést míchání pomocí míchadla poháněného motorem umístěném v horní části nádrže. Aby mělo míchadlo smysl, je možné do nádrže dávkovat například chemikálie pomocí dávkovacích čerpadel z méně objemných modrých plastových nádrží umístěných za velkými bílými nádržemi. Po nadávkování chemikálie do vody v nádrži je možné zmíněným míchadlem promíchat její obsah. Pro účely výuky byl mezi nádrže v jednom směru umístěn průtokoměr a v druhém směru senzor tlaku. O měření hladiny v každé nádrži se starají také tlakové senzory. Aby bylo možné kontrolovat možné přetečení nádrže, jsou v horní části každé nádrže umístěny senzory, které detekují hladinu a posílají digitální signál do PLC k vyhodnocení. Pro snadnější orientaci v následujícím textu byl každé komponentě přiřazen anglický název, který se nachází i fyzicky na štítku samotného zařízení. Popis komponent z obr. 1.1 a název některých z nich je popsán v tabulce 1.1.

Číslo na obrázku 1.1	Angl. pojmenování	Popis
1	–	Přívod vody
2	Pump M3	Čerpadlo z nádrže 2 do nádrže 1
3	Pump M2	Čerpadlo z nádrže 1 do nádrže 2
4	Flowmeter	Průtokoměr z nádrže 2 do nádrže 1
5	–	Měření vodivosti a pH v nádrži 1
6	Dosing pump M4 Dosing pump M5	Dávkovací čerpadla do nádrže 1
7	Stirrer M1	Míchadlo v nádrži 1
8	Tank 1	Vodní nádrž číslo 1
9	Pressure	Tlakoměr mezi nádržmi 1 a 2
10	–	Manuální ventil pro přiškrčení průtoku
11	Tank 2	Vodní nádrž číslo 2
12	–	Měření vodivosti a pH v nádrži 2

Tabulka 1.1: Přehled komponent

1.1 Nádrže

Přítomné nádrže mají objem 300 l a jsou vyrobeny z polyethylenu, který je osvědčený proti chemikáliím [9]. Jejich umístění na zařízení, které je určeno pro výukové účely je však více než dostačující, jelikož se zařízení nachází v prostorách univerzity a je spíše nepravděpodobné, že by kdy bylo použito vysoce agresivních chemických látek. V horní části nádrže *Tank 1* se také nachází míchadlo poháněné motorem, který je spuštěn digitálním výstupem PLC S7-1200. Je důležité aby míchadlo při spuštění motoru bylo ponořené. Pokud by došlo ke spuštění neponořeného míchadla, nebyl by to takový problém za podmínky, že hladina v nádrži se nezvyšuje. Zvyšující se hladina vody by po kontaktu se zapnutým míchadlem způsobila poškození

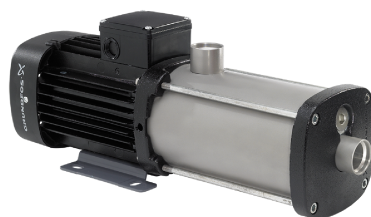
míchadla v nádrži. Proto je nutné nejdříve napustit vodu nad úroveň míchadla a až poté umožnit jeho spuštění.



Obrázek 1.2: Polyethylenová nádrž [9]

1.2 Čerpadla

Přítomná čerpadla jsou určena k čerpání vody mezi nádržemi. Na zařízení jsou umístěna dvě odstředivá čerpadla s axiálním sáním a radiálním výstupem značky Grundfos o jmenovitém výkonu 0,46 kW. Čerpadlo je vybaveno 3 fázovým asynchronním motorem chlazeným ventilátorem. Odstředivé čerpadlo funguje na jednoduchém principu. Na střed kola se zahnutými lopatkami umístěným na motorem hnané hřídele přitéká kapalina sacím otvorem. Vzniká odstředivá energie, která odmršťuje kapalinu výstupním otvorem čerpadla ven. Axiální sání znamená ve směru osy hřídele. Radiální výstup pak znamená kolmo k ose hřídele [20]. Při nastavování frekvenčního měniče čerpadla bylo nutné zadat parametry každého čerpadla, jako je jmenovitý výkon (0,46 kW), jmenovité otáčky (2770-2820 rpm), jmenovitý proud (2 A) a frekvence (50 Hz) [4].



Obrázek 1.3: Čerpadlo značky Grundfos [4]

1.3 Senzory

Senzor je prvek, který slouží pro předávání hodnot měřené fyzikální veličiny vyjádřené elektrickým signálem do řídicího systému. Na základě těchto hodnot je pak vytvořen specifický akční zásah systému. Na výukovém zařízení se nachází celkem 12 senzorů. Kromě analogových senzorů se zde vyskytují i digitální senzory. Na každé nádrži se nachází digitální senzor na vrchu nádrže pro kontrolu přetečení. Ve spodní

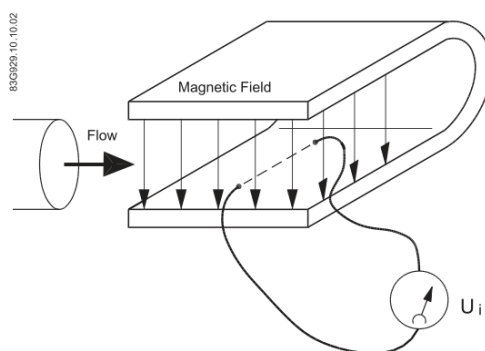
části části nádrže se nachází analogový senzor pro měření tlaku. Dále je v každé nádrži přítomen analogový senzor pro měření vodivosti a pH vody, včetně digitálního senzoru pro kontrolu přítomnosti vody na jmenovaných analogových senzorech, aby byla zajištěna správnost měření. Na potrubí mezi nádržemi se nachází analogový senzor pro měření průtoku jedním směrem a na potrubí druhým směrem se nachází analogový senzor pro měření tlaku. Přehled senzorů je uveden na obr. 1.4a až 1.4d.



Obrázek 1.4: Přehled instalovaných senzorů

1.3.1 Měření průtoku

Senzor pro měření průtoku je umístěn na potrubí vedoucí z nádrže *Tank 2* do nádrže *Tank 1*. Místo umístění průtokoměru je označeno číslem 4 na obrázku 1.1. Jedná se o magnetický senzor průtoku SITRANS MAG 5000 od firmy Siemens (viz obrázek 1.4a). Jak již z název napovídá, princip průtokoměru je založený na Faradayově zákoně elektromagnetické indukce:



Obrázek 1.5: Princip Faradayova indukčního zákona [13]

Průtokoměr obsahuje elektromagnet který generuje elektromagnetické pole. Dále jsou v magnetickém poli umístěny kolmo na směr proudění a směr elektromagnetic-

kého pole měřicí elektrody. Průtok vody generuje napětí určené následujícím vztahem:

$$U_i = L \times B \times v = k_1 \times k_2 \times v, \quad (1.1)$$

kde

U_i : indukované napětí na elektrodách,

L : délka vodiče = vnitřní průměr trubky = k_1 ,

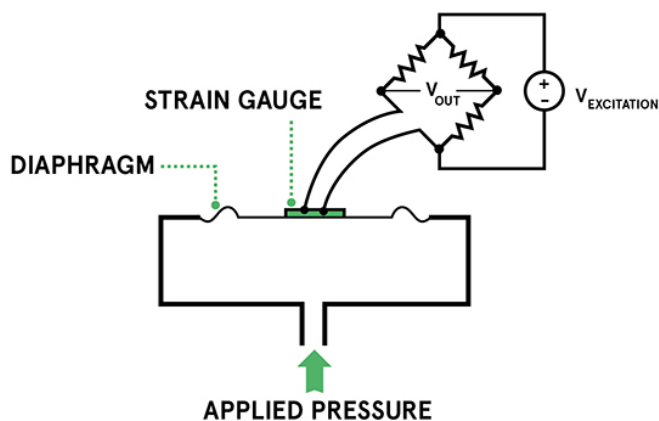
B : velikost generovaného magnetického pole = k_2 ,

v : rychlost proudící kapaliny.

Ze vztahu 1.1 vyplývá, že velikost indukovaného napětí je přímo úměrná rychlosti proudící kapaliny. Aby bylo indukováno napětí proudící kapalinou, musí být kapalina elektricky vodivá a v případě daného senzoru musí být její vodivost alespoň $5 \mu S/cm$. Velkou výhodou tohoto senzoru je bezkontaktní měření. Senzory tohoto typu neobsahují žádné mechanické části, opotřebení je tedy minimální. Další výhodou je, že měření neovlivňuje tlak, hustota ani teplota kapaliny [13].

1.3.2 Měření tlaku

Výukový model obsahuje tři senzory SITRANS P220 (obrázek 1.4b) od firmy Siemens, určené pro měření hydrostatického tlaku v nádržích *Tank 1*, *Tank 2* a dále k měření hydrostatického tlaku za čerpadlem *Pump M2*. Tento senzor má piezorezistentní buňku umístěnou na membráně v pouzdru z nerezové oceli. Jedná se o tenzometrický typ senzoru, tedy typ používaný k nepřímému měření mechanického napětí na povrchu součásti prostřednictvím měření její deformace. Princip je lze vidět na obrázku 1.6.



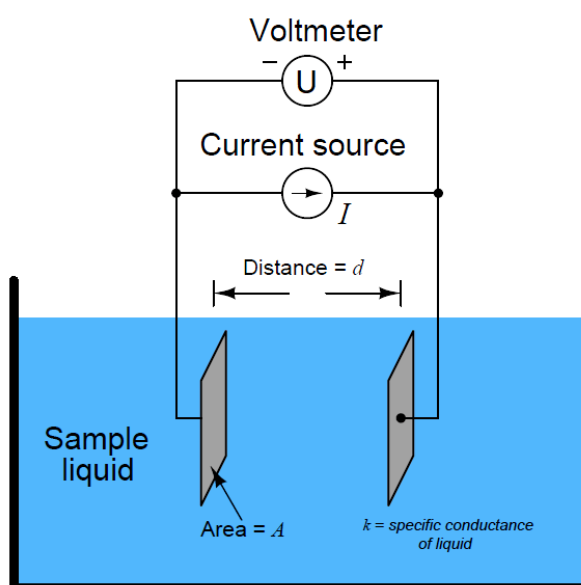
Obrázek 1.6: Princip tlakového senzoru [1]

Základním principem tohoto senzoru je změna elektrického odporu materiálu snímače umístěného na membráně. Při působení mechanického namáhání na plochu snímače dochází ke změně pohyblivosti nosičů náboje, tím pádem se mění odpor

senzoru, a tedy i elektrická vodivost. Změna odporu ve snímači se obvykle měří pomocí Wheatstoneova měřicího můstku, jenž umožňuje převést malé změny odporu snímače na výstupní napětí [1].

1.3.3 Měření vodivosti

Vodivost reprezentuje schopnost kapaliny vést elektrický proud. Pro vodu platí, že čím chemicky čistší je, tím menší je její elektrolytická vodivost. Kromě měření pH se jedná o další způsob jak kontrolovat dávkování a ředění chemikálií. Velké využití má tento typ senzorů při kontrole čistoty vody po procesu čištění filtrací či reverzní osmózou. Senzor pro měření elektrolytické vodivosti nainstalovaný na výukovém zařízení je zobrazen na obrázku 1.4c. Tento senzor funguje na principu měření protékaného proudu mezi dvěma elektrodami vykresleného na 1.7.



Obrázek 1.7: Princip dvouelektrodového měření vodivosti [5]

Senzor obsahuje dvě kovové elektrody napájené proudovým zdrojem. Kationty se pohybují směrem k záporné elektrodě a anionty se pohybují směrem ke kladné elektrodě. Čím vyšší je počet volných nosičů náboje, tím větší je protékaný proud a vodivost [5]. Pro výpočet vodivosti kapaliny platí:

$$k = \frac{G \cdot d}{A}, \quad (1.2)$$

kde

k : specifická vodivost kapaliny [S/cm],

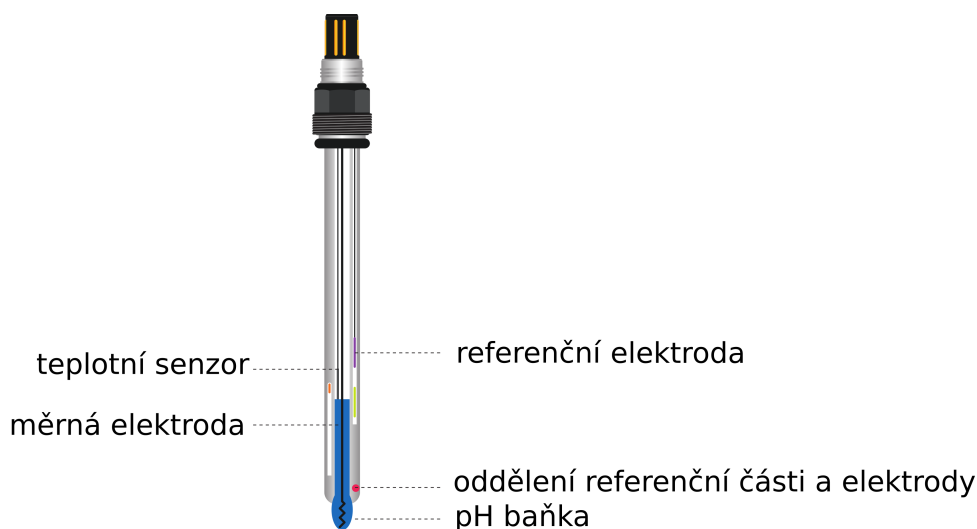
G : vodivost dána $G = 1/R[S]$, kde $R = U/I$,

d : vzdálenost mezi elektrodami [cm],

A : plocha elektrod [cm^2].

1.3.4 Měření pH

Stupnice pH (vodíkový exponent) udává kyselost prostředí podle koncentrace vodíkových iontů $[H_3O^+]$. Hodnoty pH menší než 7 (koncentrace $[H_3O^+] > 10^{-7} mol/l$) jsou označovány jako kyselé roztoky, hodnoty pH vyšší než 7 (koncentrace $[H_3O^+] < 10^{-7} mol/l$) se označují jako roztoky zásadité. Jako neutrální roztok se označuje roztok s hodnotou pH 7 (koncentrace $[H_3O^+] = 10^{-7} mol/l$) [2]. Hodnota pH se měří například u vody v bazénu, jelikož její správná hodnota zajišťuje, že bude bazénová voda čistá. Na výukovém zařízení je možné měřit hodnotu pH v obou nádržích. K měření pH zde slouží senzor DULCOTEST® PHER 112 SE od firmy ProMinent (viz obrázek 1.4d). Princip pH sondy bude vysvětlen na obecném schématu:



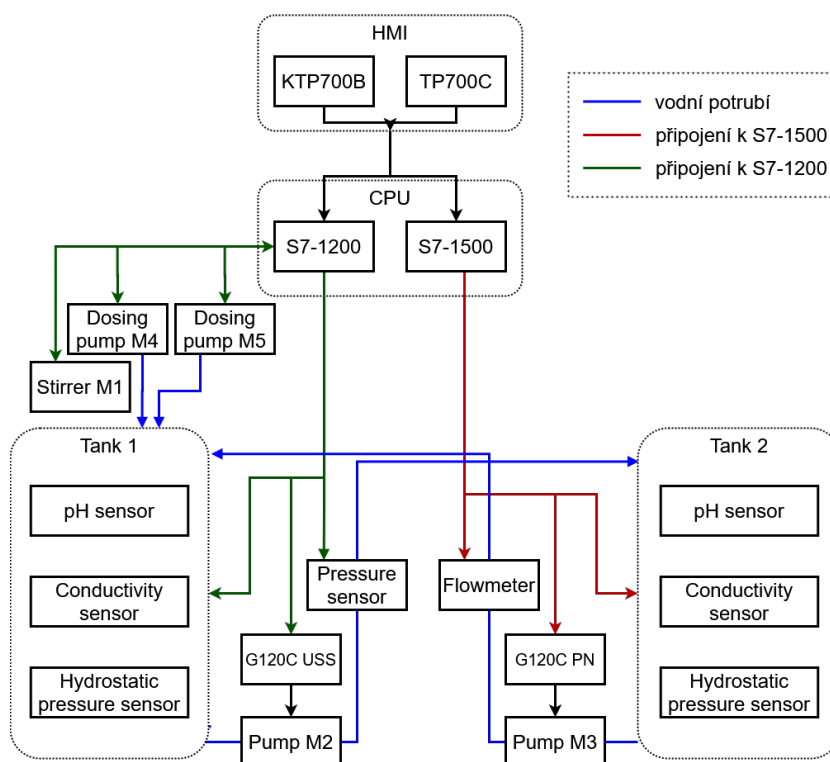
Obrázek 1.8: Obecné schéma pH sondy [3]

Sonda je založena na potenciometrickém měření pH. Obsahuje měrnou a referenční elektrodu. Referenční elektroda má konstantní potenciál. Potenciál měrné elektrody je proměnný a je funkcí aktivity vodíkových iontů, závisí tedy na pH. Měří se tedy potenciometrický rozdíl mezi měrnou a referenční elektrodou. Na měření pH má vliv okolní teplota, proto bývá u některých senzorů umístěno teplotní čidlo, které lze pak použít pro teplotní kompenzaci. Část označená jako pH baňka slouží pro oddělení měrné elektrody a měřené kapaliny [3].

Kapitola 2

Komponenty řídicího systému

V předchozí kapitole byla popsána konstrukce a její jednotlivé komponenty. V této kapitole budou popsány komponenty řídicího systému, tzn. komponenty, které jsou důležité pro chod celého výukového zařízení. „Mozek“ celého zařízení jsou dva programovatelné automaty (PLC = Programmable Logic Controller) Siemens S7-1200 a S7-1500. Rozhraní mezi operátorem a strojem (HMI = Human Machine Interface) zajišťují operátorské panely Siemens KTP700 Basic a TP700 Comfort. Jelikož jsou v konstrukci zařízení přítomná dvě čerpadla, jsou také nezbytnou součástí řídicího systému frekvenční měniče, které daná čerpadla ovládají. Fyzické propojení komponent s jednotlivými PLC je vyobrazeno na následujícím diagramu:



Obrázek 2.1: Diagram propojení komponent řídicího systému s komponentami konstrukce zařízení

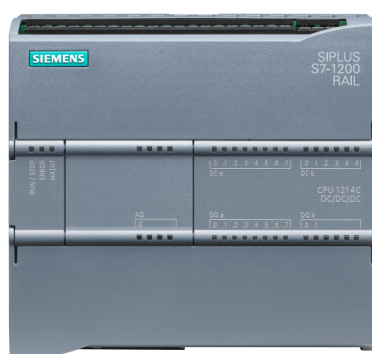
Lze vidět, že všechna měření v nádrži *Tank 1* (vodivost, pH) a měření tlaku v potrubí za čerpadlem *Pump M2*, jsou přivedena do PLC S7-1200. Z tohoto PLC je také digitálním signálem ovládáno míchadlo *Stirrer M1* a digitálním signálem jsou ovládána dávkovací čerpadla *Dosing pump M4* a *Dosing pump M5*. Z jmenovaného PLC je také možné ovládat po sériové lince měnič čerpadla *Pump M2* pomocí protokolu USS/MB. K druhému PLC S7-1500 jsou připojena měření v nádrži *Tank 2* (vodivost, pH) a měření průtoku za čerpadlem *Pump M3*. Toto čerpadlo je možné ovládat z PLC S7-1200 i S7-1500, jelikož je jeho měnič připojen k síti používající průmyslový Ethernet protokol PROFINET. V této práci je však ovládáno z S7-1500. Při popisu se vyházelo z výkresu [21].

2.1 Programovatelné automaty

Programovatelný logický automat neboli PLC slouží k aktivování výstupních signálů dle naprogramovaného algoritmu rozhodujícího na základě čtených signálů ze vstupů. Výhodou programovatelných automatů je jejich flexibilita, tedy možnost naprogramovat PLC pro konkrétní průmyslovou úlohu a dále jejich robustnost při použití v průmyslovém prostředí. Nezanedbatelnou výhodou je modulární koncepce, tedy možnost přidání k základnímu CPU další rozšiřující moduly podle potřeby (například moduly přídavných I/O či komunikační moduly).

2.1.1 Siemens S7-1200

První programovatelný automat, který bude v této podkapitole popsán je PLC řady SIMATIC S7-1200. Rozměrově se jedná o kompaktnější a levnější verzi než následující popisovaná řada S7-1500. Zařazuje se pro použití v menších až středních aplikacích. Základní CPU může být rozšířeno o signálové, komunikační a technologické moduly. Samotné CPU je na následujícím obrázku:



Obrázek 2.2: SIMATIC S7-1200 CPU 1214C [16]

Základní CPU má označení DC/DC/DC. První DC znamená, že PLC je napájeno 24 V stejnosměrným napětím. K napájení může být použito také střídavé napětí

s hodnotou 120/230 V (označení AC). Označení DC na druhé pozici určuje typ napětí na vstupech PLC, tedy stejnosměrné 24 V napětí. Poslední pozice určuje typ výstupu, který může být buď MOSFET tranzistorový se stejnosměrným výstupním proudem, anebo reléový stejnosměrný či střídavý. Tento konkrétní typ PLC má 14 digitálních 24 V stejnosměrných vstupů a 10 stejnosměrných digitálních výstupů. Na výukovém zařízení je PLC rozšířeno o 2 komunikační moduly (USS/MB, PRO-FIBUS) a o 3 signálové moduly (přídavné digitální a analogové vstupy i výstupy). Pro program a data je určeno 100 KB paměti.

2.1.2 Siemens S7-1500

Druhé přítomné PLC v zařízení je řady SIMATIC S7-1500. Jedná o PLC pro použití ve středních až vyšších aplikacích. Stejně jako S7-1200 může být i toto PLC rozšířeno o další moduly. Toto CPU může být využito pro středně rychlé procesy. PLC je zobrazeno na dalším obrázku:



Obrázek 2.3: SIMATIC S7-1500 CPU 1512C [17]

První velký rozdíl oproti S7-1200 je, že S7-1500 má v sobě integrován malý displej, na kterém je například možné zjistit aktuální stav PLC. Znatelný rozdíl je i v počtu vstupů i výstupů. K dispozici je 32 digitálních vstupů a 32 digitálních výstupů. Toto PLC má již v základu i 5 analogových vstupů a 2 analogové výstupy. Pracovní paměť má vyhrazena 250 KB pro program a 1 MB pro data.

2.2 Frekvenční měnič

Frekvenční měnič slouží pro ovládání otáček střídavého motoru. Zajišťuje možnost nastavit různé hodnoty otáček, také umožňuje plynulý rozběh či zastavení motoru bez mechanických a proudových nárazů v aplikacích se setrvačností. Tyto vlastnosti lze využít pro optimalizaci a zkvalitnění výroby. V neposlední řadě snižuje použití frekvenčního měniče energetickou spotřebu, prodlužuje životnost a snižuje údržbu zařízení.

Ke změně otáček motoru dochází pomocí změny frekvence přiváděného napětí

na motor. Frekvence otáček asynchronního motoru je úměrná frekvenci napájecího napětí. Pro řízení otáček motoru tedy čerpadla jsou k dispozici na tomto zařízení tři frekvenční měniče SINAMICS G120C od firmy Siemens. Každý instalovaný měnič má jiný způsob komunikace (PROFINET, PROFIBUS, USS/MB) s PLC. Jak měnič vypadá je vidět na obrázku 2.4.



