

**Semestrální *projekt Prostředky průmyslové
automatizace 2022/2023*
letní semestr**

**PLC S7 1500 – Realizace řízení PID
regulátoru**

BAR0469
Alexander Baršč

Obsah

1	Zadání projektu a cíl řešení.....	2
2	Počáteční implementace v TIA Portal.....	2
2.1	Hardwarová konfigurace.....	2
2.2	Změření přechodové charakteristiky.....	2
3	Identifikace v Matlabu.....	4
4	Implementace v TIA Portal	5
4.1	PID regulátor s konstantami z Matlabu	6
4.2	Realizace PID regulátoru pomocí auto-tunningu v TIA Portal	8
5	Realizace HMI rozhraní.....	9
6	Závěr	10

Obrázek 1.	Hardwarová konfigurace pro daný projekt	2
Obrázek 2.	Přenos soustavy.....	2
Obrázek 3.	Implementace soustavy v TIA PORTAL V17	3
Obrázek 4.	Spuštění jednotkového skoku	3
Obrázek 5.	Změřená přechodová charakteristika	4
Obrázek 6.	Porovnání naměřených dat a identifikované soustavy	4
Obrázek 7.	Získané konstanty z PID Tune.....	5
Obrázek 8.	Realizace programu v TIA PORTAL	5
Obrázek 9.	Zadání konstant pro PID regulátor z Matlabu	6
Obrázek 10.	Graf řízení s PID Regulátorem naleděným v Matlabu (Červeně je akční zásah, černě požadovaná hodnota, zeleně výstup soustavy)	6
Obrázek 11.	Konstanty PID Regulátoru z Matlabu po úpravě.....	7
Obrázek 12.	PID Regulátor z Matlabu po úpravě konstant (Červeně je akční zásah, černě požadovaná hodnota, zeleně výstup soustavy)	7
Obrázek 13.	Konstanty získané z autotuningu pro PID Regulátor.....	8
Obrázek 14.	PID Regulátor navržený dle autotuningu v TIA Portalu (Červeně je akční zásah, černě požadovaná hodnota, zeleně výstup soustavy)	8
Obrázek 15.	Porovnání výstupů ze soustav	9
Obrázek 16.	Porovnání výstupů z PID regulátorů.....	10

1 Zadání projektu a cíl řešení

Realizujte řízení s S7 1500 pomocí PID regulátoru (PID Compact) pro vybranou soustavu z knihovny „LSim” controlled system library (viz. odkazy níže).

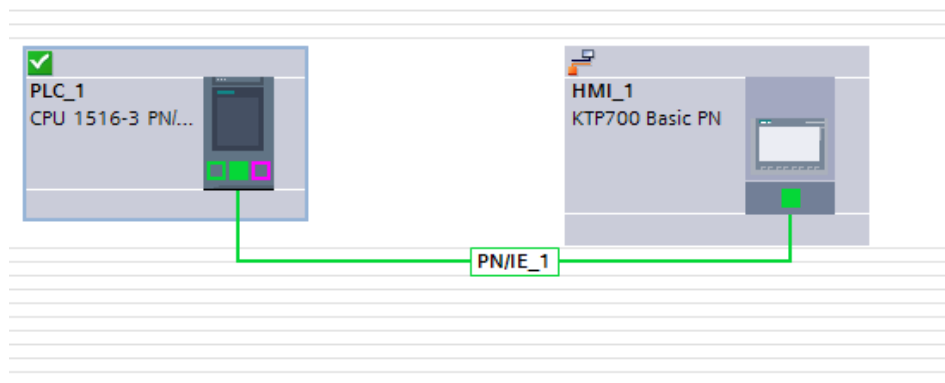
Změřte přechodovou charakteristiku a proveďte export hodnot do souboru. Soubor použijte jako vstupní hodnoty do Matlabu. Proveďte identifikaci soustavy v Matlabu a návrh konstant regulátoru. Porovnejte nastavení konstant s nastavením pomocí selftuningu v PID Compact.

Zvolená soustava LSim_PT2aper

2 Počáteční implementace v TIA Portal

2.1 Hardwarová konfigurace

Vzhledem k tomu, že daný regulátor a soustava je pouze realizována jako interní proces ve zvoleném PLC, tak není nutné připojovat k PLC další externí karty, ale stačí nám samotné PLC s dodatečným HMI panelem.



