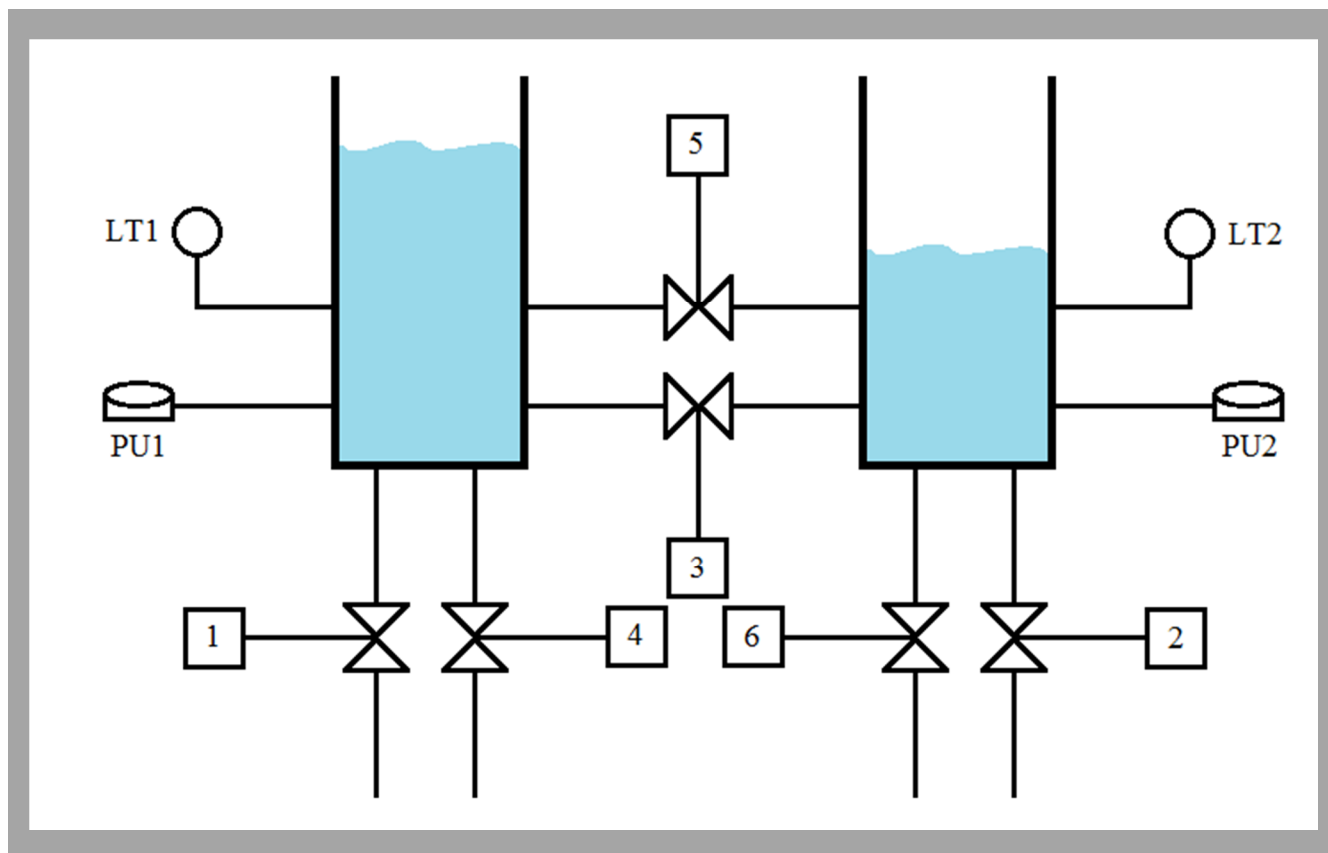




Reglering av tanksystem med PLC

Programutveckling i CoDeSys-miljö

Examensarbete inom högskoleingenjörsprogrammet Mekatronik

OLIVER LARSSON
ADAM SUNAL

FÖRORD

Examensarbetet på 15HP har utförts på Institutionen för Signaler och System på Chalmers tekniska högskola, och är den avslutande kursen för 3 års högskoleingenjörsutbildning för att bli högskoleingenjör i mekatronik. Arbetet har utförts under 7 veckor på Chalmers tekniska högskola, campus Lindholmen under slutet av våren och början på sommaren. Examinator för arbetet är Bertil Thomas och handledare är Veronica Olesen.