Obrázek 2.4: Sinamics G120C [18]

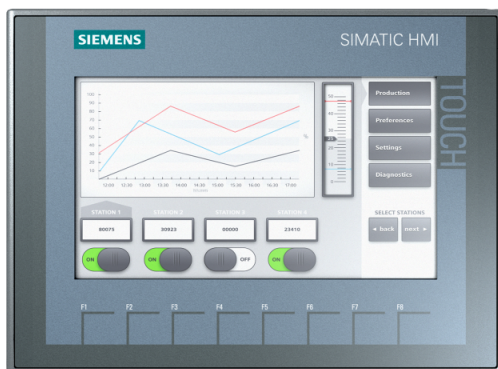
Již bylo zmíněno, že měniče bylo nutné před používáním nastavit. To se provádí dle parametrů motoru, pro který je měnič určen. Měniče na tomto zařízení byly nastaveny pro čerpadla Grundfos, jehož parametry jsou uvedeny v podkapitole 1.2. U frekvenčního měniče bylo možné zvolit i druh řízení a to U/f nebo vektorové řízení. Výběr závisí na vlastnostech aplikace jako je např. dynamika či přesnost počtu otáček. V této práci bylo zvoleno řízení U/f, což znamená, že měnič řídí napětí a frekvenci motoru v konstantním poměru, tj. například při snížení frekvence na poloviční hodnotou se sníží i hodnota napětí na polovinu. To způsobuje konstantní kroutící moment ve velkém rozsahu a otáčky motoru se mění v závislosti na jeho zatížení [10].

2.3 Operátorské panely

Jako rozhraní člověk-stroj (HMI) slouží na zařízení dva operátorské panely firmy Siemens. Oba panely disponují barevnými dotykovými displeji. Operátorské panely slouží pro ovládání stroje, nastavování jeho parametrů a sledování dat. Umožňují vykreslit grafy, dělat záznamy a vytvářet přehled o aktuálních a potvrzených chybách. Operátorské panely, které jsou k dispozici na výukovém zařízení jsou popsány v následujících oddílech.

2.3.1 KTP700 Basic

První popisovaný operátorský panel je opět od firmy Siemens s označením SIMATIC HMI KTP 700 Basic. Jedná se o sedmipalcový dotykový panel s barevným displejem. Navíc má tento panel pod displejem k dispozici osm funkčních tlačítek, které umožňují rychlou obsluhu panelu. Tento panel lze vidět na obrázku 2.5.



Obrázek 2.5: SIMATIC HMI KTP700 Basic [14]

Barevnost toho panelu činí 64K. Rozložení obrazu je 800 x 480 px. K dispozici je 10 MB paměti pro samotné prostředí a k vytváření data logů.

2.3.2 TP700 Comfort

Druhý panel SIMATIC HMI TP700 Comfort patří k pokročilejší řadě než předchozí jmenovaný panel a může být použit pro komplexnější aplikace. Operátorský panel je zobrazen na obrázku 2.6.



Obrázek 2.6: SIMATIC HMI TP700 Comfort [15]

Oproti KTP700B postrádá tento panel funkční tlačítka pod displejem. Nabízí však 16 miliónů barev a paměť větší o 2 MB. Tento panel navíc běží na operačním systému Windows CE 6.0.

Kapitola 3

Vývojový software

Tato kapitola se zabývá prostředím, které bylo použito pro psaní řídicího softwaru a také implementovanými funkčními bloky, které tento software, a tedy i PLC nabízí. Oba dva procesory tedy PLC S7-1200, S7-1500 a oba dva operátorské panely KTP700B, TPC700C je možné programovat z jednoho vytvořeného programu v prostředí TIA Portal od firmy Siemens. Toto prostředí umožňuje jednoduše vytvořit síť, nastavit IP adresu všech komponent a poté je podle potřeby propojit.

Vývojový software také umožňuje využít již vytvořené funkční bloky. V této práci bylo těchto bloku využito hned několik. První bloky, které byly využity, byly bloky pro posílání dat mezi přítomnými PLC. Pro tento účel bylo využito bloků PUT a GET. Jako další bylo nutné komunikovat s frekvenčními měniči pro ovládání jednotlivých čerpadel. K tomu slouží funkční bloky SinaSpeed (pro komunikaci s měničem po Profinetu) a USS_DRV (měnič komunikující po USS). A nakonec bylo využito bloku pro regulaci PID_Compact, jenž byl poté mírně modifikován a použit ve funkčním bloku, který byl pojmenován PID_Control.

V této a následující kapitole budou uváděny datové typy v anglickém znění a jejich popis i rozsah je uveden v tabulce 3.1.

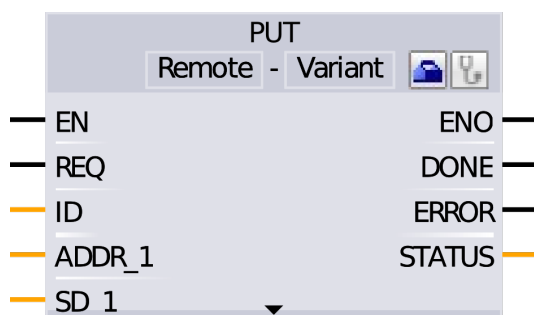
Typ	Popis	Velikost (bit)	Rozsah
BOOL(x)	boolean	1	true nebo false
BYTE(b)	hexadecimální číslo / BCD kód	8	-128 až +127
WORD(w)	hexadecimální číslo / BCD kód	16	-32 768 až +32 767
DWORD(dw)	hexadecimální číslo / BCD kód	32	-2 147 483 647 až +2 147 483 647
INT(i)	celé desítkové číslo se znaménkem	16	-32 768 až 32 767
DINT(di)	celé desítkové číslo se znaménkem	32	-2 147 483 648 až +2 147 483 647
REAL(r)	číslo s plovoucí desetinnou čárkou	32	-3.402823e+38 až +3.402823e+38

Tabulka 3.1: Tabulka datových typů [11]

Všechny informace této kapitoly jsou čerpány z manuálu o programu TIA Portal [11].

3.1 Funkční bloky PUT a GET

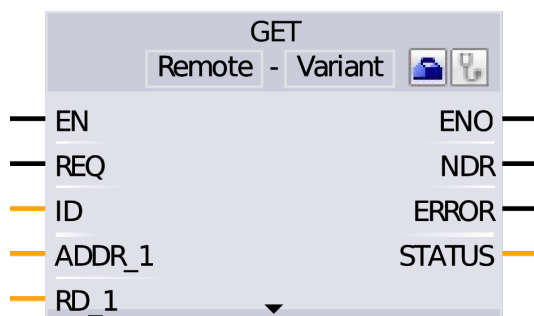
Funkční bloky PUT a GET byly využity pro oboustranné posílání dat mezi PLC S7-1200 a S7-1500. V každém PLC musely být vytvořeny instance obou funkčních bloků pro správnou funkčnost odesílání a přijímání dat. Nejdříve bude popsán blok PUT znázorněný na obrázku 3.1.



Obrázek 3.1: Funkční blok PUT

Funkce PUT slouží k zápisu dat na druhé S7 PLC. Proto je nutné v konfiguraci tohoto bloku zvolit IP adresu PLC příjemce. Konfigurační okno lze otevřít pomocí ikony modrého nosiče náradí. Na vstup REQ, který aktivuje zápis dat, byl přiveden periodický binární signál o frekvenci 2.5 Hz. Vstup ID slouží k specifikaci připojení k partnerskému PLC. Dále jsou uvedeny vstupy ADDR_1 a SD_1. K výměně dat je nutné specifikovat, která data z daného PLC budou zapisována a na jaké místo druhého PLC budou data poslána. K tomu slouží právě jmenované ukazatele. Ukazatel ADDR_1 slouží k specifikaci oblasti, do které budou data v druhém PLC zapsána. Ukazatel SD_11 slouží k specifikaci lokální oblasti, ze které budou data posílána na druhé PLC. Zmíněné oblasti jsou datové bloky, které jsou popsány v podkapitole 5.4. Výstup DONE slouží k indikaci stavu předávání. Pokud je hodnota tohoto výstupu *true*, data byla úspěšně předána do druhého PLC. Došlo-li k chybě při předávání, aktivuje se výstup ERROR. Výstup STATUS je proměnná typu WORD a lze podle jeho hodnoty dohledat, k jaké chybě popřípadě došlo.

Další funkční blok GET slouží ke čtení dat z druhého S7 PLC. Tento blok je vykreslen na následujícím obrázku 3.2.

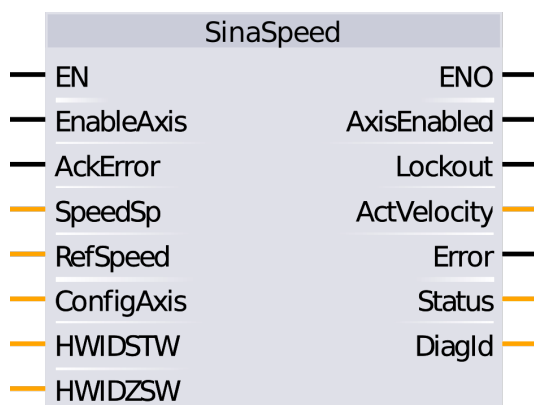


Obrázek 3.2: Funkční blok GET

Tento blok je nutné nastavit stejně jako funkční blok PUT. Vstupy REQ a ID mají stejnou funkci jako u předchozího bloku. Vstup ADDR_1 je v tomto případě ukazatel na místo (datový blok) v druhém PLC, ze kterého se bude číst. Vstup RD_1 je ukazatel na místo lokálního PLC, do kterého se zapisují čtená data. Výstup NDR má stejnou funkci jako výstup DONE u předchozího bloku. Pokud je jeho hodnota *true*, značí to úspěšné přečtení dat. Dále výstupy ERROR a STATUS jsou stejné jako u bloku PUT.

3.2 Funkční blok SinaSpeed

Dalším uvedeným blokem, který je již implementován ve vývojovém prostředí TIA Portal je blok pro řízení frekvenčního měniče (viz podkapitola 2.2), a tedy i motoru přes PROFINET s názvem *SinaSpeed*. Instance toho bloku byla vytvořena v PLC S7-1500, ze kterého je řízen motor *Pump M3*. Tento blok má následující rozložení vstupů a výstupů:



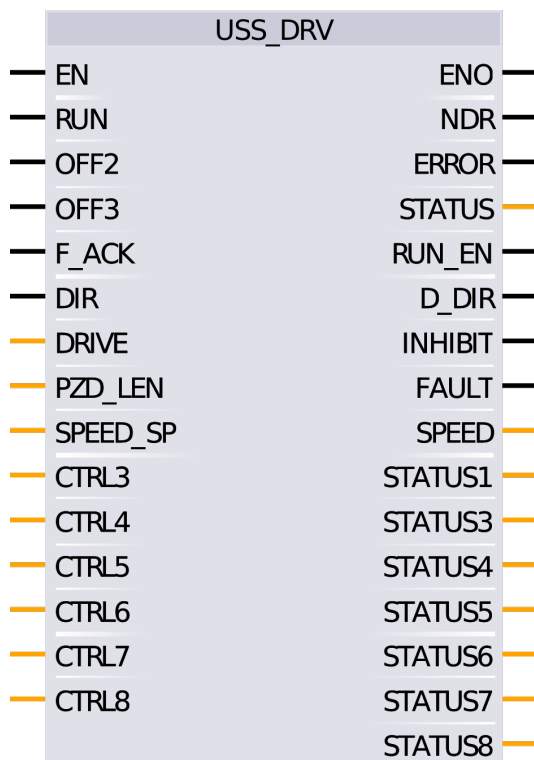
Obrázek 3.3: Funkční blok SinaSpeed

Ke spuštění motoru slouží vstup datového typu BOOL *EnableAxis*, který musí být nastavený na hodnotu *true*. Pokud nastane na motoru nějaká chyba, je nutné ji potvrdit vstupem *AckError*, aby bylo možné motor znovu spustit, a to za předpokladu že chyba byla odstraněna. Vstup s názvem *SpeedSp* datového typu REAL

slouží k nastavení požadovaných otáček motoru (setpoint). Pro nastavení jmenovitých otáček motoru slouží vstup `RefSpeed` datového typu `REAL`. Vstup `ConfigAxis` datového typu `WORD` slouží k nastavení dalších parametrů motoru jako je například otočení směru otáček. Vstupy `HWIDSTW` a `HWIDZSW` jsou komunikační parametry, na které musí být přiveden symbolický název, či ID hardware slotu motoru. Jedná se o tzv. telegram, který obsahuje všechny parametry a aktuální hodnoty motoru a musí být identický s nastaveným telegramem ve frekvenčním měniči. Výstup `AxisEnabled` typu `BOOL` je určen k indikaci běhu motoru. Dále `Lockout` je aktivní, pokud je aktivování potlačení zapnutí motoru v jeho parametrech. Výstup `ActVelocity` typu `REAL` obsahuje aktuální otáčky motoru. Pokud je aktivní chyba motoru, je hodnota výstupu `Error true`. Výstupní proměnná `Status` datové typu `INTEGER` má hodnotu podle aktuálního stavu motoru. Nakonec výstupní proměnná `DiagID` typu `WORD` popisuje chybu komunikace.

3.3 Funkční blok USS_DRV

Další blok určený pro řízení měniče, a tedy i motoru, je funkční blok `USS_DRV`. Tento blok slouží pro výměnu dat s motorem přes USS protokol po sériové lince. Instance tohoto bloku byla vytvořena v PLC S7-1200, který je propojen s frekvenčním měničem čerpadla *Pump M2*. Blok je na obrázku 3.4.



Obrázek 3.4: Funkční blok `USS_DRV`

Pro spuštění motoru slouží proměnná datového typu `BOOL` na vstupu pojmenovaná `RUN`. Vstupy `OFF2` a `OFF3` slouží k nastavení způsobu zastavení motoru (elek-

trické zastavení, rychlý stop). Dále bit `F_ACK` je určen pro potvrzení chyby. Vstupní bit `DIR` určuje směr otáčení motoru. Vstupní parametry `DRIVE` (adresa měniče/motoru) a `PZD_LEN` (délka `WORD`) se musí shodovat s nastavenými parametry měniče. Vstup `SPEED_SP` typu `REAL` nastavuje v procentech velikost otáček motoru. Řídící slova `CTRL3` až `CTRL8` jsou uživatelsky nastavitelná. Výstup `NDR` indikuje příjem nových komunikačních dat motoru. `ERROR` je aktivní, pokud nastala chyba v komunikaci. Proměnná `STATUS` popisuje stav komunikačního cyklu pomocí datového typu `WORD`. Další výstupní proměnná `RUN_EN` je aktivní, pokud motor běží a `D_DIR` odpovídá směru otáčení motoru. Výstup `FAULT` je určen pro indikaci chyby řízení motoru. Dále proměnná na výstupu typu `REAL` pojmenovaná `SPEED` má hodnotu aktuální rychlosti motoru. Proměnné typu `WORD` `STATUS1` až `STATUS8` obsahují stavové bity řízení, kde `STATUS1` má již určené nastavení a další v pořadí jsou již uživatelsky nastavitelné.

3.4 Funkční blok `PID_Compact`

Jeden z dalších důležitých bloků, který byl k dispozici je funkční blok PID regulátoru `PID_Compact`. Tento blok byl lehce modifikován (viz podkapitola 4.4) a použit pro regulační úlohy popsané v kapitole 7. `PID_Compact` poskytuje PID regulátor, jenž má integrované ladící funkce koeficientů regulátoru s názvem Pre-tuning a Fine tuning. Tento regulátor funguje na základě následující rovnice:

$$y = K_P[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x)] \quad (3.1)$$

kde

y : akční zásah regulátoru (output),

w : požadovaná hodnota regulované veličiny (setpoint),

x : skutečná hodnota regulované veličiny (input),

K_P : proporcionální zesílení regulátoru,

T_I : integrační časová konstanta,

T_D : derivační časová konstanta,

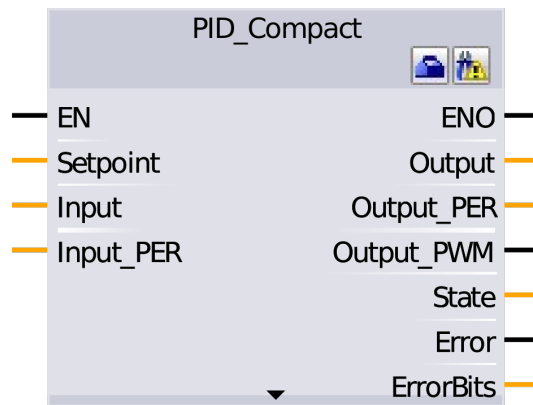
a : konstanta filtru derivační složky,

b : váha žádané veličiny u proporcionální složky,

c : váha žádané veličiny u derivační složky,

s : Laplaceův operátor.

Uvedená rovnice je integrována do funkčního bloku na obrázku 3.5.



Obrázek 3.5: Funkční blok PID_Compact

Tato složená verze funkčního bloku obsahuje vstup datového typu REAL s názvem **Setpoint**. Jedná se o požadovanou hodnotu regulované veličiny, která je v rovnici 3.1 označena písmenem w . Další proměnnou stejného typu na vstupu je proměnná **Input**. Na tento vstup je přivedena skutečná hodnota regulované veličiny, v rovnici 3.1 označena písmenem x . Vstup **Input_PER** je také požadovaná hodnota veličiny, ale jedná se o její analogovou hodnotu. Pomocí parametrů daného funkčního bloku je možné vnitřně přepínat mezi reálnou a analogovou hodnotou skutečné regulované veličiny. Pokud se jedná o analogovou hodnotu, je možné vnitřně nastavit její škálování.

Na pravé straně funkčního bloku PID_Compact se nachází výstupní proměnná typu REAL s názvem **Output**. Jedná se o výstup regulátoru, tedy jeho akční zásah, jenž se pohybuje v rozmezí od -100.0 % do 100.0 % a v rovnici 3.1 je označen písmenem y . Integer výstup **Output_PER** má stejný význam jako výstup **Output**, jedná se ale o jeho analogovou hodnotu. Pro regulaci je možné využít i PWM výstup označený jako **Output_PWM** (hodnota false nebo true). V jakém stavu se nalézá regulátor indikuje výstup **State** typu INTEGER, jenž se vždy nachází v jednom následujících stavů:

- 0 = Neaktivní,
- 1 = Pretuning,
- 2 = Finetuning,
- 3 = Automatický režim,
- 4 = Manuální režim.

Pokud se regulátor nachází v chybě, aktivuje se výstup **Error**. Důvod chyby je pak vyjádřen hodnotou typu DWORD na výstupu **ErrorBits**. S tímto blokem je svázáno velké množství nastavitelných parametrů obsažených v datové struktuře. Tato struktura nebude z důvodu velké obsáhlosti graficky uvedena. Struktura obsahuje důležité koeficienty regulátoru uvedené v rovnici 3.1, včetně jejich zálohy. Dále jsou ve struktuře obsažena omezení jako je omezení požadované a skutečné regulované veličiny a také omezení výstupu. To je také důvod proč se jedná o regulátor s anti-windupem,

aby bylo zabráněno unášení integrační složky. Dalšími obsaženými parametry jsou nastavení ladících algoritmů Pre-tuningu a Fine tuningu, jejichž funkce je popsána v následujících oddílech.

3.4.1 Pre-tuning

Procedura před-ladění (Pre-tuning) slouží ke stanovení parametrů z maximálního nárůstu skutečné regulované veličiny. Spustit ji lze pouze za následující podmínky:

$$\begin{aligned} |Setpoint - Input| > 0.3 \cdot |InputUpperLimit - InputLowerLimit|, \\ |Setpoint - Input| > 0.5 \cdot |Setpoint|, \end{aligned} \quad (3.2)$$

kde *Setpoint* je požadovaná regulovaná veličina, *Input* je skutečná regulovaná veličina, *InputUpperLimit* je maximální hodnota skutečné regulované veličiny a *InputLowerLimit* je její minimální hodnota. Po splnění této podmínky je výstup regulátoru (akční zásah) nastaven skokově na maximální povolenou hodnotu. Procedura je aktivní, dokud není dosaženo inflexního bodu skutečné regulované veličiny. Z jejího maximálního růstu jsou pak stanoveny koeficienty regulátoru uvedené v rovnici 3.1. Tento algoritmus je možné inicializovat i přes zašumění signálu, a to za podmínky, že hodnota růstu skutečné regulované veličiny bude zřetelně vyšší než její zašumění. Pro získání lepších parametrů je dobré provést dále Fine tuning popsáný v následujícím oddílu.

3.4.2 Fine tuning

Další algoritmus sloužící k ladění koeficientů regulátoru je tzv. doladění (Fine tuning). Procedura spočívá v generování konstantních, omezených oscilací skutečné regulované veličiny. Na základě frekvence a amplitudy skutečné regulované veličiny jsou vypočteny koeficienty regulátoru uvedené v rovnici 3.1. Tento algoritmus generuje regulátor s parametry, jenž zajišťují lepší regulaci i s většími rušeními než regulátor s parametry z Pre-tuningu.

Kapitola 4

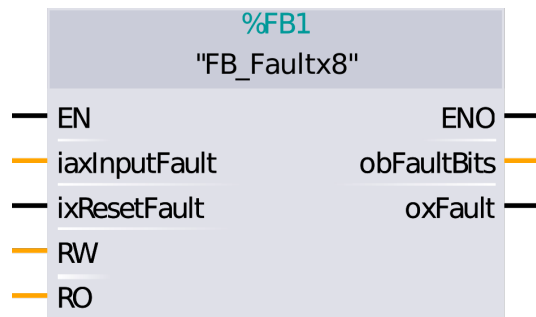
Funkční bloky pro manipulaci

Pro chyby, měření a pumpy bylo nutné navrhnout funkční bloky¹, které zajistí správnou manipulaci s jednotlivými komponentami. Jako první bylo důležité navrhnout blok pro manipulaci s chybami, jelikož je možné využít jeho funkci v dalších funkčních blocích. Dále bylo nutné navrhnout funkční blok pro zpracování měření a blok pro manipulaci s jednotlivými čerpadly. Každý funkční blok by měl být univerzální a jeho parametry by měly být nastavitelné tak, aby bylo možné použít jejich instance na jednotlivé chyby, měření a pumpy. Nastavitelnost parametrů jednotlivých funkčních bloků z operačního panelu zajišťují datové bloky. Motivací pro návrh těchto bloků je následovně snadnější naprogramování řídicího algoritmu pro regulační úlohy, jelikož se není nutné starat o obsluhu komponent, o které se nezávisle na řízení starají právě vytvořené funkční bloky. Pro účely snadnějšího rozpoznávání datového typu u proměnné byla zavedena konvence v pojmenování, kdy první písmeno názvu proměnné značí jestli se jedná o vstupní proměnnou (písmeno i) či výstupní proměnnou (písmeno o). V případě že se nejedná o výstupní ani vstupní proměnnou, není v názvu ani jedno z uvedených písmen a ihned následuje písmeno značící datový typ, například pro datový typ BOOL je na druhém místě písmeno x (viz tabulka 3.1 v kapitole 3).

4.1 Funkční blok pro chyby

Funkční blok pro manipulaci a zpracování chyb je jeden z nejkritičtějších. Správná funkce takového bloku zajišťuje bezpečnost zařízení a chrání jednotlivé komponenty před jejich poškozením. U chyby je nutné zajistit její aktivaci a udržování aktivního stavu i po negaci vyhodnocovacího signálu. K deaktivaci aktivní chyby by mělo dojít jiným signálem tzv. resetem. Aby nebylo nutné vytvářet velký počet instancí pro jednotlivé chyby, bylo vhodné navrhnout blok, který zpracuje několik chyb zároveň.