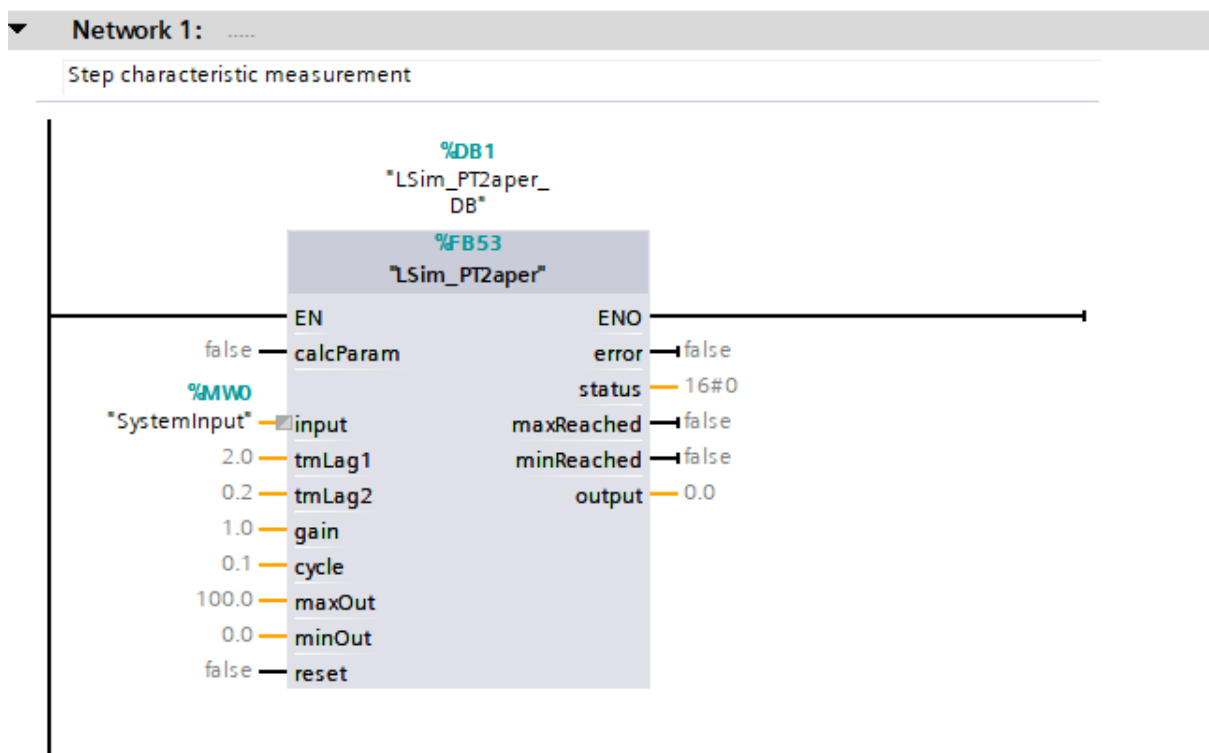
OBRÁZEK 1. HARDWAROVÁ KONFIGURACE PRO DANÝ PROJEKT

2.2 Změření přechodové charakteristiky

Nejdříve je nutné do prostředí TIA Portal importovat knihovnu „LSIM“, poté musíme do programových bloků přidat funkční blok „LSim_PT2aper“, jenž pak můžeme použít v našem programu. Tato soustava má následující přenos:

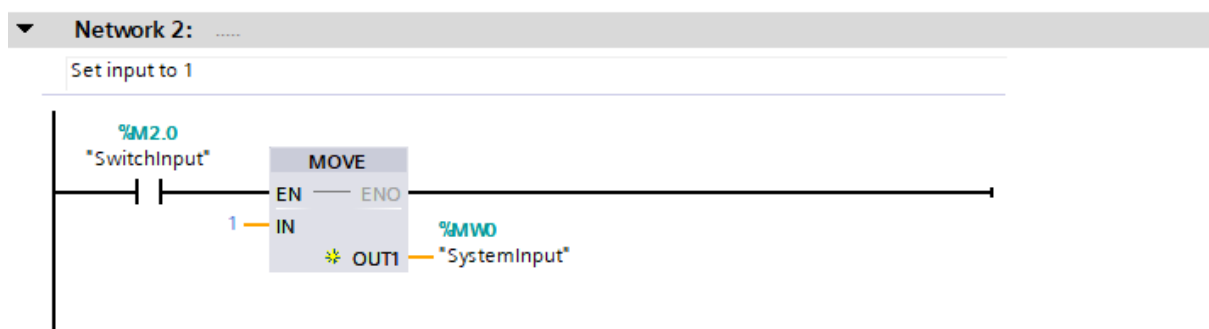
$$F(p) = \frac{gain}{(tmLag1 * p + 1) * (tmLag2 * p + 1)}$$