Vi vill tacka:

Morgan Osbeck för omfattande hjälp och engagemang i de flesta av arbetets områden, samt för att ha gett idén till examensarbetet. Göran Hult för hjälp med hårdvaru-relaterade problem samt regleringsproblem. Veronica Olesen för att ha bidragit med kunskap, feedback och utmanande uppgifter som har utvecklat examensarbetet, samt oss själva som ingenjörer. Bertil Thomas för en mycket bra skriven bok som har hjälpt mycket under arbetets gång, samt för tillåtelsen att använda bilder ur boken.

Göteborg, 2016-08

Adam Sunal & Oliver Larsson

SAMMANFATTNING

Projektet har genomförts med syftet att testa och verifiera styrningen av en tankprocess med hjälp av en PLC som har programmerats i CoDeSys miljö. Dessutom har olika reglerprinciper tagits fram med hjälp av MATLAB och implementerats i CoDeSys.

Tester har sedan utförts för att kontrollera om systemet lever upp till de krav som ställs. Testerna har varit automatiserade för att eliminera inverkan av den mänskliga faktorn. Styrning av ett antal magnetventiler och lampor har utförts parallellt för att visa på möjligheterna med användandet av en PLC. Ett antal dokument har också tagits fram för att underlätta användning av systemet för framtida användare, dessutom har ett antal förslag för användandet av systemet i en laboration tagits fram.

Systemet är uppbyggt av ett tanksystem med två tankar och ett antal ventiler, samt en Crevis NA-9372 programmerbart I/O med ett antal tilläggskort.

Projektet har resulterat i ett välfungerande styrsystem med ett antal olika sidofunktioner tillgängliga, samt ett antal dokument och en rapport som beskriver arbetet under projektet och ger förslag inför framtida användande i form av laborationer.

ABSTRACT

This project has been conducted with the purpose of testing if it is possible to control a tank-process with a PLC operating in CoDeSys environment. To test the system, different control algorithms were tested in MATLAB and then finally implemented in CoDeSys.

Different tests have been conducted on the system to verify that it lives up to the standard that a control system should. The tests have been automated, to eliminate the human factor.

A number of side functions, such as lamps and solenoid valves have been controlled simultaneously to prove the possibilities with a PLC acting as a process regulator. A number of support documents have also been created to support future users. These documents contain information about possible problems and configuration, as well as tips on how the system can be used in a laboratory setup.

The result was very satisfying, the system worked very well, a fast and smooth control of the process was achieved. The conclusions drawn from this project is that a CoDeSys PLC can be used to control a tank process. The system also has a very big potential for future use in education setups, mainly because of the combination of process control and control of digital outputs.