¹Návrh vychází z funkčních bloků použitých ve firmě ProMinent Systems spol. s r.o.



Obrázek 4.1: Funkční blok pro manipulaci s chybami FB_Faultx8

Výsledkem návrhu je funkční blok (obrázek 4.1), který zpracovává 8 chyb současně. Vstup s názvem `iaxInputFault` je pole datového typu `BOOL` o velikosti 8. Na tento vstup jsou přivedeny vyhodnocovací signály jednotlivých chyb. Pokud je tedy jeden ze vstupů `true`, chyba je vyhodnocena jako aktivní a na výstupu bloku je aktivován bit chyby se stejným pořadím proměnné s názvem `obFaultBits`. K deaktivaci chyb pak slouží vstup `ixResetFault`. Tento vstup nastaví všechny aktivní chyby jako `false`.

Výstup `obFaultBits` je proměnná typu `BYTE`, jenž představuje aktivní chyby. Chyba je aktivní i poté, co vyhodnocovací signál ze vstupu `iaxInputFault` spadne z `true` do `false`. Důvod proč tato proměnná není také pole je takový, že datový typ `BYTE` je možné lehce přiřadit k proměnné datového typu `WORD`. Tento datový typ pak slouží k indikaci chyb na operátorském panelu. Dále je zde výstup typu `BOOL` s názvem `oxFault`, který je aktivní, pokud je aktivní alespoň jedna chyba. Při větším počtu instancí tohoto funkčního bloku je vhodné přivést tento výstup do indikace tzv. obecné chyby, která pak může být použita ke změně stavu zařízení, např. k jeho zastavení.

4.1.1 Datový blok pro chyby

Proměnné `RW` a `RO` jsou vstupně/výstupní (obrázek 4.1). Na tyto periferie jsou přivedeny data z data bloku `DB_GlobalPLC` (viz podkapitola 5.1). Tato data slouží ke konfiguraci chyb a jsou nastavitelná z operátorského panelu.

■	▼	Faults[1]	Struct
■	▼	RO	"UDT_FaultRO"
■		bFaultInput	Byte
■		bFaultBits	Byte
■	▶	aiFaultCounter	Array[0..7] of Int
■	▶	asTag	Array[0..7] of Strin...
■	▼	RW	"UDT_FaultRW"
■	▶	axFaultDeactivate	Array[0..7] of Bool
■	▶	axFaultInverse	Array[0..7] of Bool
■	▶	axResetCounter	Array[0..7] of Bool
■	▶	aiFaultDelayTime	Array[0..7] of Int

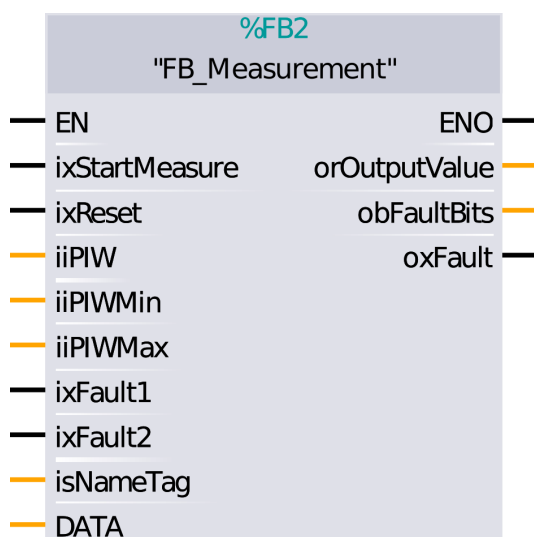
Obrázek 4.2: Data bloku FB_Faultx8

Struktura RO značí tzv. „ReadOnly“ data, tedy data určená pouze pro čtení z operátorského panelu. Na jednotlivé bity byte proměnné s názvem `bFaultInput` jsou přivedeny vstupní *bity* pole `iaxInputFault`. Proměnná `bFaultBits` typu BYTE obsahuje již vyhodnocené chyby (jedná se o stejné hodnoty jako na výstupu funkčního bloku `obFaultBits`). Pole osmi proměnných typu INTEGER pojmenované `aiFaultCounter` obsahuje počet aktivací jednotlivých chyb a pole `asTag` osmi proměnných typů STRING umožňuje pojmenovat jednotlivé chyby.

Dále je zde struktura RW obsahující tzv. „ReadWrite“ data. Tyto data jsou nastavitelná z operátorského panelu a umožňují konfigurovat vyhodnocení chyb, např. zpoždění aktivace chyby, inverze chyby, atd. Pole `axFaultDeactivate` umožní deaktivovat vyhodnocení chyby. Jeli tedy například pátý bit tohoto pole aktivní, tak po příchodu chyby na pátý bit vstupu `iaxInputFault` (`bFaultInput`), zůstane pátý bit bytu `bFaultBits` ve stavu *false*, chyba je tedy neaktivní. Pole `axFaultInverse` umožňuje inverzi chyby. Jeli tedy na vstupu `iaxInputFault` stav *false*, je chyba aktivována. Proměnná `axResetCounter` slouží k resetu čítače chyb `aiFaultCounter` a `aiFaultDelayTime` umožňuje nastavit zpoždění aktivace chyby.

4.2 Funkční blok pro měření

Ve funkčním bloku určeném pro měření (obrázek 4.3) probíhá důležité normování a poté škálování analogového signálu. Dále v bloku probíhá kontrola mezních hodnot měřeného signálu a případné aktivování chyby či varování při jejich překročení. Pro každé měření je vytvořena instance tohoto bloku.



Obrázek 4.3: Funkční blok pro manipulaci s měřením FB_Measurement

Vstup `ixStartMeasure` datového typu `BOOL` aktivuje dané měření, pokud je tento vstup ve stavu `false`, měření neprobíhá a případná přichodí chyba není vyhodnocena a výstup `oxFault` bude tedy vždy `false`. Až při aktivním vstupu `ixStartMeasure` se může aktivovat chyba `oxFault`, například z důvodu překročení nastavených mezních hodnot měřeného signálu či při nastavení jednoho ze vstupů `ixFault1` nebo `ixFault2` do stavu `true`. Tyto chyby je poté opět možné resetovat vstupem `ixReset`, pouze pokud už nejsou aktivní. Na vstup `iiPIW` datového typu `INTEGER` je přiváděn surový signál přímo z analogového vstupu PLC či modulu. Vstupy `iiPIWMin` a `iiPIWMax` pak vymezují tento signál a obě PLC (S7-1200 a S7-1500) mají tento rozsah 0 až 27648. Parametry `ixFault1` a `ixFault2` jsou přidavné chyby. Mohou zde být přivedeny signály, které například kontrolují, zda je daný senzor připraven. Vstupní parametr `isNameTag` datového typu `STRING` umožňuje pojmenovat dané měření.

Výstup typu `REAL` `orOutputValue` obsahuje již normovaný a škálovaný signál. Parametry `obFaultBits` a `oxFault` jsou popsány v podkapitole 4.1.

4.2.1 Datový blok pro měření

Stejně jako u funkčního bloku pro chyby obsahuje tento blok data pouze pro čtení (RO) a data pro čtení i zápis (RW). Na rozdíl od jmenovaného bloku jsou tyto struktury spojené do jedné s názvem `DATA` (obrázek 4.3).

▼ Measurements[1]	"UDT_Measurement"
■ ▼ RO	Struct
■ rOutputValue	Real
■ xFault	Bool
■ iPIW	Int
■ iStatus	Int
■ sNameTag	String
■ ► FaultsRO	"UDT_FaultRO"
■ ▼ RW	Struct
■ rMeasuredValueMin	Real
■ rMeasuredValueMax	Real
■ rMeasuredOffset	Real
■ diTimeStartupDelay	DInt
■ ▼ Limits	Struct
■ xLimitsDeactivation	Bool
■ rWarningMin	Real
■ rWarningMax	Real
■ rFaultMin	Real
■ rFaultMax	Real
■ ► FaultsRW	"UDT_FaultRW"

Obrázek 4.4: Data bloku FB_Measurement

Data pouze pro čtení začínají parametrem `rOutputValue`. Tento parametr obsahuje stejnou hodnotu jako výstup funkčního bloku `orOutputValue`. Chyba `xFault` byla také popsána, stejně jako `compPIW`, což je již jmenovaná hodnota surového signálu (vstup `iiPIW` funkčního bloku). Dále je zde proměnná typu INTEGER s názvem `iStatus` značící stav daného měření. Měření je vždy v jednom z následujících pěti stavů:

- 0 = měření zastaveno (vstup `ixStartMeasure` je *false*),
- 1 = aktivní měření bez chyby,
- 2 = chyba (překročení mezních analogových hodnot, přídavné chyby),
- 3 = varování (překročení mezních hodnot `rWarningMin` či `rWarningMax`),
- 4 = chyba (překročení mezních hodnot `rFaultMin` či `rFaultMax`).

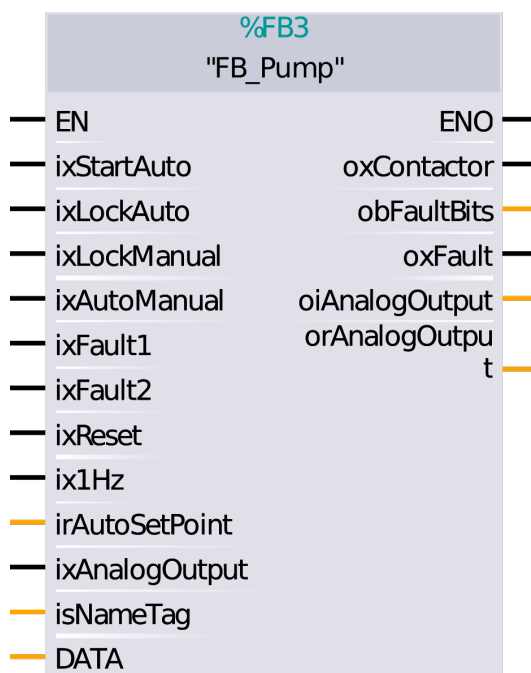
Z proměnné `sNameTag` typu STRING je možné vyčíst název měření pojmenovaný na vstupu `isNameTag`. Struktura `FaultsRO` je podrobněji přiblížena v oddílu 4.1.1.

Struktura pro čtení i zápis (RW) začíná proměnnými `rMeasuredValueMin` a `rMeasuredValueMax`, kterými se nastaví měřený rozsah dané fyzikální veličiny. Například pro průtokoměr je tento rozsah 0 až 10000 litrů za hodinu. Parametr `rMeasuredOffset` umožňuje korekci měření. Dále je zde proměnná `diTimeStartupDelay`, která nastaví zpoždění začátku měření. Struktura `Limits`

zahrnuje již zmíněné mezní hodnoty pro měření. Jeli tedy například požadavek na spuštění čerpadla při minimální a maximální hladině v nádrži, nastaví se `rFaultMin` na hodnotu 20 l a `rFaultMax` na hodnotu 150 l. Bude-li `rOutputValue` mimo tento rozsah, bude měření ve stavu 4 a výstup `oxFault` bude *true*. Tuto chybu je pak možné použít například pro odstavení čerpadla. Stejný postup je při nastavení varování. Tyto limity je možné deaktivovat parametrem `xLimitsDeactivation`. O struktuře `FaultsRW` se lze dočíst v oddílu 4.1.1.

4.3 Funkční blok pro čerpadlo

Funkční blok pro čerpadlo zajišťuje jeho spuštění a případné odstavení, pokud nastala chyba či rovnou blokaci spuštění, pokud nejsou splněné dané podmínky, jako je například nedostatečná hladina vody v nádrži, ze které je čerpáno. Čerpadlo je možné spustit v jednom z režimů, a to v režimu manuálním či automatickém. Manuální spuštění slouží k individuální aktivaci čerpadla a spuštění provádí operátor z operátorského panelu. Automatické spuštění je pak následkem splnění určité posloupnosti procesu celého zařízení. Avšak není vyloučeno, že by nebylo možné spustit čerpadlo manuálně i přesto, že celé zařízení je v automatickém režimu. To už je záležitostí individuálního nastavení a zvážení toho jaký vliv bude mít takové spuštění na celý proces řízení.



Obrázek 4.5: Funkční blok pro manipulaci s čerpadlem FB_Pump

Požadavek na spuštění čerpadla zajišťuje vstup `ixStartAuto`. Je to požadavek v rámci automatického spuštění, jelikož manuální spuštění je zajištěné pomocí proměnné struktury na vstupu `DATA` (viz následující oddíl 4.3.1), která stejně jako v předchozích funkčních blocích slouží ke konfiguraci a k monitorování daného bloku

z operátorského panelu. Pokud spuštění není nijak zablokováno, je na výstupu bloku aktivován parametr `oxContactor`. Tento výstup je pak možné přivést na příslušný výstup PLC či vstup daného bloku pro ovládání čerpadla po dané komunikační sběrnici (kapitola 3).

Již bylo řečeno, že spuštění čerpadla je možné zablokovat. K tomu slouží vstupy `ixLockAuto` a `ixLockManual`. Pokud je `ixLockAuto` aktivní a na vstup `ixStartAuto` přijde požadavek na spuštění, čerpadlo není spuštěno a výstup `oxContactor` zůstává ve *false*. Identicky funguje blokáce manuálního spouštění pomocí vstupního parametru `ixLockManual`. Vstup `ixAutoManual` slouží k přepínání mezi automatickým a manuálním spouštěním a určuje tedy, který parametr čerpadlo aktivuje. Je nutno podotknout, že tyto vstupy nejsou vyhodnoceny jako chybový stav, a proto zůstává `oxFault` deaktivovaný i přesto, že je jeden z jmenovaných vstupů aktivní. Pro aktivaci chyby slouží přídatné chyby ve formě vstupu `ixFault1` a `ixFault2`. Po aktivování těchto chyb není možné spustit čerpadlo v žádném režimu. To je možné až po odstranění chyb a jejich resetování vstupem `ixReset`. Na vstup `ix1Hz` datového typu `BOOL` je nutné přivést vteřinový puls, ten slouží k počítání operačních hodin daného čerpadla (inkrementace uplynulých vteřin a následný přepočítání na hodiny).

Další z důležitých vlastností pumpy je zpracování požadované hodnoty otáček čerpadla. Požadovaná hodnota otáček se nastaví zadáním na vstupní proměnnou `irAutoSetPoint`. Rozsah zadané požadované hodnoty je 0.0 až 100.0 %, která je škálována na výstup `oiAnalogOutput` v rozsahu 0 až 27 648. Na výstupní proměnnou `orAnalogOutput` je přiveden bez jakýchkoliv úprav, vstup `irAutoSetPoint` nebo `rManualValue` (viz oddíl 4.3.1) podle režimu čerpadla. Povolení analogového řízení otáček se provádí vstupem `ixAnalogOutput`.

4.3.1 Datový blok pro čerpadlo

Datový blok pro čerpadlo je také pojmenován `DATA`, je rozdělen na strukturu s proměnnými určenými pouze pro čtení z operačního panelu (`RO`) a strukturu určenou pro čtení i zápis (`RW`). Přehled této data struktury je zobrazen na obrázku 4.6.

■ ▼ Pumps[1]	"UDT_Pump"
■ ▼ RO	Struct
■ xLockedAuto	Bool
■ xLockedManual	Bool
■ xAnalogOutput	Bool
■ xContactor	Bool
■ xCommonFault	Bool
■ diOperationHours	DInt
■ diPumpStarts	DInt
■ rAutoValue	Real
■ iStatus	Int
■ sNameTag	String
■ iAnalogOutput	Int
■ rAnalogOutput	Real
■ ► FaultsRO	"UDT_FaultRO"
■ ▼ RW	Struct
■ xManualOn	Bool
■ xResetOperationHours	Bool
■ xResetPumpStarts	Bool
■ iOnDelayTime	Int
■ iOffDelayTime	Int
■ rManualValue	Real
■ ► FaultsRW	"UDT_FaultRW"

Obrázek 4.6: Data bloku FB_Pump

Datová struktura obsahuje proměnné `xLockedAuto` a `xLockedManual` typu `BOOL`, které jsou přímou indikací vstupů funkčního bloku `ixLockAuto` a `ixLockManual`. Proměnná `xAnalogOutput` je opisem proměnné `ixAnalogOutput`. `xContactor` představuje výstup bloku `oxContactor`, indikaci zdali je či není čerpadlo spuštěné. Proměnná `diOperationHours` typu `DOUBLE INTEGER` obsahuje počet operačních hodin čerpadla, tedy počet hodin, kdy je čerpadlo spuštěné. Stejně tak `diPumpStarts` čítá počet spuštění čerpadla. Z operačního panelu je potřeba vyčíst, v jakém stavu se nachází dané čerpadlo. Obdobně jako v případě bloku určeného pro měření se i zde nachází k tomu určená proměnná `INTEGER` s názvem `iStatus`. Čerpadlo se vždy nachází v jednom z následujících stavů:

- 0 = čerpadlo je vypnuté v automatickém režimu,
- 1 = čerpadlo je spuštěné v automatickém režimu,
- 2 = chyba (přídavné chyby),
- 3 = čerpadlo je vypnuté v manuálním režimu,
- 4 = čerpadlo je spuštěné v manuálním režimu.

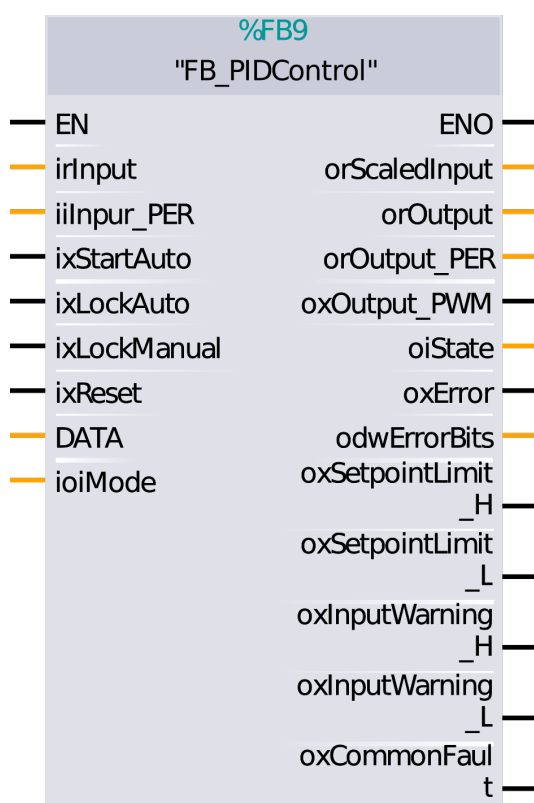
Název čerpadla pojmenované prostřednictvím vstupu `isNameTag` lze vyčíst z proměnné `sNameTag`. Zadaná požadovaná hodnota otáček čerpadla ze vstupu funkčního bloku je zapsána v proměnných `iAnalogOutput` i `rAnalogOutput`. Struktura

FaultsR0 je popsána v oddílu 4.1.1.

Podstruktura RW slouží ke čtení i k zápisu. Z této struktury lze spustit čerpadlo, pokud se nachází v manuálním režimu, a to pomocí `xManualOn`. Operační hodiny a počet spuštění čerpadla lze vynulovat pomocí proměnných `xResetOperationHours` a `xResetPumpStarts`. U čerpadla je také možné nastavit zpoždění jeho zapnutí i vypnutí v automatickém režimu. V návrhu řídicího softwaru je však nutné počítat s tím, že pokud se nastaví libovolný čas a řídicí software se odvíjí od spuštění čerpadla, může to mít vliv na celý běh zařízení. Pakliže je možné nastavit otáčky čerpadla, provede se to pomocí proměnné `rManualValue` typu REAL. O struktuře `FaultsRW` se lze dočíst v oddílu 4.1.1.

4.4 Funkční blok pro regulátor

V kapitole 3 byl uveden funkční blok regulátoru `PID_Compact`, který bylo nutné lehce modifikovat. Stejně jako u předchozích vlastních funkčních bloků, byla i u tohoto modifikovaného bloku vytvořena datová struktura která korespondovala s datovou strukturou jmenovaného funkčního bloku. I tato datová struktura byla přivedena na vstup s názvem `DATA` funkčního bloku na obrázku 4.7.



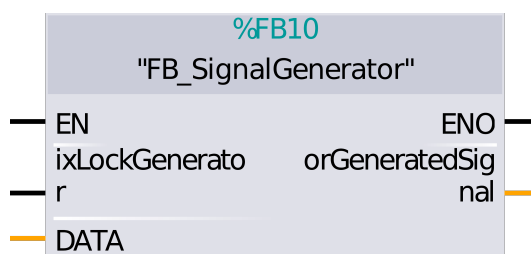
Obrázek 4.7: Funkční blok pro regulátor `FB_PIDControl`

Lze vidět, že tomuto funkčnímu bloku byl navíc přidán vstup `ixStartAuto`, který slouží pro spuštění regulátoru a dále vstupní proměnné `ixLockAuto` a `ixLockManual`, které jak již podle názvu vypovídá, blokují chod regulátoru. Dalším

důvodem proč byl `PID_Compact` modifikován na `FB_PIDControl` je ten, že režim regulátoru (tedy jeden z ladících režimů – automatický/manuální) bylo možné spustit nastavením proměnné datové struktury a navíc bylo nutné potvrzení zvoleného režimu další proměnnou. V tomto bloku se režim spustí ihned po změně k tomu určené proměnné bez nutnosti potvrzení další proměnnou. Dále pak byla přidána další struktura pro koeficienty, která je určená pro jejich manuální nastavení.

4.5 Funkční blok pro generátor signálu

V této práci bylo nutné navrhnout generátor signálu. Vygenerovaný signál s danými parametry byl pak přiveden na vstup požadované hodnoty otáček čerpadla (viz podkapitola 4.3), aby bylo možné vytvořit chybu v jedné z regulačních úloh (kapitola 7). Funkční blok je jednoduchý a je zobrazen na obrázku 4.8.



Obrázek 4.8: Funkční blok pro generování signálu `FB_SignalGenerator`

Generátor je možné zablokovat, a to vstupem `ixLockGenerator`. Generovaný signál je vyveden na výstup s názvem `orGeneratedSignal`. Dále stejně jako v předchozích případech, má tento blok vstupně-výstupní proměnnou `DATA`, kde je možné zvolit typ generovaného signálu a nastavit jeho parametry. Struktura je popsána v následujícím oddílu.

4.5.1 Datový blok pro generátor signálu

Rozdělení struktury je opět stejné jako u měření, čerpadla atd. Pomocí proměnných obsažených v této struktuře lze zvolit průběh generovaného signálu a stanovit jeho parametry. Obsah struktury je následující:

■ ▼ SignalGenerators[1]	*UDT_SignalGenera...
■ ▼ RO	Struct
■ rGeneratedSignal	Real
■ ▼ RW	Struct
■ xStartGenerating	Bool
■ iGeneratorType	Int
■ rMaxOutput	Real
■ rMinOutput	Real
■ ▼ StepParams	Struct
■ rAmplitude	Real
■ diStartDelay	DInt
■ ▼ SquareParams	Struct
■ rAmplitude	Real
■ diPeriod	DInt
■ diPulseWidth	DInt
■ ▼ SineParams	Struct
■ rAmplitude	Real
■ diPeriod	DInt
■ rOffset	Real
■ ▶ TriangleParams	Struct
■ ▶ SawtoothParams	Struct

Obrázek 4.9: Data bloku FB_SignalGenerator

Struktura pro čtení obsahuje jedinou proměnnou a to `rGeneratedSignal`, do které se zapisuje generovaný signál. Hodnota je stejná jako na výstupu bloku `orGeneratedSignal`.