OBRÁZEK 2. PŘENOS SOUSTAVY



OBRÁZEK 3. IMPLEMENTACE SOUSTAVY V TIA PORTAL V17

V záložce „Traces“ můžeme realizovat měření, jenž se spustí na vzestupné hraně bitu „SwitchInput“, který na vstup pustí jednotkový skok. Po spuštění měření se vzorky odebírají při spuštění periodického přerušení, jenž se spouští každých 250 us.



OBRÁZEK 4. SPUŠTĚNÍ JEDNOTKOVÉHO SKOKU

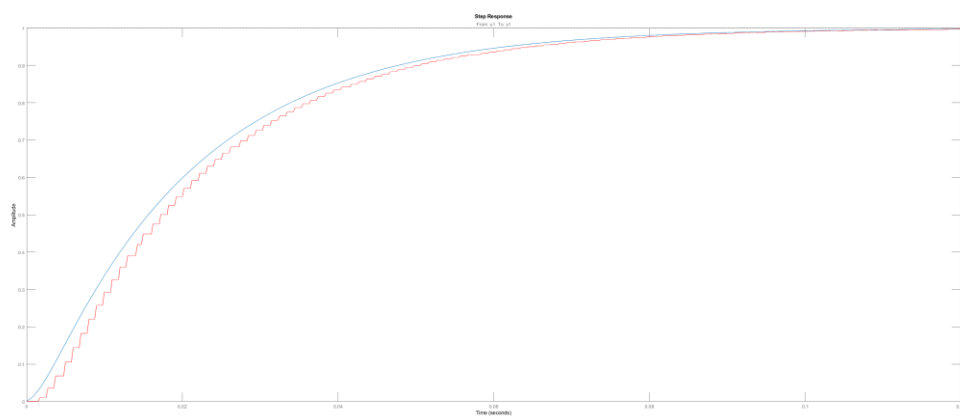


OBRÁZEK 5. ZMĚŘENÁ PŘECHODOVÁ CHARAKTERISTIKA

3 Identifikace v Matlabu

V Matlabu jsem realizoval jednoduchý skript, kde provedu import z naměřených csv dat, ty poté vložíím do nástroje „Ident“ a pomocí něho provedu identifikaci soustavy, tato identifikace mi vrátí přenos, pro který už pomocí zabudovaných funkcí mohu snadno navrhnout PID regulátor s vhodnými konstantami.

Na následujícím grafu se nachází porovnání přechodové charakteristiky identifikované soustavy (označena modře) a naměřená data z TIA Portalu (označeny červeně).



OBRÁZEK 6. POROVNÁNÍ NAMĚŘENÝCH DAT A IDENTIFIKOVANÉ SOUSTAVY

Po vykonání skriptu vyšly následující parametry PID regulátoru:

```
Grl =

      1      1
      Kp * (1 + ---- * ---- + Td * s)
                Ti      s

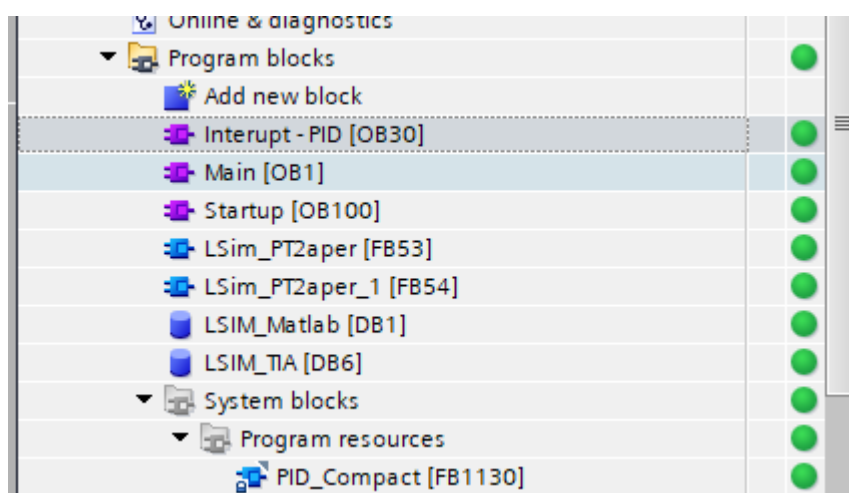
with Kp = 1.61, Ti = 0.0124, Td = 0.00176

Continuous-time PID controller in standard form
```

OBRÁZEK 7. ZÍSKANÉ KONSTANTY Z PID TUNE

4 Implementace v TIA Portal

V projektu TIA Portal jsem implementoval přerušení s názvem Interrupt – PID. Toto přerušení je vykonáno co 100 ms, a dochází uvnitř jeho obsluhy k realizaci řízení soustavy. Také jsem realizoval PID regulátor pomocí automatického ladění v TIA Portal.



OBRÁZEK 8. REALIZACE PROGRAMU V TIA PORTAL

Uvnitř HMI panelu jsem implementoval jednoduchou vizualizaci, kde je možné porovnat oba regulátory v akci. Kdykoliv můžeme změnit požadovanou hodnotu neboli „Set Value“.

4.1 PID regulátor s konstantami z Matlabu

☒ Enable manual entry

Proportional gain: 1.61

Integral action time: 0.0124 s

Derivative action time: 0.00176 s

Derivative delay coefficient: 0.2

Proportional action weighting: 1.0

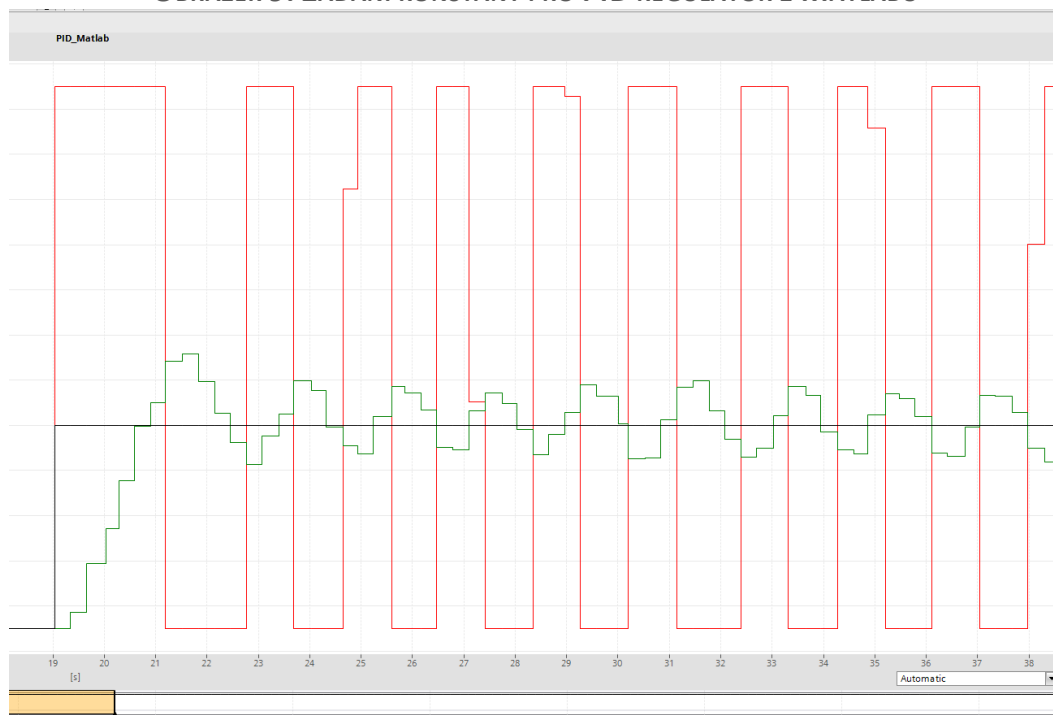
Derivative action weighting: 1.0

Sampling time of PID algorithm: 0.1 s

Tuning rule

Controller structure: PID

OBRÁZEK 9. ZADÁNÍ KONSTANT PRO PID REGULÁTOR Z MATLABU



OBRÁZEK 10. GRAF ŘÍZENÍ S PID REGULÁTOREM NALEDĚNÝM V MATLABU (ČERVENĚ JE AKČNÍ ZÁSAH, ČERNĚ POŽADOVANÁ HODNOTA, ZELENĚ VÝSTUP SOUSTAVY)

Po následném otestování v TIA Portal jsem ale došel k zjištění, že tento PID regulátor není pro regulaci moc použitelný, protože jeho akční zásah má kmitavý průběh a výstup se nikdy neustálí na požadovanou hodnotu. Proto jsem začal manuálně ladit konstanty PID regulátoru, dobrý výsledek jsem získal po změně konstanty I na hodnotu 0,5.