Nastavitelné proměnné začínají proměnnou pro spuštění generování signálu `xStartGenerating`. Na výběr je pět průběhů signálu, a to skok hodnoty, obdélníkový průběh, sinusový průběh, trojúhelníkový průběh a pilovitý průběh. Zvolení z vyjmenovaných průběhů se provede nastavením hodnoty `iGeneratorType`, přičemž výběr začíná nulou tedy skokem. Každý z průběhů má svoji strukturu, která obsahuje parametry daného průběhu. Proměnné `rMaxOutput` a `rMinOutput` omezují generovaný signál, jehož hodnota se tedy pohybuje v rozmezí těchto parametrů nehlédě na nastavenou amplitudu daného průběhu.

U skoku signálu lze nastavit jeho amplitudu `rAmplitude` a čas skoku od startu generování `xStartGenerating`. U obdélníkového průběhu se také nastavuje jeho amplituda, dále jeho perioda `diPeriod` a šířka obdélníku `diPulseWidth`. U sinusového průběhu se také nastavuje amplituda, perioda a jeho posunutí po ose y (offset) `rOffset`. Stejně parametry jako pro sinusový průběh se nastavují pro trojúhelníkový i pilovitý průběh.

Kapitola 5

Data a datové rozhraní

V předchozí kapitole byly popsány funkční bloky. Každý z těchto funkčních bloků má vstupně-výstupní rozhraní na které se přivede datová struktura náležící dané instanci. V této kapitole bude popsána datová struktura, ve které jsou shromážděny všechny datové struktury, tj. data pro všechna měření, čerpadla, chyby, regulátory a generátory signálu. Také bude popsán způsob výměny dat přes rozhraní, které umožní přístup z operátorského panelu ke konkrétní struktuře dané komponenty zařízení (konkrétní datová struktura). Všechny tyto datové struktury jsou sdruženy do jednoho datového bloku. Práci s daty usnadňuje tzv. PLC datový typ neboli UDT (User-defined Data Type). UDT představuje datovou strukturu složenou z různých datových typů. V této práci bylo vytvořeno několik struktur tohoto typu určených pro chyby, měření, čerpadlo atd. Jak taková struktura může vypadat lze vidět obrázku 5.1.

UDT_Measurement						
Name	Data type	Default value	Accessible from HMI..	Writable from HMI..	Visible in HMI en..	
RO	Struct		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
rOutputValue	Real	0.0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
xFault	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
iPIW	Int	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
iStatus	Int	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
sNameTag	String	"	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
FaultsRO	*UDT_FaultRO*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
RW	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
rMeasuredValueMin	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
rMeasuredValueMax	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
rMeasuredOffset	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
diTimeStartupDelay	DInt	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Limits	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
FaultsRW	*UDT_FaultRW*		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
iIndex	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Obrázek 5.1: UDT pro měření

Lze vidět, že struktura je stejná jako na obrázku 4.4, což je právě instance UDT_Measurement na obrázku 5.1. Taková je výhoda UDT datového typu, jelikož není nutné kopírovat celé struktury. Jeli nutné použít stejnou strukturu, stačí například v datovém typu či na vstupu funkčního bloku inicializovat strukturu jednoduchým napsáním daného názvu, v tomto příkladě UDT_Measurement. U UDT

lze nastavit, zdali bude daný datový typ přístupný z HMI, zdali bude možné upravit daný typ z HMI nebo zdali bude viditelný z HMI.

5.1 Datový blok pro všechna data PLC

Ke shromáždění dat ze všech chyb, měření, čerpadel, regulátorů a generátorů signálu byl vytvořen datový blok s názvem `DB_GlobalPLC`. Každé PLC (S7-1200 i S7-1500) má jeden tento blok. Zde byly vytvořeny jednotlivé struktury pomocí již popsaného UDT. Datový blok vypadá následovně:

DB_GlobalPLC								
Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible...	Setpoint	
Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
General	*UDT_General*		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
RO	Struct		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
RW	Struct		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Faults	Array[1..2] of Struct		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Faults[1]	Struct		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Faults[2]	Struct		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Pumps	Array[1..2] of *UDT_Pump*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Pumps[1]	*UDT_Pump*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Pumps[2]	*UDT_Pump*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Measurements	Array[1..4] of *UDT_Measur...		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Measurements[1]	*UDT_Measurement*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Measurements[2]	*UDT_Measurement*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Measurements[3]	*UDT_Measurement*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Measurements[4]	*UDT_Measurement*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
PIDControls	Array[1..2] of *UDT_PID*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
PIDControls[1]	*UDT_PID*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
PIDControls[2]	*UDT_PID*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SignalGenerators	Array[1..1] of *UDT_Signal...		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SignalGenerators[1]	*UDT_SignalGenerator*		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Obrázek 5.2: Datový blok `DB_GlobalPLC`

Celý datový blok obsahuje jednu strukturu „všeobecných (General) dat“, která je popsána v dalším oddíle. Následují struktury pro chyby, čerpadla, měření, PID regulátory a generátor signálu. Tyto struktury byly přivedeny na jednotlivé instance funkčních bloků (viz kapitola 4), které byly inicializovány v k tomu určených funkčních blocích (viz podkapitola 5.3). Je nutné podotknout, že tento datový blok je přístupný pouze v PLC a jeho proměnné nejsou viditelné z HMI. Z tohoto důvodu bylo nutné vytvořit další datový blok, jehož struktury jsou viditelné z HMI. Tento datový blok je popsán v další podkapitole.

5.1.1 Datová struktura General

Všechny datové struktury uvedené v kapitole 4 jsou spojeny s instancí funkčního bloku. V datovém bloku `DB_GlobalPLC` je obsažena struktura `General`, která není

přivedena na žádný funkční blok a obsahuje přehled o nastavení a stavu celého zařízení. Její složení je následující:

UDT_General						
Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	
RO	Struct		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
xFaultsCommonFault	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
xMeasurementsCommon...	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
xPumpsCommonFault	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
xPIDControlsCommonFault	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
xCommonFault	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
xConnectionS71200Check	Bool	false	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
RW	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
xResetHMI	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
iASControl	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
xPLCMaster	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
iStationRegime	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
xRegulatedPump	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
xRegulatedTank	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Obrázek 5.3: Datová struktura řídicích veličin UDT_General

Opět je tato struktura rozdělena na struktury pouze pro čtení (RO) a pro čtení i zápis (RW). Struktura pro čtení obsahuje chyby z jednotlivých sekcí (chyby, měření, čerpadla, PID regulátory) a také chybu obecnou, která je aktivní, pokud je aktivní alespoň jedna chyba z některé sekce. Dále obsahuje periodický signál `xConnectionS71200Check`, jenž je odesílán z druhého PLC a slouží pro kontrolu stavu sítě mezi oběma PLC.

Struktura RW obsahuje řídicí veličiny. Jako první je proměnná `xResetHMI` typu `BOOL`, což je reset aktivovaný z panelu HMI či aktivovaný z druhého PLC (viz 5.4). Dále je zde proměnná `iASControl` typu `INTEGER`, která nastavuje režim celého zařízení dle následujících hodnot:

- 0 = OFF – zařízení je vypnuté,
- 1 = AUTO – zařízení je v automatickém režimu,
- 2 = MANUAL – zařízení je v manuálním režimu.

Další je proměnná pro nastavení PLC jako master (nadřazené), či slave (podřazené) PLC s názvem `xPLCMaster`. Pokud je její hodnota `false`, je nadřazené PLC S7-1500 a podřazené je PLC S7-1200. Hodnota `true` pak otočí nastavení. Proměnná `iStationRegime` typu `INTEGER` slouží pro selekci jedné z regulací (viz kapitola 7). Následující hodnoty určují regulační úlohu:

- 0 = žádná regulační úloha,
- 1 = regulace hladiny v nádrži,
- 2 = regulace průtoku,

3 = regulace tlaku.

Pokud je zvolena regulace hladiny, je možné nastavit, které z dvou přítomných čerpadel bude regulované (*false* = čerpadlo *Pump M3*, *true* = čerpadlo *Pump M2*) pomocí parametru *xRegulatedPump*. Možné je i pomocí proměnné *xRegulatedTank* zvolit v které nádrži bude regulována hladina (*false* = regulace hladiny v nádrži *Tank 1*, *true* = regulace hladiny v nádrži *Tank 2*).

5.2 Datový blok pro HMI

Pokud je v PLC obsaženo mnoho dat, může až přílišné množství čtených proměnných v HMI z PLC zpomalit běh operátorského panelu. Proto je vhodné snížit v některých případech tento počet pomocí přístupu k datům přes rozhraní, v tomto případě přes datový blok *DB_HMI*:

DB_HMI							
Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	
Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
General	*UDT_HMIGeneral*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Faults	Struct		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Pump	*UDT_Pump*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Measurement	*UDT_Measurement*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
RO	Struct		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
RW	Struct		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
iIndex	Int	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
SignalGenerator	*UDT_SignalGenerator*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
PIDControl	*UDT_PID*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

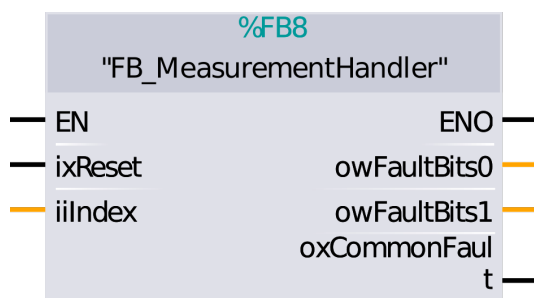
Obrázek 5.4: Datový blok *DB_HMI*

Tento blok obsahuje stejné struktury jako datový blok *DB_GlobalPLC*, avšak pouze jednu pro každé chyby, čerpadlo, měření, regulátor a generátor signálu. Další rozdíl oproti *DB_GlobalPLC* je ten, že všechny proměnné těchto struktur jsou viditelné z HMI. Také je u částí struktur *RO* a *RW* nastaveno, zda jsou zapisovatelné, či ne. Při vytváření prostředí panelu jsou tedy tyto proměnné přístupné a pracuje se pouze s proměnnými z tohoto datového bloku. Každá struktura obsahuje index (*iIndex* na obrázku), jehož funkce je vysvětlena v následující podkapitole.

5.3 Funkční blok pro předávání dat mezi PLC a HMI

V předchozích dvou podkapitolách byly představeny datové bloky *DB_GlobalPLC* a *DB_HMI*. V podkapitole viz 5.1 o *DB_GlobalPLC* bylo popsáno sjednocení všech datových struktur do daného datového bloku. Každá taková struktura musí být

přivedena na vstupně-výstupní rozhraní instance určitého funkčního bloku (viz kapitola 4). Instance těchto bloků jsou vytvářeny ve funkčním bloku pojmenovaném `FB_...Handler`. Každá část (měření, chyby atd.) má svůj „handler“, tedy například pro měření se jedná o `FB_MeasurementHandler`:



Obrázek 5.5: Funkční blok `FB_MeasurementHandler`

V tomto bloku jsou vytvářeny instance funkčního bloku `FB_Measurement` (viz 4.2). Jelikož jsou na PLC S7-1500 přivedeny čtyři analogová měření, byly vytvořeny čtyři instance. To znamená, že i datový blok `DB_GlobalPLC` obsahuje čtyři struktury pro měření, a právě v tomto funkčním bloku jsou tyto datové struktury přivedeny na dané instance. Tento blok dále sdružuje chyby `oxFault` ze všech instancí do jedné obecné chyby na výstupu pojmenované `oxCommonFault`. Dále jsou zde výstupní proměnné `owFaultBits0` a `owFaultBits1` typu `WORD`, na které jsou přivedeny výstupní proměnné `obFaultBits` typu `BYTE` ze všech instancí. Proměnná datového typu `WORD` je použita z důvodu způsobu vyhodnocování chyb v HMI. Na vstup `ixReset` funkčního bloku je přiveden i reset, tj. proměnná `xResetHMI` z `DB_GlobalPLC`. Na integer vstup `iiIndex` je pak přiveden index, který byl zmíněn v předchozí podkapitole 5.2 o `DB_HMI`. Tento index je důležitý pro další funkci „handler“ funkčního bloku a tou je selekce konkrétní struktury z HMI.

Pro snadnější pochopení bude uveden příklad pro nastavení parametrů měření. Na operátorský panel bylo vytvořeno pop-up okno (viz kapitola 6) pro měření, ze kterého je možné nastavit všechny parametry dané struktury. Jelikož jsou ale struktury v datovém bloku pro měření čtyři, bylo by nutné vytvořit v HMI čtyři okna pro každou strukturu. Tuto situaci řeší již zmíněný index, přes který se volí struktura, pro kterou se budou nastavovat parametry. Okno pro nastavení měření je tedy pouze jedno a při otevírání okna se nastaví číslo indexu struktury, jejíž parametry se budou nastavovat. Jeli tedy například potřeba nastavit parametry pro měření hladiny v nádrži, což je třetí funkční blok, a tedy i třetí struktura měření v `DB_GlobalPLC`, je potřeba při otevírání okna pro danou strukturu měření nastavit index na hodnotu tři. Tento princip ušetří práci při vytváření prostředí v operátorském panelu, ale i sníží počet současně čtených proměnných z PLC. Stejný princip je použit i pro struktury chyb, čerpadel, regulátorů a generátorů signálu. Jak taková práce s indexem funguje je vyobrazeno na obrázku 5.6.

DB_HMI
Measurement

Hydrostatic pressure in Tank 2 ✖

Description		Settings	
Common fault:	Active	Measured value max:	550.000
Raw measured value:	0	Measured value min:	0.000
Output value:	0.000	Offset:	+0.000
Status:		Measuring start delay:	0
Limits Activated			
Warning max:	230.000		
Warning min:	40.000		
Fault max:	250.000		
Fault min:	20.000		

↓ Index zvolený v HMI
iIndex = 3

DB_GlobalPLC
Measurements[index]

Measurements	Array[1..4] of "UDT..."	
Measurements[1]	"UDT_Measurement"	
Measurements[2]	"UDT_Measurement"	
Measurements[3]	"UDT_Measurement"	
RO	Struct	
RW	Struct	
rMeasuredV...	Real	0.0
rMeasuredV...	Real	550.0
rMeasured...	Real	0.0
diTimeStart...	Dint	0
Limits	Struct	
xLimits...	Bool	false
rWarnin...	Real	40.0
rWarnin...	Real	230.0
rFaultMin	Real	20.0
rFaultMax	Real	250.0

Obrázek 5.6: Princip předávání dat

Na obrázku je vidět vytvořené pop-up okno pro nastavení parametrů měření. Je to nastavení měření hydrostatického tlaku v nádrži číslo dva, což odpovídá hladině kapaliny v nádrži dvě. Toto měření je přivedeno na funkční blok pro měření číslo tři a jedná se tedy o datovou strukturu měření číslo tři. Pokud je třeba otevřít okno s danou strukturou, je nutné nastavit index na hodnotu tři.

5.4 Předávání dat mezi dvěma PLC

Struktura sítě umožňuje přístup z každého operátorského panelu (KTP700B, TP700C) k datům každého PLC (S7-1200, S7-1500). To znamená, že například z operátorského panelu KTP700B je možné nastavit parametry na PLC S7-1200 i S7-1500 a není tedy nutné posílat celé struktury z jednoho PLC do druhého. Jsou však proměnné a parametry, které posílat nutné je, jako je reset chyb, nastavení režimu obou PLC atd. Důležité je například také posílat hodnoty regulovaných ve-

ličin, aby byla dodržena zpětná vazba regulační smyčky. V podkapitole 3.1el byly popsány funkční bloky pro posílání dat mezi dvěma PLC. V této podkapitole budou stanoveny podmínky pro posílání řídicích veličin z nadřazeného PLC do podřízeného PLC, a také budou popsána data, která jsou posílána.

V oddíle 5.1.1 je popsána struktura veličin **General**, které jsou podstatné pro chod celého zařízení. Jedná se o proměnné **iASControl**, **iStationRegime**, **xResetHMI**, **xPLCMaster**, **xRegulatedPump** a **xRegulatedTank**. Vyjmenované proměnné jsou i součástí datové struktury, která je odesílána na podřízené PLC. Příklad struktury odesílané z PLC S7-1500 do PLC S7-1200 je obrázku 5.7.

DB200_Send_To_S71200						
Name	Data type	Offset	Start value	Retain	Accessible f...	
Static				<input type="checkbox"/>	<input type="checkbox"/>	
iASControl	Int	0.0	0	<input type="checkbox"/>	<input type="checkbox"/>	
iStationRegime	Int	2.0	0	<input type="checkbox"/>	<input type="checkbox"/>	
rPumpM2AutoInputS...	Real	4.0	0.0	<input type="checkbox"/>	<input type="checkbox"/>	
rTank2Level	Real	8.0	0.0	<input type="checkbox"/>	<input type="checkbox"/>	
xResetHMI	Bool	12.0	false	<input type="checkbox"/>	<input type="checkbox"/>	
xCommonFault	Bool	12.1	false	<input type="checkbox"/>	<input type="checkbox"/>	
xPLCMaster	Bool	12.2	false	<input type="checkbox"/>	<input type="checkbox"/>	
xConnectionCheck	Bool	12.3	false	<input type="checkbox"/>	<input type="checkbox"/>	
xRegulatedPump	Bool	12.4	false	<input type="checkbox"/>	<input type="checkbox"/>	
xRegulatedTank	Bool	12.5	false	<input type="checkbox"/>	<input type="checkbox"/>	

Obrázek 5.7: Data odesílána z PLC S7-1500 do S7-1200

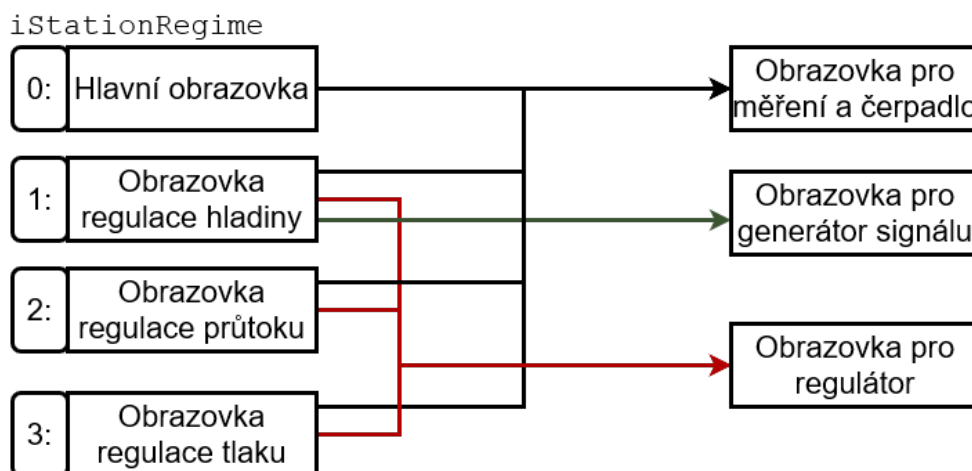
Struktura obsahuje již vyjmenované proměnné a také proměnné, jenž jsou nutné pro regulaci (**rTank2Level** a **rPumpM2AutoInputSetpoint**). Jelikož je možné nastavit jedno z přítomných PLC (S7-1200, S7-1500) jako nadřazené, a to druhé jako podřízené, bylo důležité navrhnout způsob zápisu řídicích proměnných do datového bloku **DB_GlobalPLC**. Jeli například PLC S7-1500 nastavené jako master PLC, druhé PLC S7-1200 je tedy automaticky slave PLC. V tomto případě jsou řídicí proměnné v datovém bloku **DB_GlobalPLC** v PLC S7-1500 nastavovány z operátorského panelu a v datovém bloku **DB_GlobalPLC** druhého PLC S7-1200 jsou tyto parametry zapisovány z datového bloku, který přijímá data právě z master PLC. Řídicí proměnné PLC S7-1200 jsou tedy čteny z S7-1500 a ne z operátorského panelu. Pokud se nastaví PLC S7-1200 jako master, jsou jeho řídicí parametry nastavovány z operátorského panelu a jsou odesílány na druhé slave PLC. Pokud je PLC master, nemůže mu být tento atribut odebrán. To znamená, že v případě kdy S7-1500 je master, může být proměnná **xPLCMaster** změněna pouze v tomto PLC a po změně, tedy po „odevzdání“ master atributu S7-1200, nelze tento atribut získat zpět z S7-1500, ale pouze pokud ho opět „odevzdá“ S7-1200.

Kapitola 6

Prostředí operátorského panelu

Tato kapitola se zabývá návrhem prostředí operátorského panelu. Jelikož jsou na zařízení k dispozici dva operátorské panely Siemens TP700 Comfort a Siemens KTP700 Basic (viz kapitola 2.3), musely být navržena dvě prostředí pro každý panel zvlášť, a to z důvodu rozdílného způsobu vytváření pop-up oken. Všeobecně by mělo být každé prostředí operátorského panelu přehledné, intuitivní a snadné k použití.

V této práci byly vytvořeny tři regulační úlohy, proto bylo nutné vytvořit strukturu přepínání oken při změně regulační úlohy. Již byl popsán důležitý parametr `iStationRegime` (viz oddíl 5.1.1), pomocí kterého lze zvolit právě požadovanou úlohu. Struktura obrazovek je uvedena na obrázku 6.1.



Obrázek 6.1: Struktura obrazovek

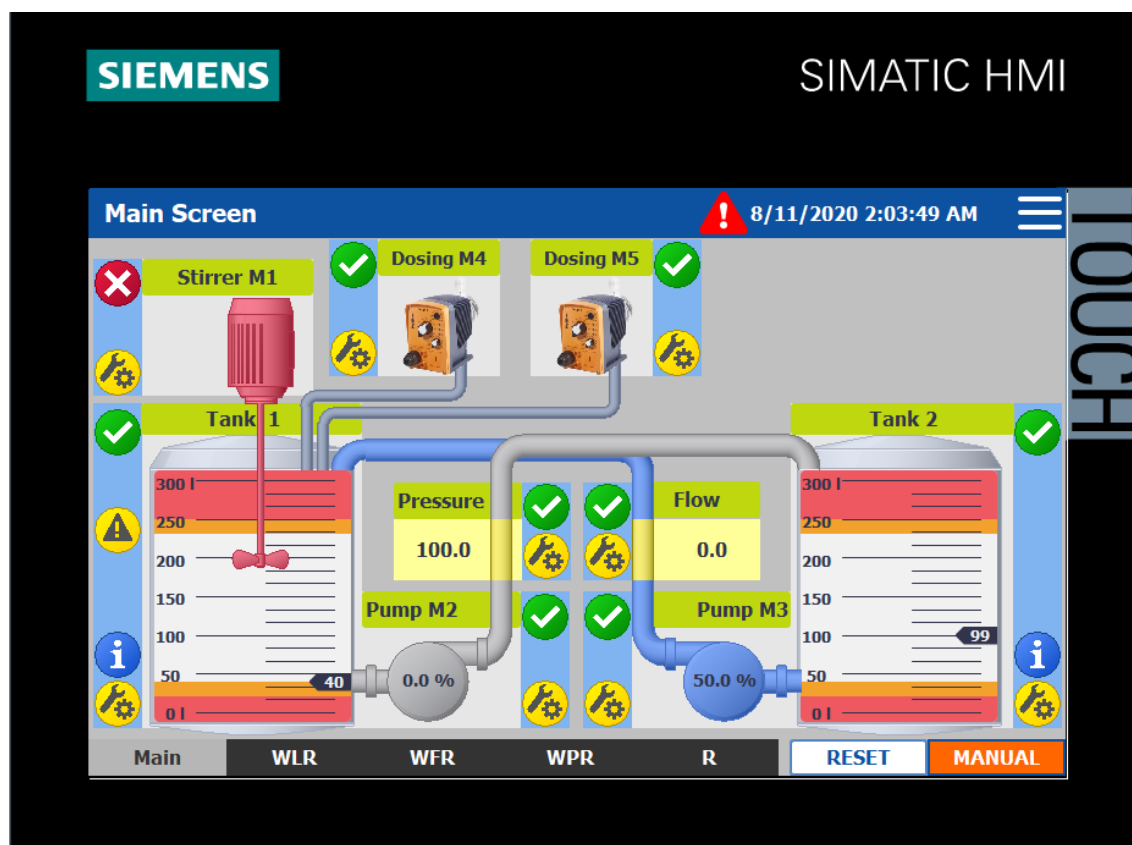
Struktura ukazuje, že priorita se přiřazuje obrazovce dle zvoleného režimu zařízení. Pokud tedy není zvolena žádná regulace je prioritní hlavní obrazovka. Avšak jeli zvolena jedna z regulací, nastaví se priorita na obrazovku dle zvolené regulace. Přepínání mezi hlavní obrazovkou a obrazovkami pro regulace je umožněno pouze pokud je zařízení zastavené tedy ve stavu OFF (`iASControl = 0`). Přístup k obrazovkám nastavení měření i čerpadla je možný v jakémkoliv režimu zařízení. Obrazovka pro nastavení regulátoru je přístupná pouze při zvolené regulaci. Nakonec obrazovka

pro generátor signálu, který slouží pro generování chyby, je přístupná pouze při regulaci hladiny, jelikož v ostatních regulacích není možné vygenerovat chybu přivedením signálu na druhé neregulované čerpadlo.