☒ Enable manual entry

Proportional gain: 1.61

Integral action time: 0.5 s

Derivative action time: 0.00176 s

Derivative delay coefficient: 0.2

Proportional action weighting: 1.0

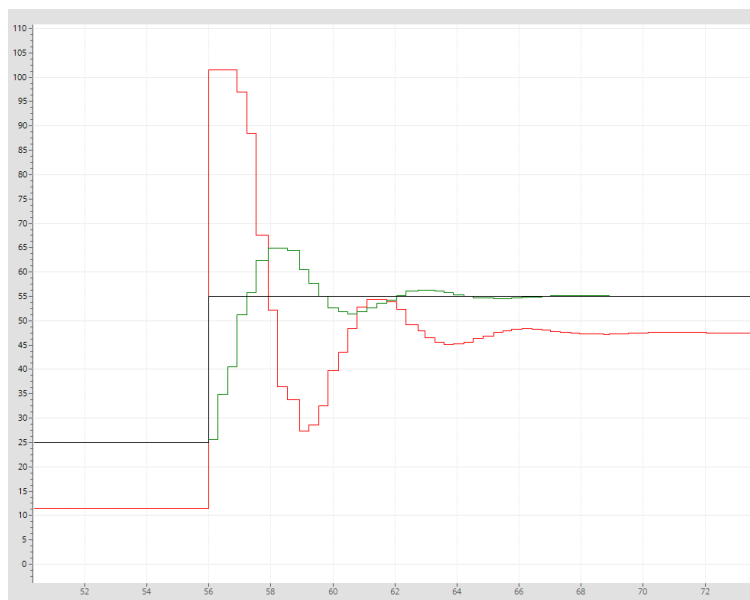
Derivative action weighting: 1.0

Sampling time of PID algorithm: 0.1 s

Tuning rule

Controller structure: PID

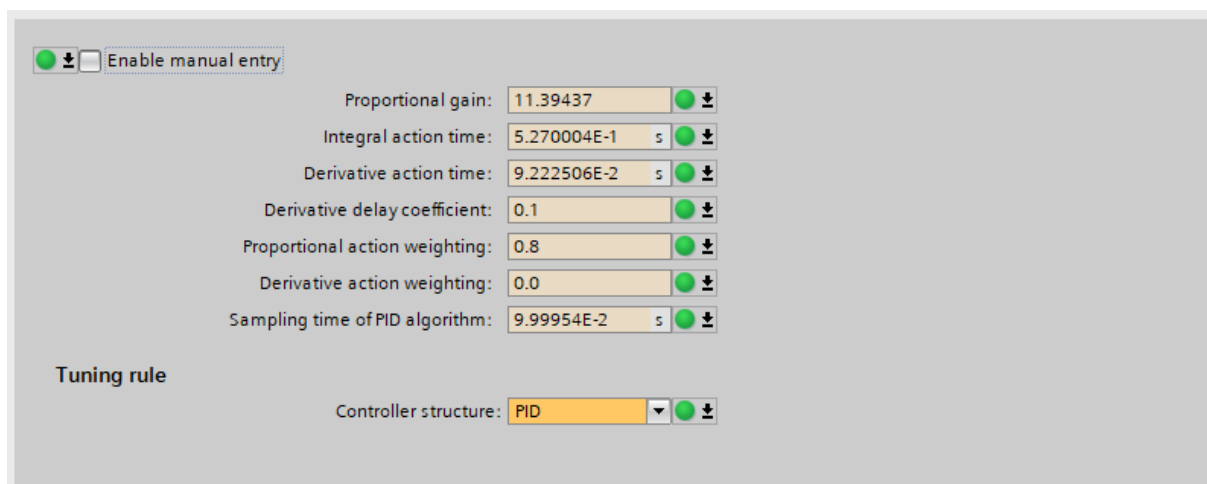
OBRÁZEK 11. KONSTANTY PID REGULÁTORU Z MATLABU PO ÚPRAVĚ



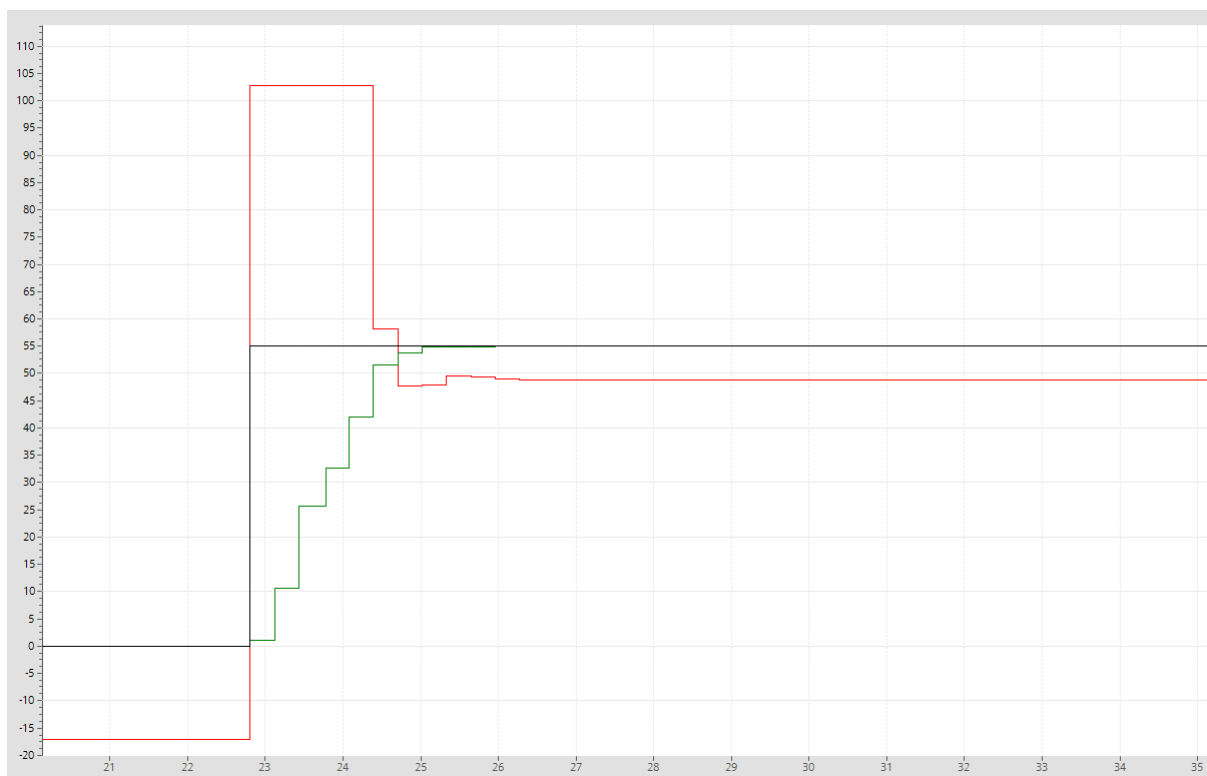
OBRÁZEK 12. PID REGULÁTOR Z MATLABU PO ÚPRAVĚ KONSTANT (ČERVENĚ JE AKČNÍ ZÁSAH, ČERNĚ POŽADOVANÁ HODNOTA, ZELENĚ VÝSTUP SOUSTAVY)

4.2 Realizace PID regulátoru pomocí auto-tunningu v TIA Portal

Další možností, jak realizovat PID regulátor, je využít v TIA Portalu funkci auto-tunningu. V rámci této funkcionality TIA Portalu na soustavu přivede jednotkový skok a pomocí vlastní identifikace zjistí, jaké správné konstanty zvolit.