Jelikož byla navržena dvě prostředí pro každý operátorský panel zvlášť, která se sice liší graficky, ale nikoliv obsahem, bude vzhled operátorského prostředí ilustrován na panelu TP700C. Pro představu bude ukázán pouze vzhled hlavní obrazovky operátorského panelu KTP700B v následující podkapitole.







6.1 Hlavní obrazovka

Hlavní obrazovka v tomto zařízení slouží k přehledu všech měření, čerpadel a chyb. Z hlavní obrazovky je možné nastavit zařízení pouze do manuálního režimu ($iASControl = 2$), protože není zvolena žádná regulační úloha a automatický režim zařízení nemá v tomto případě smysl. Na této obrazovce jsou tedy vykreslena všechna přítomná čerpadla i měření. Přítomnost dvou operátorských panelů umožnila zvolit různý grafický vzhled obou panelů, avšak jejich obsah je identický. Rozvržení je tedy identické, leč vzhled komponent nikoliv. Na operátorský panel TP700C byla navržena hlavní obrazovka na obrázku 6.2.



Obrázek 6.2: Hlavní obrazovka na operátorském panelu TP700C

Před popisem samotné obrazovky je vhodné vysvětlit význam jednotlivých ikon, které se nachází v modrém pruhu u každé komponenty:

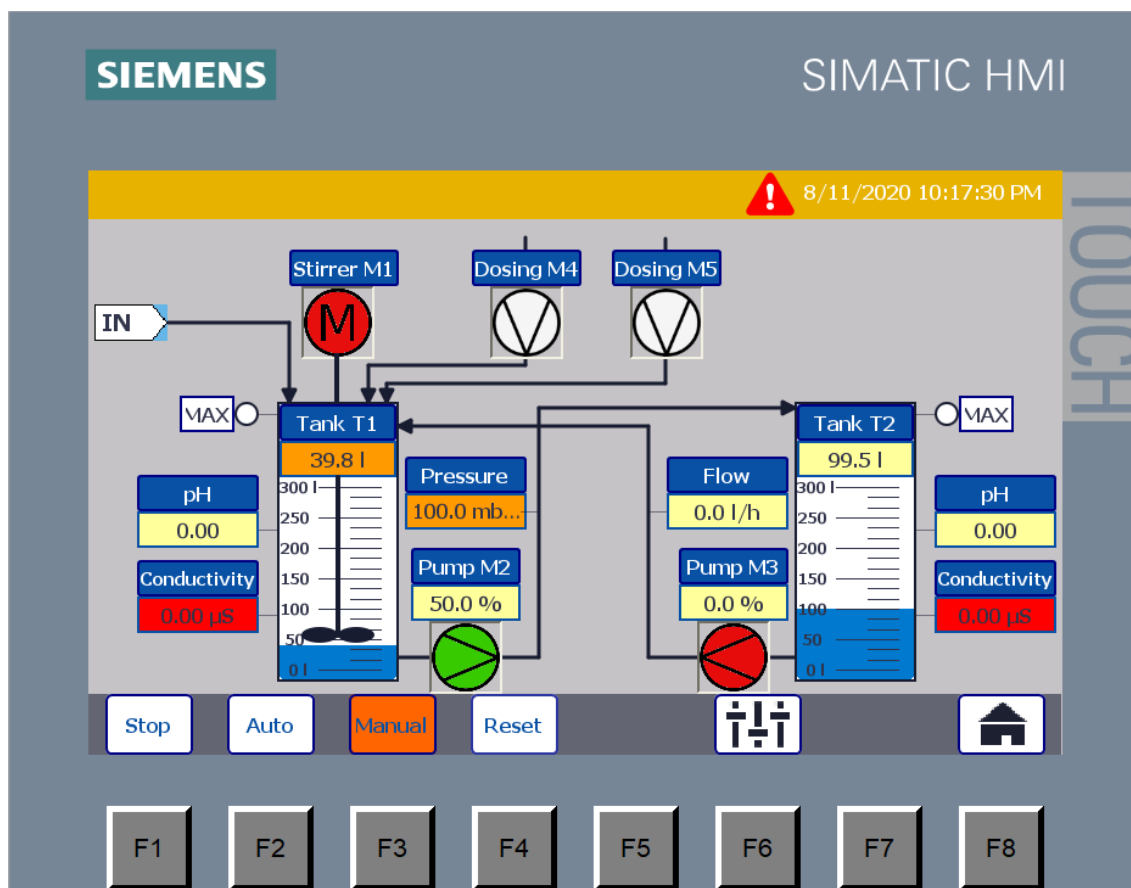
-  Komponenta je v poruše,
-  Komponenta je bez poruchy,
-  Varování - hladina v nádrži je mimo varovné limity,
-  Chyba - hladina v nádrži je mimo chybové limity,
-  Přehled měřených veličin v nádrži,
-  Otevření okna s nastavením parametrů (viz podkapitola 6.2 a 6.3).

Hlavní obrazovka obsahuje přehled o stavu všech komponent. Každá komponenta má svůj název v zeleném pruhu a pokud je některá z nich v poruše, rozsvítí se u dané komponenty příslušná indikace poruchy. Poruchu lze poznat i podle barvy dané komponenty. Pokud je barva červená, je komponenta v poruše, pokud je barva šedá, je bez chyby (měření), popřípadě není spuštěná (čerpadlo, míchadlo) a nakonec pokud je barva modrá, značí to spuštěnou komponentu (čerpadlo, míchadlo). Na obrázku 6.2 se tedy míchadlo *Stirrer M1* nachází v chybě, čerpadlo *Pump M2* je vypnuté a není v chybě, a nakonec čerpadlo *Pump M3* je spuštěné na 50% výkon. Dále je velice důležité, aby bylo možné nastavit režim zařízení a zároveň vidět v jakém režimu se aktuálně nachází (OFF, AUTO, MANUAL) a také je nutné zajistit možnost resetovat chybu z jakékoliv obrazovky. To se provádí pomocí dvou tlačítek v pravém dolním rohu. Z obrázku je vidět, že zařízení bylo v režimu MANUAL. Po kliknutí na lištu v levém dolním rohu obrazovky je umožněno zvolit regulační úlohu. Je třeba připomenout, že to lze pouze pokud je v režimu OFF. Zkratky mají následující význam:

- Main = Hlavní obrazovka,
- WLR = Water Level Regulation (Regulace vodní hladiny),
- WFR = Water Flow Regulation (Regulace průtoku vody),
- WPR = Water Pressure Regulation (Regulace tlaku vody).

U obou nádrží je také možné vyčíst z obrazovky konkrétní hladinu vody. V nádrži *Tank 1* je 40 litrů a v druhé nádrži *Tank 2* je 99 litrů vody. Také je možné podle barev poznat, která hodnota hladiny je chybová a při které hodnotě je indikováno varování. U nádrže *Tank 1* je vidět, že daná hladina se už nachází pod mezí pro varování, proto je u této nádrže vidět ikona varování. Pokud by hladina klesla pod mez pro chybu, byla by vidět ikona pro chybu a zároveň by nebylo možné spustit čerpadlo *Pump M2*, aby nedošlo k jeho poškození z důvodu nedostatku vody. Bylo již řečeno, že každá komponenta (měření, čerpadlo) má své parametry. K jejich úpravě slouží příslušné pop-up okno, které se otevře kliknutím na kulatou ikonku klíče a ozubeného kola.

Při návrhu hlavní obrazovky druhého operátorského panelu KTP700B bylo přistoupeno z grafického hlediska rozdílně. Vzhled hlavní obrazovky panelu KTP700B je na obrázku 6.3.



Obrázek 6.3: Hlavní obrazovka na operátorském panelu KTP700B

Je vidět, že rozložením komponent se tato obrazovka neliší od předchozí obrazovky, avšak vzhled jednotlivých komponent má spíše charakter schématických značek. Také na této obrazovce nejsou ikonky, které by indikovaly stav dané komponenty. Ten je možné ale opět vyčíst z barvy dané značky. Míchadlo *Stirrer M1* se opět na tomto obrázku nachází v chybě, stejně tak i čerpadlo *Pump M3*, jelikož barva jejich značek je červená. Čerpadlo *Pump M2* je spuštěné, protože má zelenou barvu a dávkovací pumpy *Dosing M4* a *Dosing M5* jsou vypnuté, ale nejsou v poruše, proto mají barvu bílou. Indikace stavu pomocí barvy funguje stejně i u všech měření. Například měření vodivosti *Conductivity* na obou nádržích mají poruchu. Oranžová barva u hodnoty hladiny vody v nádrži *Tank 1* slouží k varování. Světle žlutá barva měření pak neindikuje žádnou poruchu a měření probíhá v pořádku. Tato barva značí, že dané pole je pouze pro čtení a není možné měnit jeho hodnotu. Pokud je pole bílé, znamená to, že naopak je možné jeho hodnotu upravovat. Tento princip značení je stejný u jakéhokoliv pole v jakémkoliv okně (obrazovce) na obou panelech (TP700C, KTP700B). Nastavit režim celého zařízení (OFF, AUTO, MANUAL) je možné z dolní lišty. Stejně tak zvolit regulační úlohu pomocí druhé ikony zprava. Tlačítko v pravém dolním rohu pak slouží k návratu na domovskou stránku, tedy hlavní obrazovku či obrazovku regulace.

6.2 Pop-up okno pro měření

V předchozí kapitole byla popsána hlavní obrazovka, z které je možné otevřít pop-up okna pro nastavení parametrů daných komponent. Je-li požadováno změnit parametry měření, je nutno kliknout na žlutou ikonku klíče s ozubeným kolem v modrém pruhu u daného měření. Každá tato ikonka nastavuje jiný index struktury měření (viz podkapitola 5.3). Po kliknutí na tuto ikonku se otevře okno na obrázku 6.4.

Hydrostatic pressure in Tank 1			
Description		Settings	
Common fault:	Inactive	Measured value max:	550.00
Raw measured value:	2000	Measured value min:	0.00
Output value:	39.79	Offset:	+0.00
Status:		Measuring start delay:	0
Limits	Activated		
Warning max:	230.000		
Warning min:	40.000		
Fault max:	250.000		
Fault min:	20.000		

Obrázek 6.4: Pop-up okno pro měření

Toto okno umožňuje nastavit všechny proměnné, které obsahuje datová struktura pro měření popsána v oddíle 4.2.1. V horní liště se nachází název měření, pro které je okno otevřeno. V tomto případě se jedná o měření hydrostatického tlaku tedy hladiny vody v nádrži číslo 1 (*Tank 1*). Také je dobré připomenout, že žlutě podkreslené pole značí, že není možné měnit jeho hodnotu. Naopak bílá barva pole značí, že je možné upravit hodnotu pole. Z tohoto důvodu jsou pole žlutá u chyby, analogové hodnoty signálu, zpracovaného signálu, stavu a bílá u nastavení limit, měřicího rozsahu, offsetu a zpoždění měření. Pokud by byly limity měření deaktivovány, příslušná pole pro nastavení hodnot by změnila barvu na žlutou. Chyba může být aktivní či neaktivní. Stav může být následující:

- Measuring stopped = měření zastaveno,
- Measuring active = aktivní měření bez chyby,
- Fault = chyba (překročení mezních analogových hodnot, přídavné chyby),

- Limit warning = varování (překročení mezních hodnot *Warning min* či *Warning max*),
- Limit fault = chyba (překročení mezních hodnot *Fault min* či *Fault max*).

6.3 Pop-up okno pro čerpadlo

Stejně jako okno měření se otevírá i pop-up okno pro čerpadlo, tedy žlutou ikonkou klíče s ozubeným kolem v modrém pruhu u daného čerpadla popřípadě míchadla. Tato ikona také nastavuje hodnotu indexu (viz podkapitola 5.3) dle požadovaného čerpadla. Jak vypadá otevřené okno pro čerpadlo lze vidět na obrázku 7.3.

Pump M3		Manual control	
Description		Start:	On
Contactor:	On	Manual value	50.0 %
Common fault:	Inactive	Statistics	
Auto value:	0.0	Operation hours:	0 Reset
Status:	MANUAL ON	Start number:	6 Reset
Settings			
Delay ON:	0		
Delay OFF:	0		

Obrázek 6.5: Pop-up okno pro čerpadlo

Obsah okna odpovídá datové struktuře pro čerpadlo popsané v oddíle 4.3.1. Je zde vidět, zdali je čerpadlo vypnuté či zapnuté, zdali je v chybě a také je možné vyčíst jaká je zadaná požadovaná hodnota výkonu v automatickém režimu. Čerpadlo je vždy v jednom z následujících stavů:

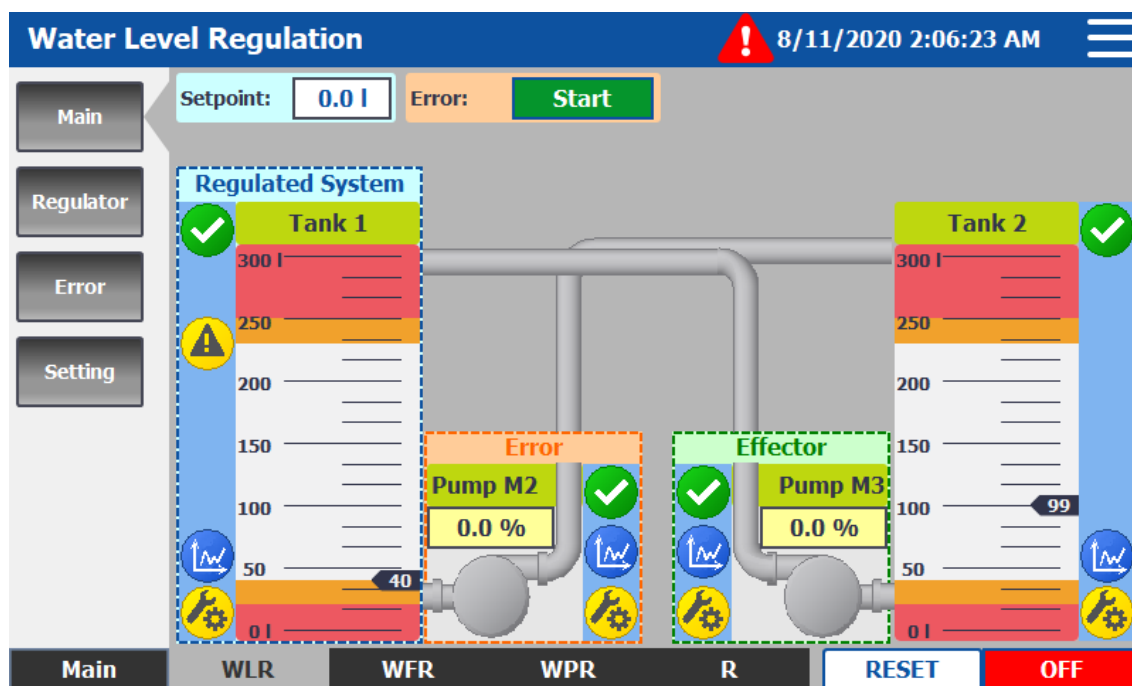
- Auto OFF = čerpadlo je vypnuté v automatickém režimu,
- Auto ON = čerpadlo je spuštěné v automatickém režimu,
- Fault = chyba (přídavné chyby),
- Manual OFF = čerpadlo je vypnuté v manuálním režimu,

- Manual ON = čerpadlo je spuštěné v manuálním režimu.

Důležitá je možnost obsluhy čerpadla v manuálním režimu. Právě z tohoto okna je možné čerpadlo manuálně spustit s nastaveným výkonem. Je důležité podotknout, že čerpadlo je možné spustit manuálně pouze pokud je celé zařízení v manuálním režimu. Pokud se jedná o čerpadlo musí být v přidružené nádrži dostatek vody, aby bylo možné spuštění čerpadla. Stejně pravidlo platí i pro míchadlo, aby lopatka míchadla nenarazila na hladinu vody, což by mohlo vést k jejímu zničení. Pokud je čerpadlo spuštěno v automatickém režimu, lze nastavit zpoždění spuštění i vypnutí. V oknu je možné i zjistit počet operačních hodin čerpadla a počet jeho spuštění.

6.4 Obrazovka pro úlohu regulace hladiny

V popisu hlavní obrazovky byla zmíněna lišta, ze které je možné zvolit regulační úlohu. Tato lišta se nachází v dolní části obrazovky (viz například podkapitola 6.1) a po kliknutí na nápis WLR se otevře obrazovka určená pro regulaci hladiny. Jedná se o upravenou verzi hlavní obrazovky. Z obrazovky byly odebrány dávkovací čerpadla, míchadlo a měření nepodstatná pro regulaci hladiny. Na obrazovce zůstávají obě nádrže a obě čerpadla. Tato obrazovka je po otevření nastavena jako domovská a jak již bylo řečeno, změnit obrazovku úlohy regulace je možné pouze ve stavu OFF. Vzhled obrazovky je následující:



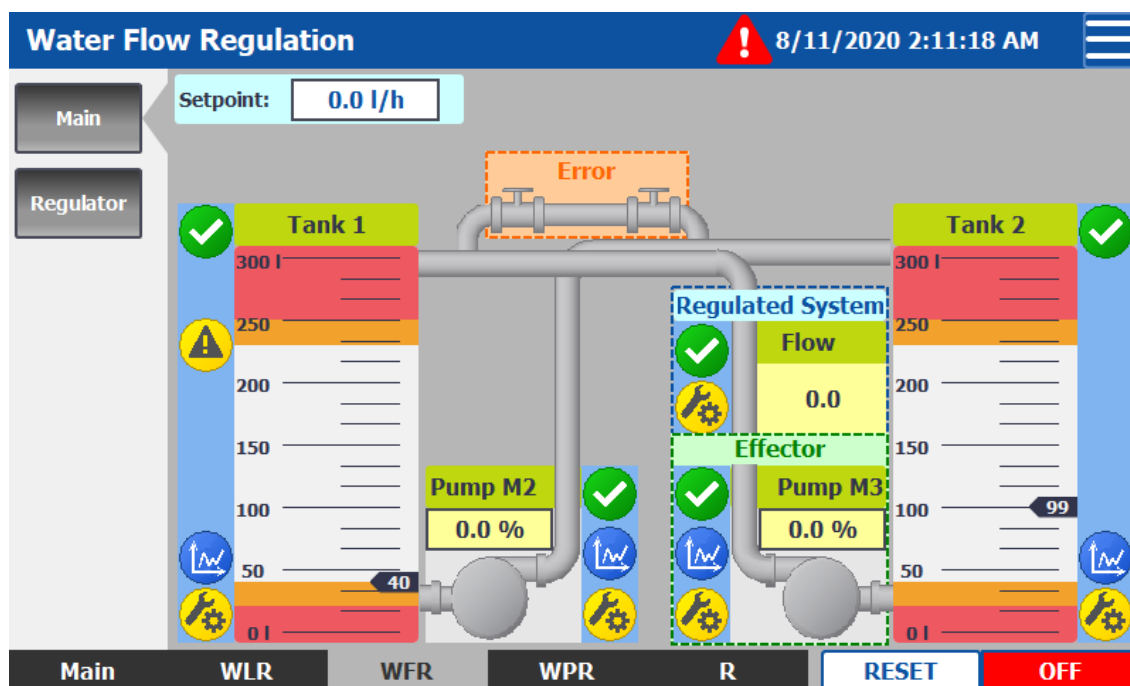
Obrázek 6.6: Obrazovka pro úlohu regulace hladiny

Kromě toho, že této obrazovce byly odebrány některé komponenty, byla zde přidána lišta na levé straně pro přepínání mezi hlavní stránkou úlohy (Main), kon-

figurace regulátoru (*Regulator*), konfigurace generátoru signálu (*Error*) a nakonec nastavení samotné regulace (*Setting*). Stisknutím tlačítka *Regulator* se otevře obrazovka pro regulátor (viz podkapitola 6.7). Stisknutím tlačítka *Error* se otevře obrazovka pro generátor signálu (viz podkapitola 6.8). Hlavní stránka obsahuje navíc označení regulovaného systému (*Regulated System*), tedy nádrže ve které je nastavena požadovaná hladina, také obsahuje označení efektoru (*Effector*), tedy regulovaného čerpadla, a nakonec je zde i označení chyby (*Error*), tedy čerpadla jež generuje chybu v regulované soustavě. Z *Main* stránky je také možné přímo nastavit požadovanou hodnotu hladiny a spustit generování chyby.

6.5 Obrazovka pro úlohu regulace průtoku

K obrazovce určené pro regulaci průtoku z nádrže *Tank 2* do nádrže *Tank 1* se přistupuje stejně jako při přístupu k jiné regulační úloze. Pokud je požadavek na otevření regulace průtoku, musí být stisknut nápis *WFR* (Water Flow Regulation) na spodní liště obrazovky. Podmínka pro otevření je stejná, zařízení musí být ve stavu *OFF*. Opět se jedná o upravenou verzi hlavní obrazovky. V tomto případě není možné měnit efektor, jelikož zařízení obsahuje pouze jedno měření průtoku, a to z nádrže *Tank 2* do nádrže *Tank 1*. Proto jsou opět odebrána dávkovací čerpadla, míchadlo a ostatní měření. Měření hladiny v nádrži zůstává. Oproti obrazovce určené pro regulaci hladiny je přidáno měření průtoku. Vzhled obrazovky lze vidět na obrázku 6.7.