OBRÁZEK 13. KONSTANTY ZÍSKANÉ Z AUTOTUNINGU PRO PID REGULÁTOR



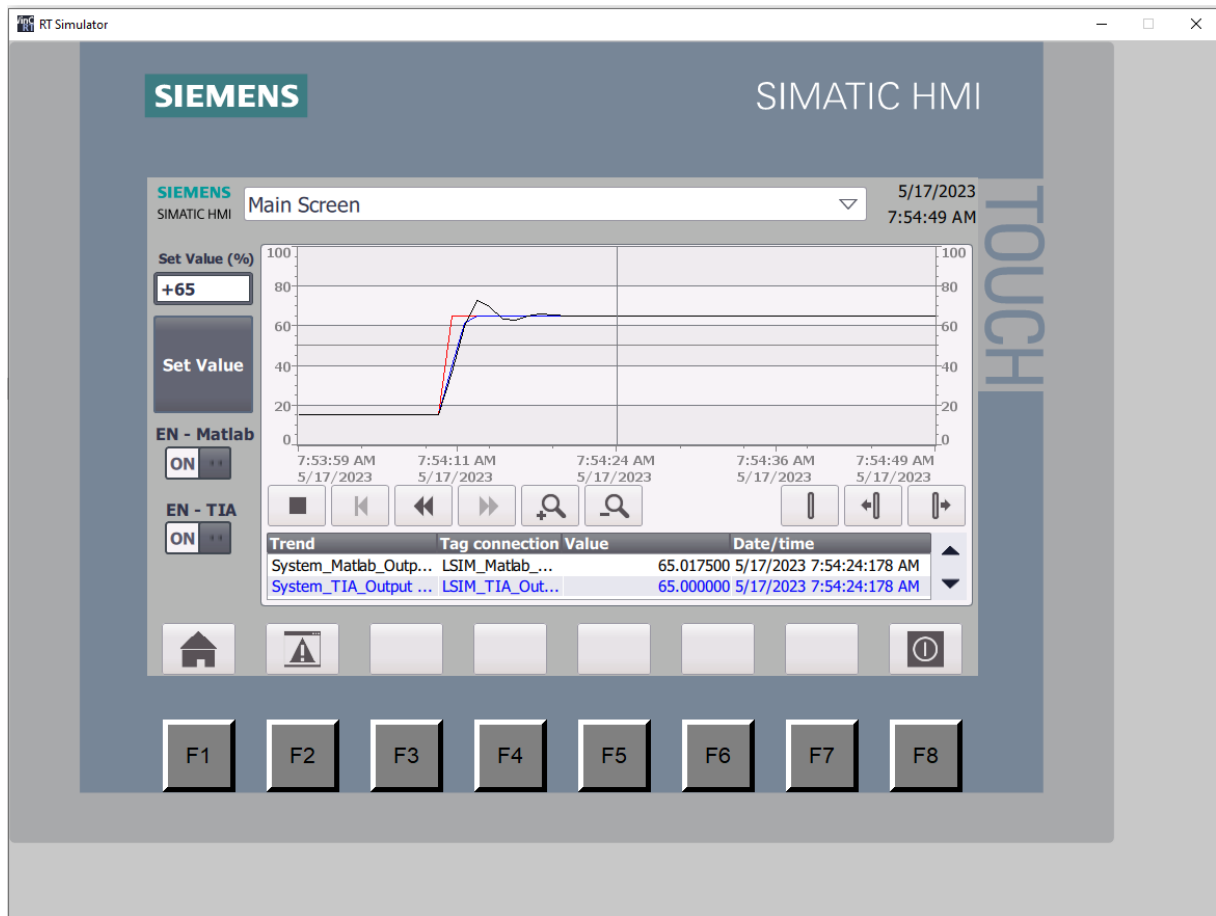
OBRÁZEK 14. PID REGULÁTOR NAVRŽENÝ DLE AUTOTUNINGU V TIA PORTALU (ČERVENĚ JE AKČNÍ ZÁSAH, ČERNĚ POŽADOVANÁ HODNOTA, ZELENĚ VÝSTUP SOUSTAVY)

Realizovaný PID regulátor jsem poté přidal dovnitř obsluhy přerušení OB30 pro řízení soustavy se stejnými parametry jak u PID regulátoru navrženého v Matlabu, aby se dali navzájem porovnat.