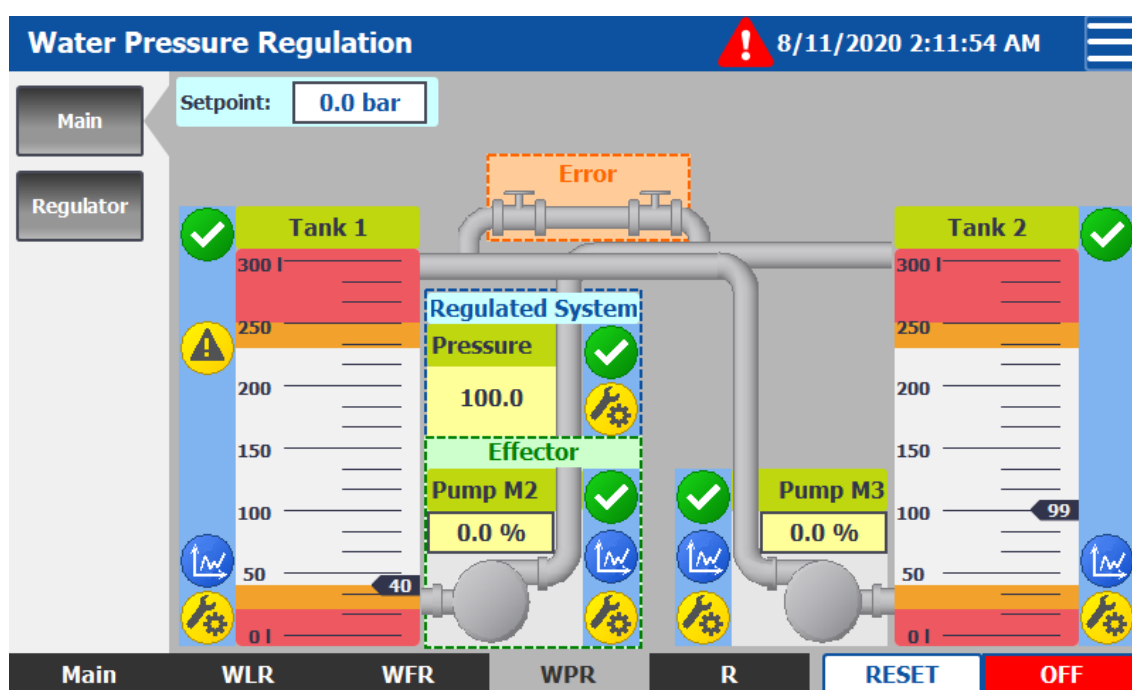
Obrázek 6.7: Obrazovka pro úlohu regulace průtoku

Kromě hlavní stránky (*Main*) je možné přepnout na obrazovku nastavení regulátoru (*Regulator*) popsané v podkapitole 6.7. Tato obrazovka neobsahuje nastavení

chyby, jelikož není možné jakýmkoliv způsobem generovat chybu prostřednictvím signálu z PLC. To je možné pouze manuální úpravou příslušného ventilu na zařízení, proto je chyba (*error*) vykreslena pouze pomocí manuálních ventilů.

6.6 Obrazovka pro úlohu regulace tlaku

Poslední dostupná obrazovka pro regulaci je obrazovka určená pro regulaci tlaku. Přístup je stejný jako k ostatním regulacím (bližší popis v podkapitolách 6.4, 6.5). Vzhled této obrazovky je téměř stejný jako u obrazovky pro regulaci průtoku. Liší se pouze odebráním měření průtoku a přidáním měření tlaku, které probíhá z nádrže *Tank 1* do nádrže *Tank 2*. Proto se mění tedy i efektor a jeho označení, které je nyní na čerpadle *Pump M2*. Vzhled obrazovky je na obrázku 6.8.

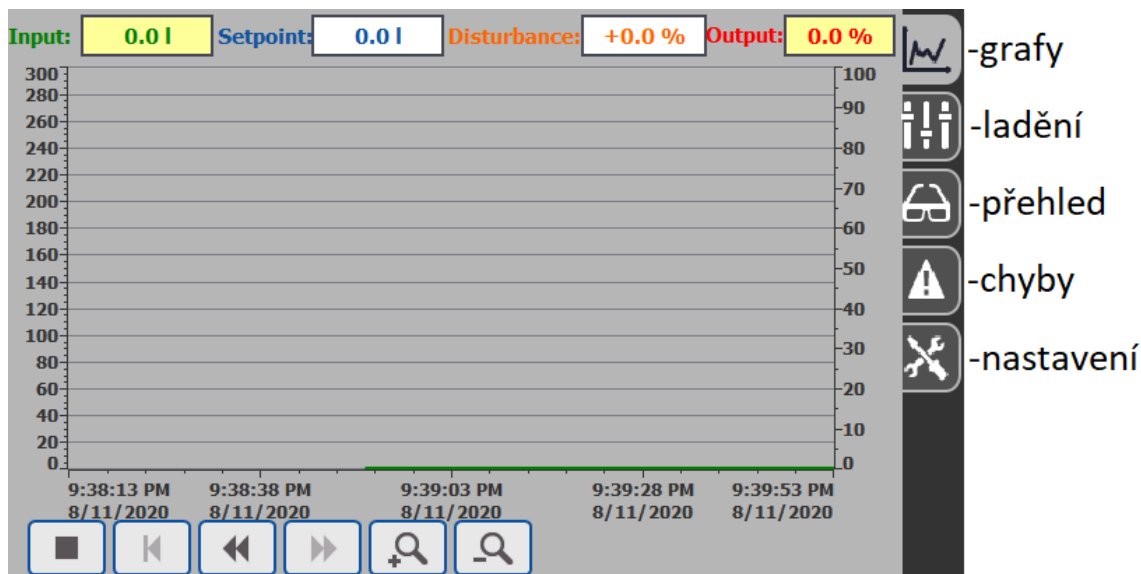


Obrázek 6.8: Obrazovka pro úlohu regulace tlaku

6.7 Obrazovka pro regulátor

Obrazovka pro regulátor je dle uvedené struktury na obrázku 6.1 přístupná pouze z obrazovek pro regulační úlohy (podkapitola 6.4, 6.5 a 6.6). I když se jedná o komplex více obrazovek bude uvedena pouze hlavní obrazovka pro regulátor. Vedlejší obrazovky budou popsány slovně. Stejně jako pop-up okno pro měření a pro čerpadlo je i struktura regulátoru selektována dle indexu, který se v případě regulace volí již na spodní liště obrazovky. Datová struktura obsahuje daleko více proměnných, než struktura pro měření a čerpadlo. Z tohoto důvodu bylo nutné vytvořit více obrazovek a navrhnout novou lištu pro přepínání mezi těmito obrazovkami. Pokud

dojde k otevření regulačního okna pomocí tlačítka Regulator z kteréhokoliv okna pro regulaci, otevře se okno na obrázku 6.9.



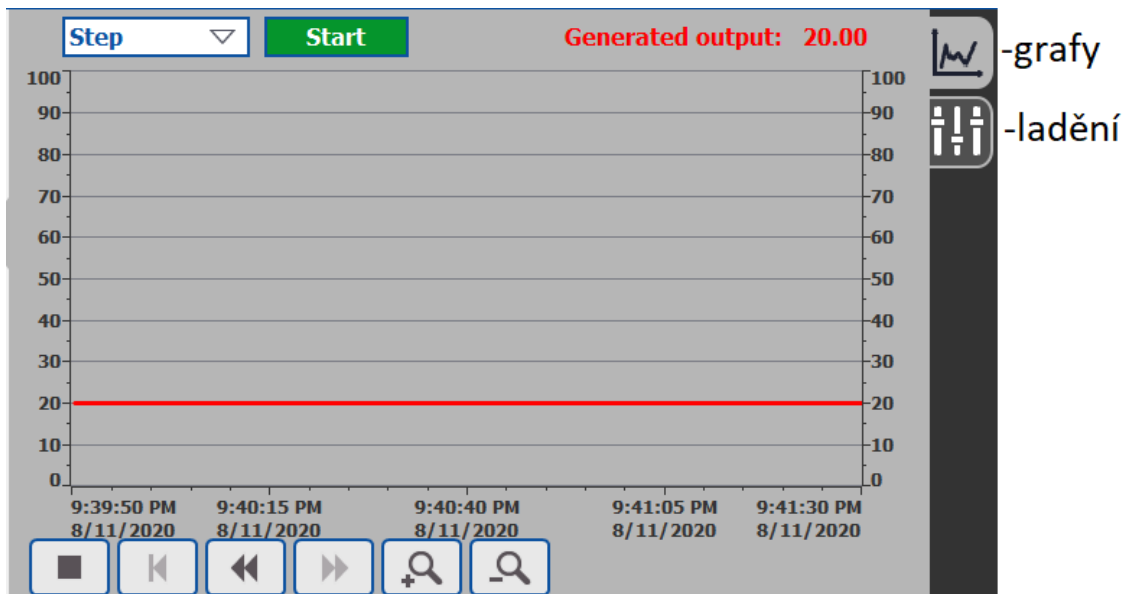
Obrázek 6.9: Výchozí obrazovka pro nastavení regulátoru

Toto je okno s grafem, které slouží k monitorování průběhu signálu na vstupu regulátoru, výstupu regulátoru a požadované hodnoty. Opět je zde pravidlo, že pokud je pole žluté, jedná se pouze o pole pro čtení. Pole *Setpoint* pak slouží k nastavení požadované hodnoty regulace. Pole *Disturbance* pak slouží k nastavení rušení. Na pravé straně lze vidět další lištu s ikonami, které slouží pro přepínání mezi okny určenými pro samostatný regulátor.

Okno ladění slouží k nastavení a spuštění ladění regulátoru. V tomto okně se volí typ ladění (Pre-tuning, Fine tuning), metoda ladění atd. Okno přehled obsahuje obrázek funkčního bloku PID_Compact a aktuální hodnoty na jeho vstupech a výstupech. Okno chyby slouží k přehledu chyb a varování daného regulátoru, nikoliv chyb celého zařízení. Poslední okno nastavení slouží pro modifikaci limit regulátoru, jako je minimální a maximální hodnota na vstupu, výstupu, škálování atd.

6.8 Obrazovka pro generátor signálu

Poslední obrazovka, která bude popsána je obrazovka pro generátor signálu, která se otevírá tlačítkem Error z obrazovky pro regulaci hladiny (viz podkapitola 6.4). Rozložení a vzhled této obrazovky je na obrázku 6.10.



Obrázek 6.10: Výchozí obrazovka pro nastavení generátoru signálu

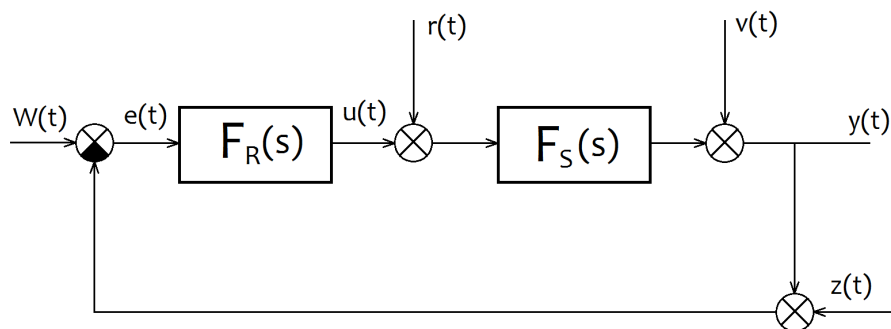
I tato obrazovka, stejně jako obrazovka pro regulátor, obsahuje lištu pro přepínání mezi grafy a laděním. Na obrazovce s grafem se vykresluje generovaný signál. Také je z této obrazovky možné změnit typ generovaného signálu a generování vypnout, či zapnout. Obrazovka označená jako ladění obsahuje parametry generovaného signálu. Tyto parametry odpovídají datové struktuře uvedené v oddíle 4.5.1.

Kapitola 7

Regulační úlohy

Tato kapitola se zabývá návrhem regulačních úloh, které mají za úkol demonstrovat chod výukového zařízení. Úkolem bylo navrhnout regulační úlohy tak, aby bylo využito co nejvíce instalovaných komponent zařízení. Překážka v návrhu některých úloh je absence odtoku na odpad v konstrukci zařízení (viz kapitolu 1). Proto byly z návrhu vyřazeny úlohy, které vyžadují dávkování chemických látek do vody. Takové regulační úlohy by vyžadovaly zajištění přívodu čerstvé vody, a tedy i odvodu vody přebytečné jelikož se jedná o izolovanou soustavu. Výsledkem jsou tři regulační úlohy, které pracují s neupravenou vodou a nevyžadují přívod „čisté“ vody. U každé regulační úlohy pak bylo využito výhradně již implementovaného funkčního bloku pro regulaci PID_Compact (viz podkapitola 3.4) od firmy Siemens.

Regulace je významný proces ve velké části průmyslových aplikací. Regulací je rozuměn zásah regulátoru do regulovaného systému (např. čerpadlo, topení atd.) tak, aby byla dosažena požadovaná hodnota určité veličiny (např. tlak, teplota). Základem regulace je regulátor a regulovaný systém, jenž tvoří regulační smyčku. Pro zajištění co nejpřesnější regulace je vhodné zavést zpětnou vazbu do regulátoru, tzv. regulační odchylku. Jedná se o výstup regulovaného systému (regulovaný výstup) odečtený od požadované hodnoty (referenční signál). Takto sestavenou soustavu lze nazývat regulační smyčka se zápornou zpětnou vazbou a její uspořádání lze vidět na obrázku 7.1.



Obrázek 7.1: Regulační smyčka [6]

kde je

$F_R(s)$ – regulátor,
 $F_S(s)$ – řízený systém,
 $w(t)$ – referenční signál,
 $e(t)$ – regulační odchylka $e(t) = w(t) - y(t)$,
 $u(t)$ – akční zásah regulátoru,
 $r(t)$ – aditivní porucha na vstupu systému,
 $v(t)$ – porucha na výstupu systému,
 $y(t)$ – regulovaný výstup,
 $z(t)$ – porucha ve zpětné vazbě (porucha senzoru).

7.1 Regulace hladiny

První regulační úloha, která byla na zařízení implementována, je úloha regulace hladiny vody v jedné z přítomných nádrží. Zařízení má k dispozici dvě nádrže, proto je možné zvolit ve které nádrži bude hladina regulována. V této konfiguraci zařízení není možné regulovat obě hladiny v nádržích současně, jelikož se jedná o izolovanou soustavu. V kapitole 1 bylo popsáno, že každá nádrž je spojena přes čerpadlo do druhé nádrže. Regulace hladiny tedy může probíhat dvěma způsoby. Buď je čerpadlem do regulované nádrže voda přiváděna z druhé nádrže, nebo je naopak voda odčerpávána z regulované nádrže. Důležité je, aby čerpadlo které není regulováno sloužilo jako generátor chyby. Je-li totiž dosaženo požadované hladiny v regulované nádrži a voda není odčerpávána (popřípadě přičerpávána) pryč, do nádrže je stále přiváděna voda a dojde k překročení požadované hladiny. Je to z důvodu konfigurace čerpadla, jelikož není doporučeno spustit čerpadlo pod určité procento výkonu. Měření hydrostatické tlaku zajišťuje senzor popsany v oddílu 1.3.2).

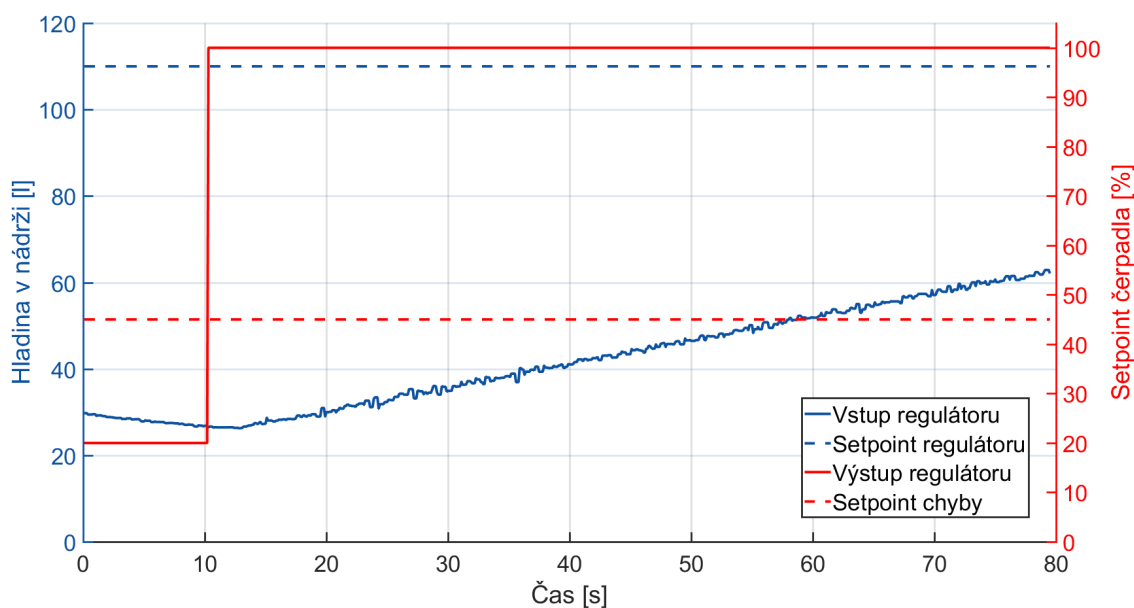
Před návrhem regulační úlohy bylo nutné zajistit, aby obě čerpadla neběžela bez dostatečné hladiny v nádrži, jelikož by došlo k zavzdušnění čerpadla. Proto bylo využito možnosti natavení limit v měřícím bloku hladiny (viz oddíl 4.2.1). Pokud by hladina klesla pod nastavenou minimální chybovou hladinu, došlo by k odstavení čerpadla pomocí blokace automatického či manuálního chodu (viz podkapitola 4.3). Měřenou hodnotu hladiny, tedy výstup z měřícího bloku příslušné nádrže bylo nutné přivést na vstup funkčního bloku pro regulaci (viz podkapitola 4.4). Požadovaná hodnota se zadává z obrazovky operátorského panelu (viz podkapitola 6.4).

Poté co byl připraven algoritmus v PLC, bylo použito ladících algoritmů funkčního bloku `PID_Compact` (viz 3.4). Jak již bylo popsáno, tento blok umožňuje automaticky naladit koeficienty regulátoru a to pomocí tzv. Pre-tuningu (před-ladění) a Fine tuningu (doladění). Před spuštěním ladících algoritmů byly nastaveny parametry regulátoru. Pro regulaci hladiny v nádrži byl zvolen PI regulátor. Derivační složka regulátoru byla tedy nulová, a to z důvodu padající vody na hladinu, která způsobovala šum měření, což by mělo na regulátor s derivační složkou negativní vliv. Po zvolení typu regulátoru byl nejdříve spuštěn algoritmus Pre-tuningu. Je vhodné připomenout, že díky úpravě tohoto bloku v podobě funkčního bloku `PID_Control` (viz podkapitola 4.4), je možné ladění spustit pouze pokud je zařízení v manuálním režimu. Dále bylo přistoupeno k doladění parametrů, tedy spuštění Fine tuningu.

U Fine tuningu bylo možné zvolit ladící metodu koeficientů. V této úloze byla zvolena Ziegler-Nichols metoda pro PI regulátor. Tyto algoritmy stanovily koeficienty regulátoru. Funkčnost tohoto regulátoru s danými parametry byla ověřena několika měřeními s různými průběhy chybového signálu.

7.1.1 Průběh Pre-tuningu

Pro ladění koeficientů byl prvně spuštěn ladící algoritmus Pre-tuning. Aby bylo možné tento algoritmus spustit, bylo nutné dodržet podmínku 3.2 uvedenou v podkapitole 3.4. Jelikož byl v regulaci nataven horní limit vstupu na 250 l, musel být minimální rozdíl mezi požadovanou hodnotou (setpoint) a aktuální hodnotou v nádrži, tedy vstupem bloku regulátoru (input), alespoň 75 l. Dodržení této podmínky umožnilo spustit ladění regulátoru. Průběh ladění je znázorněn na grafu 7.2.

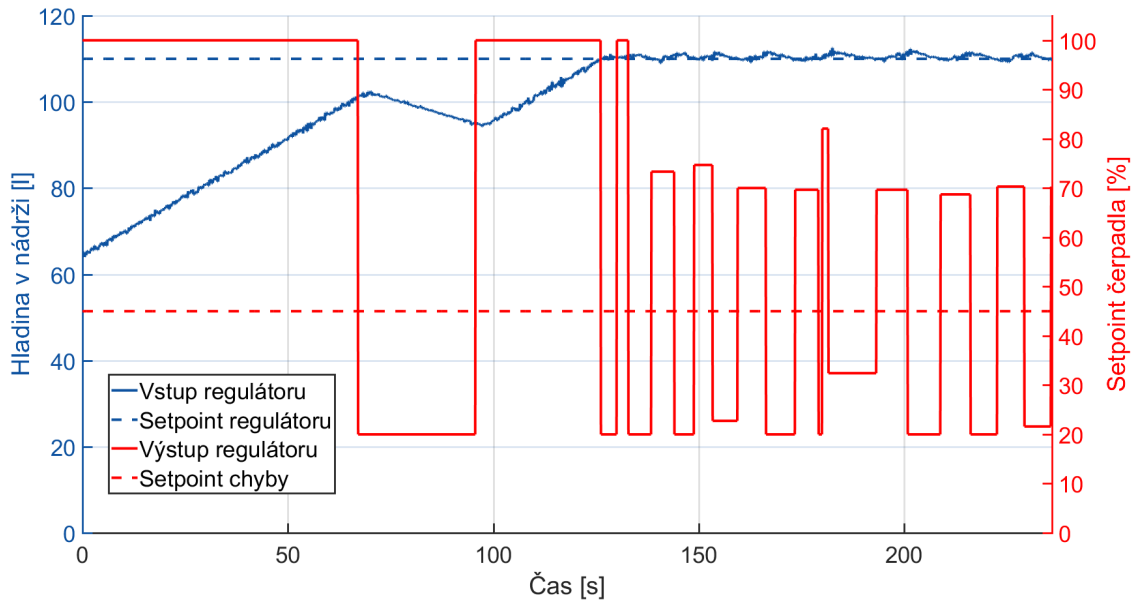


Obrázek 7.2: Průběh Pre-tuningu u úlohy regulace hladiny

Ladění regulátoru bylo spuštěno a začíná v čase 0 s. V tento čas je na výstupu regulátoru 20% výkon, jenž je odeslán na vstup funkčního bloku pro čerpadlo (4.3). Na výstupu regulátoru není 0% výkon, jelikož byla nastavena v parametrech regulátoru minimální hodnota výstupu na 20 %. Je také vidět, že hladina v nádrži při tomto výkonu klesá, protože druhé čerpadlo, které slouží jako chybové je spuštěno a odčerpává vodu z nádrže. V čase kolem 10 s je výkon skokově změněn na 100 %. V tento okamžik začíná hladina v nádrži stoupat a algoritmus hledá inflexní bod, který je nalezen v 80 s. Algoritmus v tento čas končí a na základě růstu hladiny vypočítá koeficienty, které jsou uloženy do příslušné struktury. Koeficienty jsou popsány v tabulce 7.1, jenž se nachází v oddílu 7.1.3.

7.1.2 Průběh Fine tuningu

Poté co byl dokončen Pre-tuning bylo možné spustit algoritmus Fine tuningu, který by měl stanovit na základě Pre-tuningu lepší koeficienty regulace. Princip Fine tuningu je popsán v oddílu 3.4.2. Jak vypadá jeho průběh je znázorněno na grafu 7.3.



Obrázek 7.3: Průběh Fine tuningu u úlohy regulace hladiny

Z grafu je možné vyčíst, že nejdříve se regulátor snaží dosáhnout požadované regulované veličiny. Jakmile je dosaženo blízkosti požadované regulované veličiny, v tomto případě 110 l, začnou se na výstupu regulátoru generovat oscilace. V grafu tyto oscilace začínají přibližně v 70 s. Z amplitudy a frekvence pak stanoví koeficienty regulátoru. Po dokončení Fine tuningu byly dané koeficienty uloženy do struktury aktuálních parametrů regulátoru a parametry z Pre-tuningu byly přesunuty do struktury určené k zálohování. Vyhodnocené koeficienty jsou uvedeny v tabulce 7.1, kterou lze nalézt v následujícím oddílu.

7.1.3 Stanovené konstanty

Z obou ladících algoritmů byly stanoveny koeficienty regulátoru. Tyto koeficienty jsou uvedeny v tabulce 7.1.

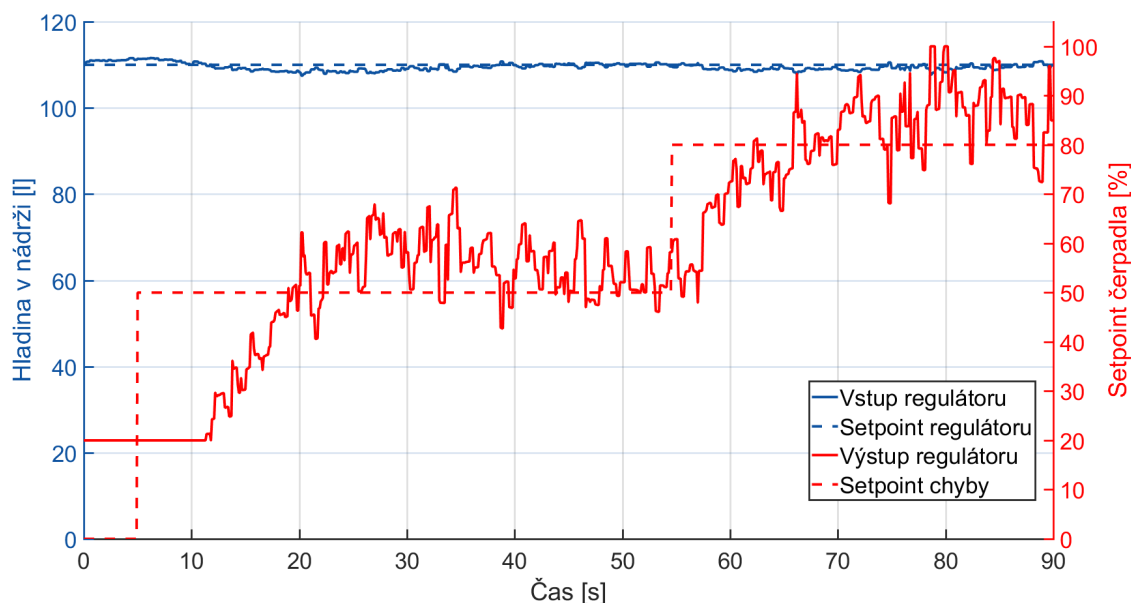
Symbol	Popis	Pre-tuning	Fine tuning
K_p	Proporcionální zesílení	34.3	13.0
T_I	Integrační konstanta	10.3 s	11.0 s
T_D	Derivační konstanta	0.0 s	0.0 s
a	Koeficient derivačního zpoždění	0.1	0.1
b	Vážení proporcionální zásahu	0.8	0.8
c	Vážení derivačního zásahu	0.0	0.0

Tabulka 7.1: Koeficienty regulátoru hladiny v nádrži

Z 7.1 tabulky lze vidět, že proporcionální zesílení stanovené pomocí Pre-tuningu bylo sníženo po dokončení Fine tuningu. Integrační konstanta byla mírně zvětšena a derivační konstanta zůstala samozřejmě nulová, jelikož byl zvolen PI regulátor. Nižší hodnota proporcionálního zesílení zajistí nižší skoky výstupu regulátoru. Regulované čerpadlo tak nebude tolik namáháno skoky hodnoty požadovaného výkonu.

7.1.4 Skoková chyba regulace

S parametry stanovenými z algoritmu Fine tuning bylo možné otestovat regulaci. Testování probíhalo změnou výkonu čerpadla, které sloužilo pro generování chyby, tj. odběru vody z nádrže. Požadovaná hladina při tomto testu byla konstantní. Výkon čerpadla odebírající vodu z nádrže byl skokově dvakrát změněn. Průběh testu lze vidět v grafu 7.4.



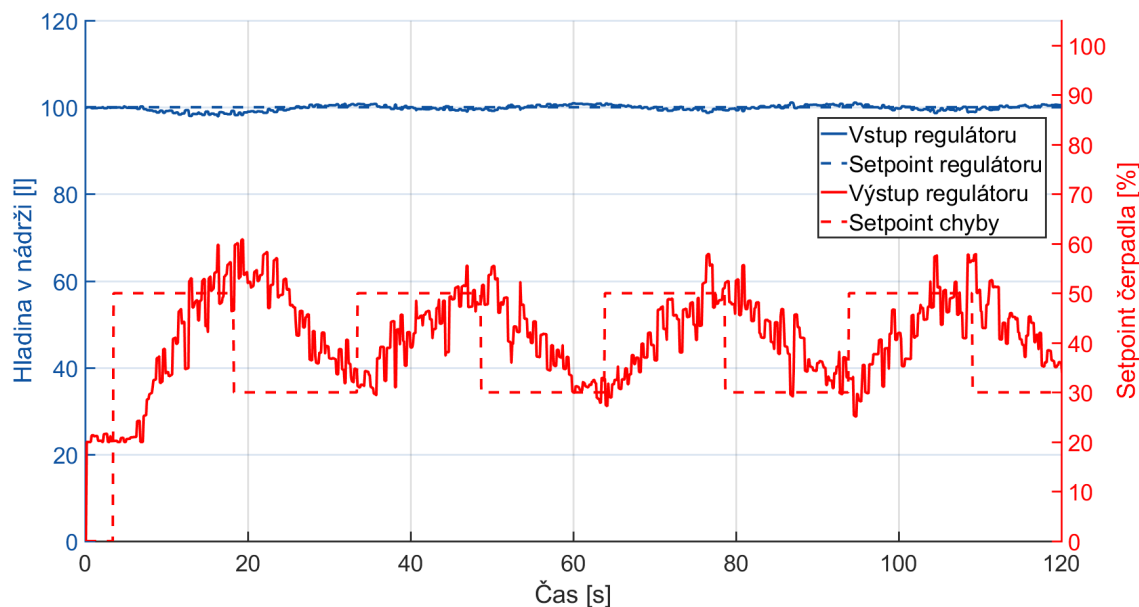
Obrázek 7.4: Průběh regulace na skokový signál chyby

Z grafu 7.4 lze vidět, že požadovaná hladina v regulované nádrži byla 110 l. Zpočátku bylo čerpadlo, které odčerpává vodu z regulované nádrže, tedy způsobuje

chybu regulace, nastaveno na 0% výkon. V čase 5 s skočila hodnota jeho výkonu na 50 % výkonu. Lze vidět, že voda v nádrži začala klesat do okamžiku, kdy regulátor zvýšil výkon čerpadla dostatečně na to, aby vyrovnal chybu tedy pokles hladiny. Také lze vidět, že hladina v regulované nádrži se pohybuje maximálně v rozmezí do 2 l od požadované hodnoty. Dále je splněn požadavek na hodnotu výstupu regulátoru, tedy výkon který je přiváděn na regulované čerpadlo, jenž by se neměl měnit příliš skokově v krátkém časovém okamžiku. Tím jsou myšleny například početné skoky ze 90 % na 30 %, které by po delší době mohly mít negativní vliv na čerpadlo. V čase 54 s byl proveden další skok chyby z 50 % na 80 %. Regulátor opět zareagoval zvýšením hodnoty svého výstupu a vyrovnal hladinu zpět na 110 l.

7.1.5 Obdélníková chyba regulace

Dalším využitým typem signálu z generátoru signálů (viz podkapitola 4.5) byl obdélníkový tvar. Perioda generátoru byla nastavena na 30 s, šířka pulsu na 15 s, amplituda činila 20 % a offset 30 %. Pomocí generátoru byl tedy vygenerován periodický čtvercový signál, jenž byl opět přiveden na vstup čerpadla určeného pro odčerpávání vody z regulované nádrže.

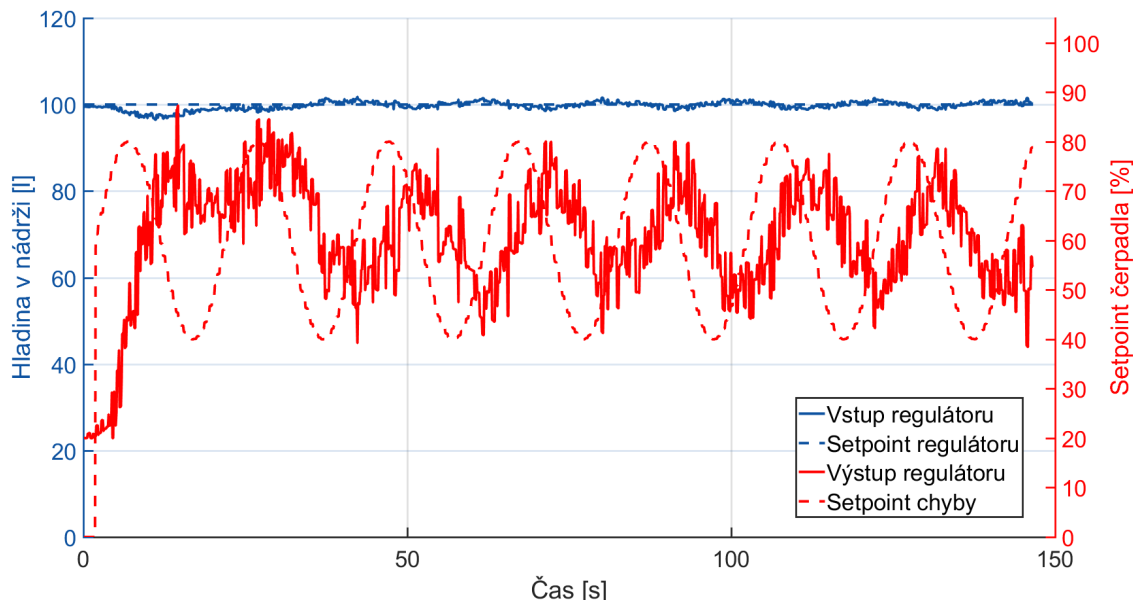


Obrázek 7.5: Průběh regulace na obdélníkový signál chyby

Graf 7.5 zobrazuje odezvu regulátoru na chybový signál generátoru, jenž byl spuštěn ve 4 vteřině. Požadovaná hodnota v regulované nádrži byla nastavena na 100 l. Lze vidět, že hladina se opět pohybuje ve velmi malém rozmezí požadované hodnoty. Výstup regulátoru má sinusový průběh.

7.1.6 Sinusová chyba regulace

Jako poslední byl nastaven sinusový typ na generátoru signálu. Perioda generátoru byla zvolena 20 s, amplituda byla 20 % a offset 60 %. Opět byl tento signál přiveden na vstup čerpadla určeného pro generování chyby regulace.



Obrázek 7.6: Průběh regulace na sinusový signál chyby

Z grafu 7.6 je zřejmé, že výstup regulátoru, tedy výkon regulovaného čerpadla má sinusový průběh stejně jako výkon chybového čerpadla. Tento signál na výstupu je však opožděn oproti chybovému signálu. I tento typ signálu, obdobně jako předchozí uvedené signály, neměl negativní vliv na průběh regulace. Hladina se stále pohybuje v blízkosti požadované hodnoty. Odchylka od požadované hodnoty činí maximálně 2 l.

7.2 Regulace průtoku

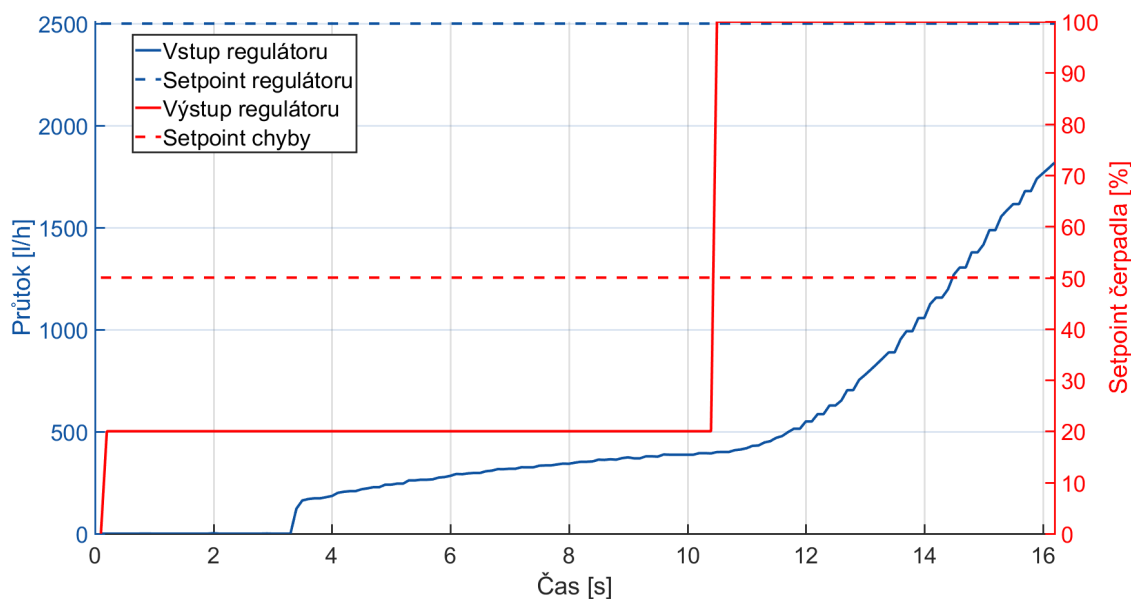
Jako druhá regulace, která byla navržena na výukové zařízení, je regulace průtoku vody z druhé nádrže *Tank 2* do nádrže *Tank 1*. Průtokoměr, který byl v této úloze použit je popsán v oddíle 1.3.1. Rozdíl v této úloze oproti předchozí je nemožnost způsobit v regulaci chybu pomocí některé z přítomných komponent. Chybu by bylo možné způsobit úpravou manuálního ventilu. Průtok vody se v této úloze pohybuje v rozmezí 0 až 3000 l/h. K regulaci byl opět použit funkční blok `PID_Control` popsaný v podkapitole 4.4.

Postup při stanovení parametrů byl obdobný jako u regulace hladiny. Nejdříve byl spuštěn Pre-tuning proces, který stanovil koeficienty regulátoru. Poté byl zvolen Fine tuning, který na základě parametrů z Pre-tuningu určil přesnější parametry pro regulaci. U této úlohy byla zvolena PID struktura regulace, jelikož zde nedo-

chází k velkému zašumění signálu a náběh otáček čerpadla, tedy nárůst průtoku je relativně rychlý.

7.2.1 Průběh Pre-tuningu

Pro ladění parametrů byl opět jako první spuštěn implementovaný algoritmus Pre-tuning funkčního bloku `PID_Control`. Stejně jako v předchozí úloze, bylo i u této úlohy nutné dodržet podmínku 3.2 uvedenou v podkapitole 3.4. Maximální povolený vstup regulátoru činil 3500 l/h. To tedy znamená, že minimální zadaná požadovaná hodnota průtoku musela činit minimálně 1050 l/h, aby byl spuštěn Pre-tuning. Průběh procesu Pre-tuning je zobrazen v grafu 7.7.

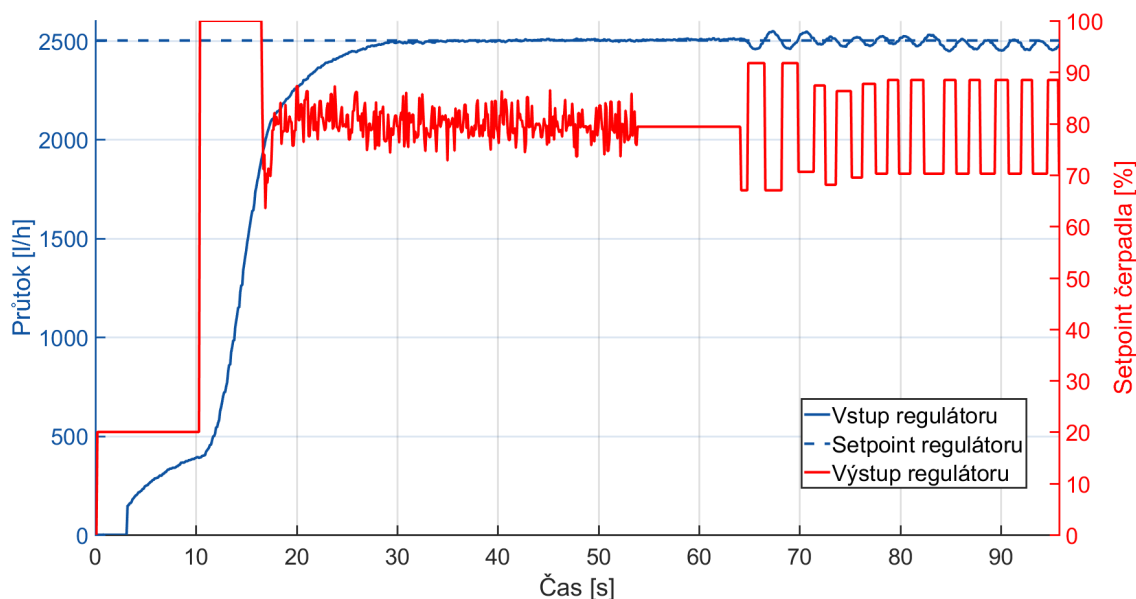


Obrázek 7.7: Průběh Pre-tuningu u úlohy regulace průtoku

Po spuštění tento algoritmus nastavil výstup regulátoru, tj. otáčky čerpadla, na maximální povolenou hodnotu, v tomto případě opět 100 %. Skok výstupu lze vidět v čase 10.4 s. To způsobilo růst průtoku potrubím a algoritmus začal vyhledávat inflexní bod. Po dosažení inflexního bodu byl algoritmus automaticky zastaven a nalezené parametry, stanovené z maximálního růstu regulovaného signálu, byly uloženy do aktuálních parametrů datové struktury regulátoru. Parametry jsou zapsány v tabule 7.2 v oddílu 7.2.3. Pro zjištění přesnějších parametrů byl spuštěn proces Fine tuningu popsáný v následujícím oddílu.

7.2.2 Průběh Fine tuningu

Pro získání lepších parametrů byl opět spuštěn Fine tuning (viz oddíl 3.4.2). Fine tuning u regulační úlohy průtoku má průběh vykreslený na obrázku 7.8.



Obrázek 7.8: Průběh Fine tuningu u úlohy regulace průtoku

Graf zobrazuje nárůst skutečné regulované veličiny (vstup regulátoru) na zadanou regulovanou veličinu 2500 l/h. V 64 s začal regulátor generovat oscilace na výstupu, aby z nich pak následně mohly být vypočteny koeficienty regulátoru obdobně jako v předchozí regulační úloze. Výsledné parametry tohoto Fine tuningu jsou uvedeny v tabulce 7.2.

7.2.3 Stanovené konstanty

Oba algoritmy vypočítaly parametry regulátoru, jejichž hodnoty jsou vypsány v tabulce 7.2.

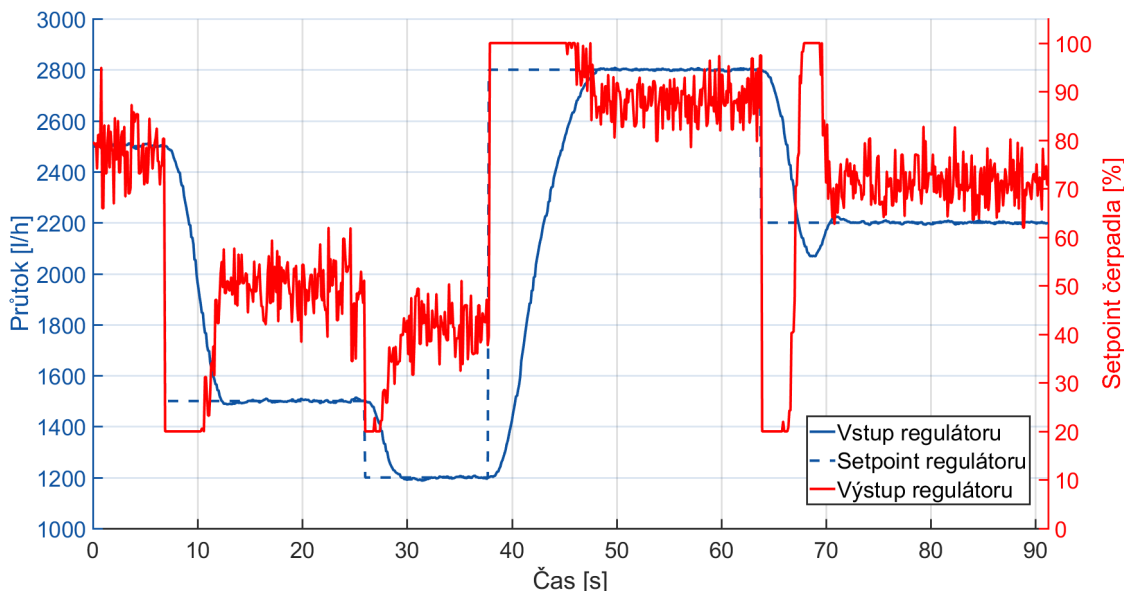
Symbol	Popis	Pre-tuning	Fine tuning
K_p	Proporcionální zesílení	0.1	0.4
T_I	Integrační konstanta	4.9 s	1.2 s
T_D	Derivační konstanta	0.9 s	0.3 s
a	Koeficient derivačního zpoždění	0.1	0.1
b	Vážení proporcionální zásahu	0.8	0.3
c	Vážení derivačního zásahu	0.0	0.0

Tabulka 7.2: Koeficienty regulátoru průtoku

Proporcionální zesílení regulátoru se pohybuje v desetínách, jelikož hodnota vstupního signálu se pohybuje v tisících a výstup se pohybuje v hodnotách do 100. Malá hodnota zajišťuje malé skoky výstupu regulátoru, tj. malé skoky výkonu na regulovaném čerpadle. Proces Fine tuningu hodnoty integrační a derivační časové konstanty snížil. Pro ověření funkčnosti regulátoru s parametry stanovenými Fine tuningem byl proveden test, jenž je popsán v následujícím oddílu.

7.2.4 Regulace na skokovou změnu požadované hodnoty

U regulace průtoku nebyl využit generátor signálu. Do regulace nebyla ani vnesena žádná umělá chyba, která by ovlivnila její průběh. K otestování parametrů stanovených pomocí Pre-tuningu a poté Fine tuningu byl proveden test, ve kterém se skokově měnila požadovaná hodnota průtoku. Jak se regulátor choval při tomto testu je znázorněno v grafu 7.9.