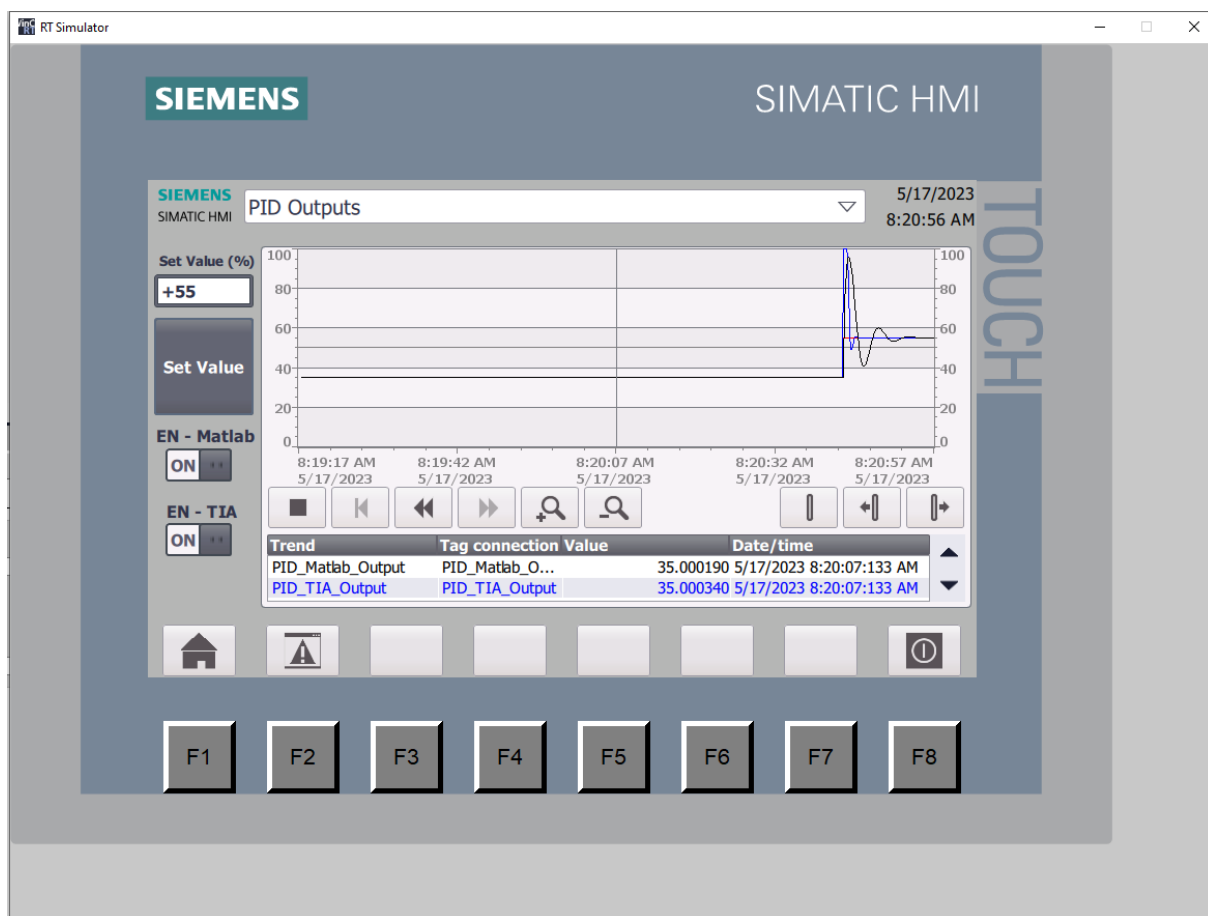
5 Realizace HMI rozhraní

Pro uživatelsky příjemné ovládání jsem realizoval rozhraní na HMI panelu, které má dvě obrazovky, na prvním lze porovnat výstup ze řízených soustav a na druhé lze porovnat výstup z PID regulátorů.

Uživatel může libovolně nastavit žádanou hodnotu a dle potřeby může zapnout/vypnout individuální regulátory se soustavou.



OBRÁZEK 15. POROVNÁNÍ VÝSTUPŮ ZE SOUSTAV



OBRÁZEK 16. POROVNÁNÍ VÝSTUPŮ Z PID REGULÁTORŮ

6 Závěr

Cílem projektu bylo realizovat PID regulátor pro simulovanou soustavu z knihoven pro TIA Portal. Pro soustavu jsem naměřil přechodovou charakteristiku, díky které jsem pak provedl identifikaci v System Identification Toolboxu v Matlabu. Identifikovanou soustavu jsem porovnal s naměřenými daty, pro kontrolu správnosti.

Pomocí funkce pidtune v Matlabu jsem provedl návrh konstant PID regulátoru ve standardním tvaru. Tento regulátor jsem pak realizoval v samotné aplikaci v TIA Portalu a otestoval ho, nicméně jeho výstup byl příliš kmitavý a tedy nedostačující. Po úpravě integrační konstanty jsem se dostal k použitelnému výsledku. Tato metoda ale nazahrnuje další parametry co je nutné v TIA Portalu nastavit. Poté jsem realizoval PID regulátor pomocí auto-tuningu přímo v rozhraní v TIA Portalu. Z porovnaných dat je zřejmé že tento regulátor má lepší vlastnosti a proto je pro návrh regulátoru PLC lepší použít tuto variantu.