Obrázek 7.9: Průběh regulace se skokovou změnou požadovaného průtoku

Graf znázorňuje regulaci, která už byla v procesu. V čase 0 byla hodnota průtoku na hodnotě 2500 l/h, což odpovídá zadané požadované hodnotě. V čase 7 s byla požadovaná hodnota změněna na hodnotu 1500 l/h. Ve stejný okamžik došlo okamžitě ke snížení výkonu regulovaného čerpadla na 20 % výkonu, což odpovídá minimální povolené hodnotě výstupu regulátoru. Hodnota průtoku začala klesat a jakmile se dostala do blízkosti požadované hodnoty, regulátor zvýšil výkon čerpadla, aby nedošlo k přílišnému poklesu pod zadanou hodnotu průtoku. Hodnota průtoku klesla v jednotkách l/h pod požadovanou hodnotu a poté došlo k vyrovnání. Odezvu regulátoru na nastavení vyšší požadované hodnoty než je předchozí, lze vidět v čase 38 s. Zde dojde ke skoku z hodnoty 1200 l/h na 2800 l/h. Výkon čerpadla se okamžitě zvýší na 100 % a před dosažením průtoku zadané požadované hodnoty se výkon sníží přibližně na 90 %, aby se zamezilo přeregulování. S uvedeného grafu lze tvrdit, že parametry stanovené ladicím algoritmem jsou dostačující.

7.3 Regulace tlaku

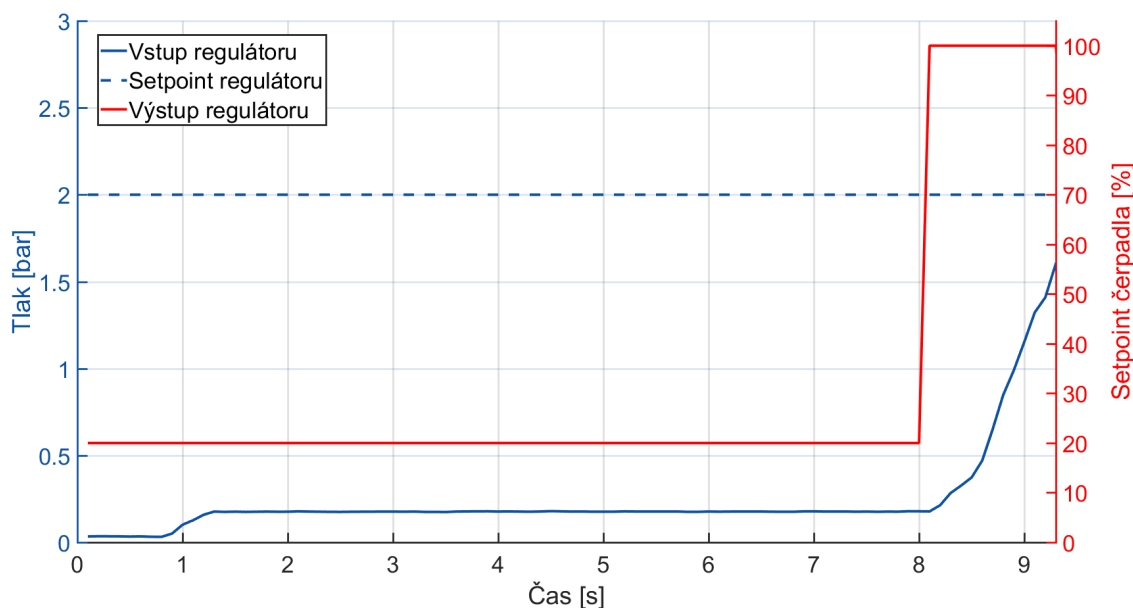
Poslední regulace, kterou bylo možné implementovat bez toho, aby poté nutné vypustit vodu z nádrží, je regulace tlaku v potrubí, kterým je spojena nádrž *Tank 1* s nádrží *Tank 2*. V této úloze bylo nutné využít k přiškrcování průtoku vody ventil,

který je umístěn za měřením tlaku (viz kapitola 1). Přiškrcení průtoku mělo za následek zvýšení tlaku. Sensor, který byl použit a jeho umístění na zařízení, je popsán v oddílu 1.3.2. I tato úloha má určitá omezení, jako je například maximální tlak, kterého je možné dosáhnout. Je nutné zajistit, aby potrubím stále protékalo minimální množství vody. Pokud by došlo k takovému zaškrcení průtoku, že nebude protékat voda potrubím a čerpadlo stále poběží na určitý výkon, došlo by k přehřátí čerpadla. Proto bylo nutné stanovit maximální hodnotu tlaku při které je ještě možné nechat zařízení zapnuté.

Obdobně jako v předchozích úlohách, musely být i při této regulaci nejdříve určeny parametry regulátoru. Byl opět použit stejný postup. Nejdříve byl spuštěn Pre-tuning a poté Fine tuning (viz podkapitola 3.4). U této úlohy byl také zvolen PID regulátor. Hodnoty tlaku se mohou pohybovat od 0 barů do 6 barů, což je maximální měřitelná hodnota tlaku daného senzoru.

7.3.1 Průběh Pre-tuningu

Jako první byly opět stanoveny parametry regulátoru z maximálního růstu regulovaného signálu, tj. vstupu regulátoru. Podmínka 3.2 uvedená v podkapitole 3.4 je splněna, pokud je požadovaná hodnota tlaku minimálně 2 bary. Po splnění této podmínky, bylo možné spustit algoritmus Pre-tuningu. Jeho průběh je vidět v grafu 7.10.

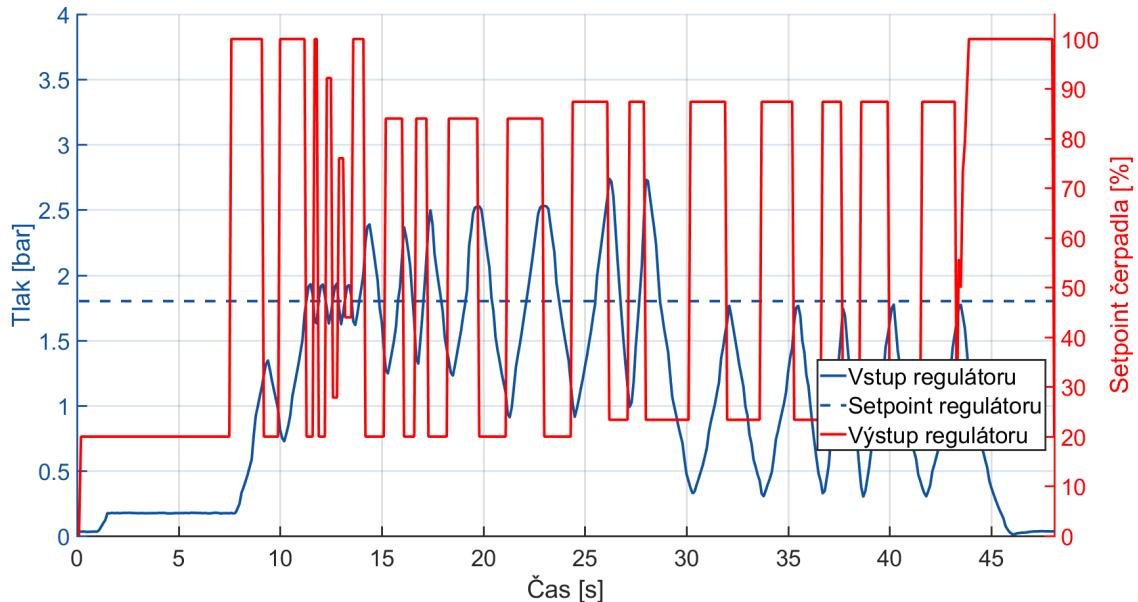


Obrázek 7.10: Průběh Pre-tuningu u úlohy regulace tlaku

Pre-tuning začíná v čase 0 s. V tento okamžik funkční blok PID_Compact počítá směrodatnou odchylku. Po skoku výstupu na 100% výkon v 8 vteřině, je měřen průběh vstupu regulátoru, tedy tlaku v potrubí, dokud není dosaženo inflexního bodu. Z této části jsou pak vypočteny koeficienty regulátoru, jejichž výsledné hodnoty jsou uvedeny v tabulce 7.3.

7.3.2 Průběh Fine tuningu

I u poslední regulační úlohy bylo opět přistoupeno k hledání parametrů pomocí implementovaného algoritmu Fine tuningu. Jak takový průběh vypadá u této úlohy je znázorněn v grafu 7.11.



Obrázek 7.11: Průběh Fine tuningu u úlohy regulace tlaku

Jelikož je nárůst tlaku velmi dynamický, začíná regulátor v tomto případě generovat oscilace už v 8 s. Stejně jako v předchozích případech, stanovil parametry (v tabulce 7.3), jenž nebyly příliš vyhovující, jak je ukázáno v oddílu 7.3.4. Proto bylo přistoupeno k manuální úpravě koeficientů z tohoto ladění.

7.3.3 Stanovené konstanty

Konstanty regulátoru tlaku určeny algoritmy Pre-tuning a Fine tuning byly doplněny o upravené parametry Fine tuningu. Hodnoty parametrů jsou v tabulce 7.3.

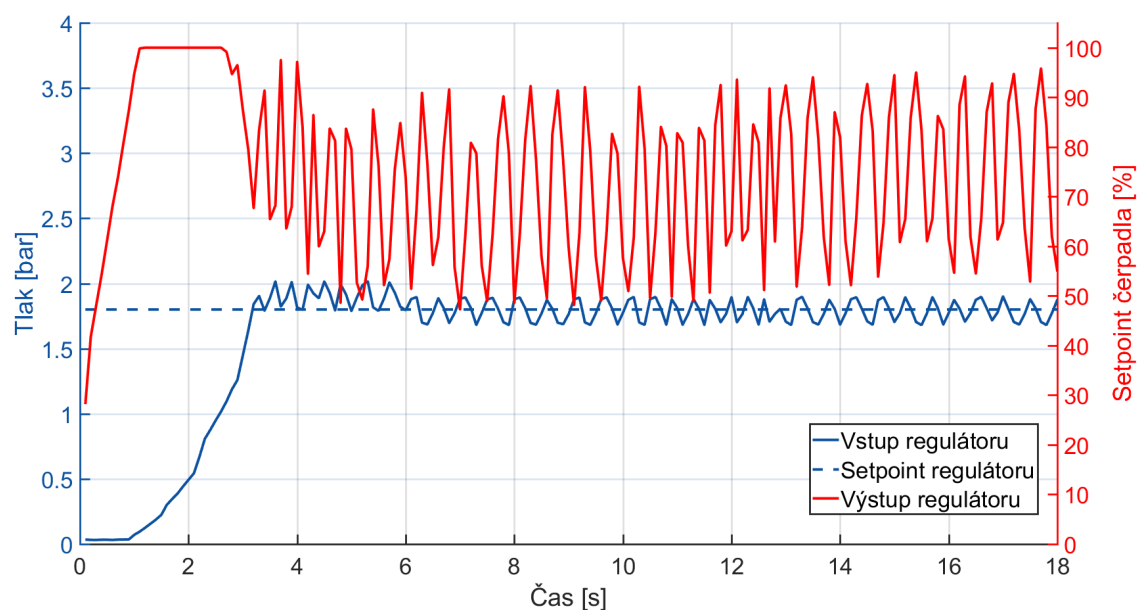
Symbol	Popis	Pre-tuning	Fine tuning	Upravené
K_p	Proporcionální zesílení	109.1	48.2	10.0
T_I	Integrační konstanta	2.3 s	1.3 s	1.3
T_D	Derivační konstanta	0.4 s	0.3 s	0.1 s
a	Koeficient derivačního zpoždění	0.1	0.1	0.0
b	Vážení proporcionální zásahu	0.8	0.3	0.0
c	Vážení derivačního zásahu	0.0	0.0	1.0

Tabulka 7.3: Koeficienty regulátoru tlaku

Z tabulky 7.3 lze vidět, že hodnota proporcionálního zesílení byla ve Fine tuningu snížena na polovinu. Upravené hodnoty byly sníženy ještě více, aby bylo zamezeno kmitání tlaku a výstupu regulátoru (viz následující oddíly). Integrovační a derivační časové konstanty byly modifikovány minimálně.

7.3.4 Regulace na skokovou změnu požadované hodnoty s parametry z Fine tuningu

Prvním ověřením parametrů regulátoru stanovených z Fine tuningu byla regulace na skokovou změnu požadované hodnoty tlaku. Průběh je vykreslen v grafu 7.12.



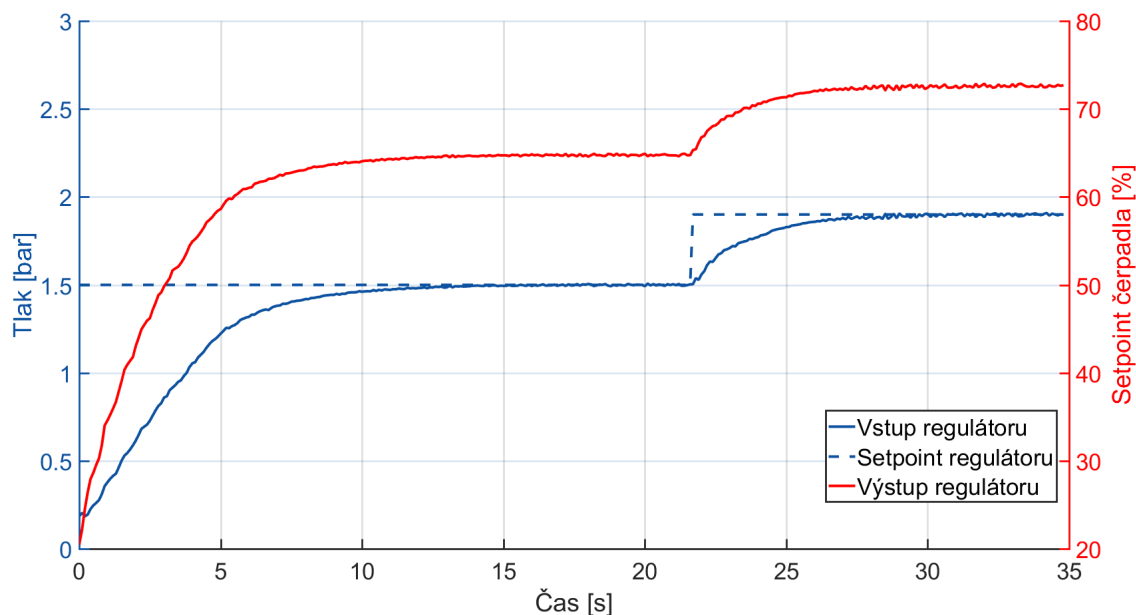
Obrázek 7.12: Průběh regulace se skokovou změnou požadovaného tlaku (parametry z Fine tuningu)

Nastavená požadovaná hodnota tlaku činila 1.8 bar. Lze vidět, že tlak se regulací zvýší na požadovanou hodnotu do 4 vteřin od zadání požadovaného tlaku. Avšak tlak nemá stabilní hodnotu, naopak se pohybuje v rozmezí 0.2 baru od požadované hodnoty. Výstup regulátoru, tedy výkon regulovaného čerpadla skáče v rozmezí téměř 45 % v krátkých časových úsecích, což může mít z dlouhodobého hlediska negativní vliv na funkčnost čerpadla. Tato regulace není příliš přesná, a proto byly hodnoty koeficientů tohoto regulátoru upraveny tak, aby byla regulace více „stabilní“.

7.3.5 Regulace na skokovou změnu požadované hodnoty s upravenými parametry z Fine tuningu

V předchozím oddílu bylo popsáno, jak parametry z Fine tuning algoritmu nebyly příliš vyhovující. Proto byly jejich hodnoty modifikovány tak, aby byla regulace

více vyhovující. Upravené parametry jsou uvedeny v tabule 7.3. Regulace po jejich nastavení je vykreslena grafem 7.13.



Obrázek 7.13: Průběh regulace se skokovou změnou požadovaného tlaku (upravené parametry z Fine tuningu)

Z grafu je zřejmé, že regulace na požadovanou hodnotu trvá o pár vteřin déle. Výsledek regulace je ovšem „stabilnější“. Odchylka tlaku v potrubí od požadované hodnoty tlaku je téměř nulová. Stejně tak průběh výstupu regulátoru a tím pádem i výkon čerpadla je daleko přijatelnější, jelikož jeho změna není skoková a při najetí na požadovanou hodnotu, se výkon mění řádově v desetinách procenta.

Kapitola 8

Závěr

Hlavním cílem této práce bylo navrhnout řídicí algoritmus fyzikálního modelu úpravy vody. V první fázi se bylo nutné seznámit s konstrukcí celého modelu a vymyslet využití jeho jednotlivých komponent. Následovalo zprovoznění jednotlivých frekvenčních motorů, aby bylo možné ovládat přítomná čerpadla. Z výkresu [21] bylo nutné zjistit, která komponenta konstrukce je připojena k PLC S7-1200 a S7-1500. Po otestování funkčnosti ovládání jednotlivých komponent bylo možné navrhnout funkční bloky, které by posloužily pro jejich individuální ovládání.

Před návrhem vlastních funkčních bloků byly popsány funkční bloky již implementované v prostředí TIA Portal. Toto prostředí umožnilo využít bloky pro předávání dat mezi dvěma S7 automaty. Dále byly k dispozici funkční bloky pro ovládání motorů pomocí frekvenčních měničů připojených přes průmyslovou síť a sériovou linku. V neposlední řadě bylo využito funkčního bloku pro PID regulaci, který má k dispozici dva implementované ladící algoritmy, jenž byly použity pro stanovení koeficientů regulátoru.

Pro ovládání komponent byly navrženy funkční bloky sloužící pro zpracování různých části zařízení. Jako první byl navržen funkční blok pro zpracování chyb, který zajistil vyhodnocení důležitých chybových digitálních signálů na vstupu každého PLC. Následoval návrh funkčního bloku pro zpracování analogového měření na vstupech PLC. Tento blok zajišťuje normování a škálování na měřený rozsah fyzikální veličiny daného senzoru. Pro čerpadla byl také navržen funkční blok. Tento blok zajišťuje možnost spuštění čerpadla v manuálním a automatickém režimu. Dále umožňuje nastavit požadovaný výkon čerpadla. Pro účely ovládání chyby regulace byl vytvořen generátor signálu s nastavitelným průběhem.

V další části práce byl popsán způsob předávání dat mezi PLC a HMI a poté způsob předávání dat z PLC na druhé PLC. Pro ovládání modelu byly vytvořeny kontrolní obrazovky umožňující spustit výukový model, nastavit jeho parametry a sledovat průběh regulačních úloh.

Všechny předchozí uvedené kroky musely být realizovány, aby mohlo být přikročeno ke stěžejní části této práce popsané na jejím konci, a tou jsou tři regulační úlohy, využívající přítomná měření hladiny, průtoku a tlaku. Jako akční členy byly zvoleny jednotlivá čerpadla ovlivňující hodnoty daných měření. U každé regulace bylo využito implementovaného funkčního bloku regulátoru a parametry regulátoru byly automaticky vypočteny dvěma ladícími algoritmy. Ověření funkčnosti jednotli-

vých úloh bylo provedeno příslušnými experimenty, jejichž výsledky jsou zobrazeny pomocí grafů. Výsledkem jsou funkční regulační úlohy s přijatelnou regulací na požadovanou hodnotou.

Literatura

- [1] AVNET.COM. Piezoresistive pressure sensors. <https://www.avnet.com/wps/portal/abacus/solutions/technologies/sensors/pressure-sensors/core-technologies/piezoresistive-strain-gauge/>, [cit. 18-7-2020].
- [2] BIOCHEMICKÝ ÚSTAV, LÉKAŘSKÁ FAKULTA, M. U. Přehled základních pojmů z obecné chemie. <https://is.muni.cz/el/med/podzim2011/BSBC011p/30547295/kyseliny.html>, 2011- [cit. 12-7-2020].
- [3] CHROMSERVIS. Průvodce měřením ph v laboratořích. https://www.chromservis.eu/images/pdf/chromservis_pruvodce-mereni_CZ.pdf, 2017- [cit. 12-7-2020].
- [4] GRUNDFOS. Cm 1-5 a-r-i-e-aqge f-a-a-n. <https://product-selection.grundfos.com/product-detail.product-detail.html?custid=GMA&productnumber=96807014&qcid=1020215021,,> 2020- [cit. 12-7-2020].
- [5] INSTRUMENTATIONTOOLS.COM. Two electrode conductivity probes principle. <https://instrumentationtools.com/two-electrode-conductivity-probes-principle/>, [cit. 20-7-2020].
- [6] MELICHAR, J. Lineární systémy 1. http://www.kky.zcu.cz/uploads/courses/ls1/LS1_Ucebni_texty_2011.pdf, 2011 - [cit. 5-8-2020].
- [7] PROMINENT. Conductivity sensor – lf (t) product range. <https://www.prominent.com/resources/OperatingInstructions/English/12408/985128-BA-DT-043-10-13-EN-LFT-K1-Leitfaehigkeitssensor-EN.pdf>, 2013- [cit. 16-7-2020].
- [8] PROMINENT. Data sheet ph-combination probes pher 112 se. <https://www.prominent.com/resources/DataSheet/English/618/TDS-05-98-PHER-112-SE-D-GB-F-E.pdf>, 2013- [cit. 18-7-2020].
- [9] PROMINENT. Flyer - dosing tanks. <https://www.prominent.com/en/Products/Products/Tanks/p-dosing-tanks.html>, 2020- [cit. 10-7-2020].
- [10] SIEMENS. *MANUAL SINAMICS G120C*, 2014- [cit. 16-7-2020]. https://cache.industry.siemens.com/dl/files/780/99683780/att_863315/v1/G120C_List_Manual_LH13_0414_eng.pdf.

- [11] SIEMENS. Simatic step 7 basic/professional v15.1 and simatic wincc v15.1. <https://support.industry.siemens.com/cs/document/109755202/simatic-step-7-basic-professional-v15-1-and-simatic-wincc-v15-1?dti=0&lc=en-B0>, 2018- [cit. 15-7-2020].
- [12] SIEMENS. Pressure transmitter sitrans p220 (7mf1567). [https://support.industry.siemens.com/cs/document/50489739/pressure-transmitter-sitrans-p220-\(7mf1567\)?dti=0&lc=en-WW](https://support.industry.siemens.com/cs/document/50489739/pressure-transmitter-sitrans-p220-(7mf1567)?dti=0&lc=en-WW), 2019- [cit. 12-7-2020].
- [13] SIEMENS. Sitrans fm mag 1100/1100 f. <https://support.industry.siemens.com/cs/document/109778048/sitrans-fm-mag-1100-1100-f?dti=0&lc=en-WW>, 2019- [cit. 12-7-2020].
- [14] SIEMENS. 6av2123-2gb03-0ax0. <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6AV2123-2GB03-0AX0>, [cit. 25-7-2020]. Katalogový produkt.
- [15] SIEMENS. 6av2124-0gc01-0ax0. <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6AV2124-0GC01-0AX0>, [cit. 25-7-2020]. Katalogový produkt.
- [16] SIEMENS. 6es7214-1ag40-0xb0. <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7214-1AG40-0XB0>, [cit. 25-7-2020].
- [17] SIEMENS. 6es7512-1ck01-0ab0. <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7512-1CK01-0AB0>, [cit. 25-7-2020]. Katalogový produkt.
- [18] SIEMENS. 6sl3210-1ke12-3ap2. <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6SL3210-1KE12-3AP2>, [cit. 25-7-2020]. Katalogový produkt.
- [19] TIŠL, J. *Panel - Universita*. ProMinent Systems spol. s r.o., 2017. Konstrukční výskres.
- [20] VOJÁČEK, A. Zajímavé principy měření - elektromagnetické (indukční) průtokoměry. <https://automatizace.hw.cz/zajimave-principy-mereni-elektromagneticke-indukcni-prutokomery>, 2009- [cit. 12-7-2020].
- [21] WOPRŠÁLEK, J. *Automation workplace*. ProMinent Systems spol. s r.o., 2017. Výskres - 6017800940-0010.