INNEHÅLLSFÖRTECKNING

1. INLEDNING	1
1.1 Bakgrund	1
1.2 Syfte	1
1.3 Avgränsningar	1
1.4 Mål	1
2. TEORI	2
2.1 Tanksystem	2
2.2 Första ordningens system	2
2.3 Andra ordningens system	3
2.4 Dödtid och förstärkning	3
2.5 Lambda-metoden	3
2.6 Kaskadreglering	4
2.7 Framkoppling	4
2.8 Derivatafilter	5
2.9 Anti Wind-Up	5
2.10 Digitala och analoga signaler	6
2.11 PLC-moduler	6
3. METOD	8
3.1 Stegvarsanalys	8
3.2 Dimensionering av regulatorer	8
3.3 Tester och dokumentation	8
4. UTFÖRANDE	9
4.1 Implementering i PLC	9
4.2 Enkeltank överföringsfunktion	11
4.3 Dubbeltank överföringsfunktion	12
4.4 Enkeltanksreglering med störningar och dödtid	13
4.5 Dubbeltanksreglering med störningar och dödtid	16
4.6 Kaskadreglering med störningar och dödtid	21
4.7 Dynamisk störningslogik med dödtid	26
4.8 Stabilitetsmarginaler för de olika regulatorerna	31
5. RESULTAT OCH DISKUSSION	33
REFERENSER	35
BILAGOR	37
A. Kod i CoDeSys	37
B. Konfigurering	45
B.1 Konfigurering av PLC och CoDeSys	45
B.2 Lägg till bibliotek i CoDeSys	45
B.3 Infoga block från OSCAT Basic (FBD)	45
C. Problem som kan uppstå	46
D. Förslag inför labb	47

BEGREPP

PLC - Programmable Logic Controller, ett programmerbart styrsystem som ofta används inom industrin.

PID-regulator - Proportionell Integrerande Deriverande regulator en väldigt vanlig regulator inom reglertekniken.

K - En beteckning av förstärkning inom reglerteknik. Används både för att beteckna en process förstärkning samt en regulators förstärkning.

T_I - En beteckning av integraltiden i en PID-regulator. Det vill säga hur stort bidrag till utsignalen den integrerande delen ska ge.

T_D - En beteckning av derivatatiden Det vill säga hur stort bidrag till utsignalen den deriverande delen ska ge.

CoDeSys - En utvecklingsmiljö för programmerbara styrenheter.

OSCAT - Open Source Community for Automation Technology, En sammanslutning där användare delar med sig av och hämtar olika funktioner som är användbara inom automationsteknologi.

1. INLEDNING

1.1 Bakgrund

I kurser i reglerteknik på Chalmers, campus Lindholmen används idag laborationer på tanksystem, där nivån i en tank styrs med hjälp av en inbyggd regulator eller manuellt. I samband med utvecklingen av en ny kurs i styr- och reglerteknik önskar man se över möjligheten att utveckla en ny laboration där tanksystemet styrs med hjälp av en PLC i CoDeSys-miljö.

1.2 Syfte

Arbetet går ut på att testa hur bra det nya styrsystemet fungerar för tanksystemet, samt hur användarvänligt det är för framtida användare. Utöver enkla regleringar av enkel- och dubbeltank, skall även mer avancerade reglerprinciper testas för att se hur det nya systemet hanterar dessa, samt hur både de enkla och de avancerade regleringarna hanterar störningar och liknande. Konfigurering och problem som uppstår skall även dokumenteras, för att förbereda framtida användare.

1.3 Avgränsningar

De avgränsningar som införts är framförallt en tidsbegränsning, arbetet får max ta 10 veckor heltidsarbete. Vissa begränsningar i utrustningen existerar också, bland annat finns endast en fungerande pump att tillgå per tanksystem. Ekonomin är även den begränsad, eftersom arbetet är ett test för framtiden. Antalet alternativa reglerprinciper är även de begränsade, eftersom de skall vara användbara i en grundkurs.

1.4 Mål

Huvudmål

- Utvärdera olika reglerprinciper som används på tankarna med PLC.
- Förbereda användandet av en ny laboration för framtida användare.

Detalj mål

- Testa och dokumentera reglering av enkeltank.
- Testa och dokumentera programmering av PID-regulator i CoDeSys, både i strukturerad text och med ett färdigt block.
- Testa och dokumentera reglering av dubbeltank.
- Testa och dokumentera användandet av kaskadreglering.
- Testa och dokumentera användandet av störningslogik, exempelvis framkoppling.
- Testa de olika reglerprinciperna hur bra de klarar av död tid.
- Dokumentera konfiguration av systemet.
- Dokumentera eventuella problem som uppstår i systemet.
- Ta fram eller ge förslag på utformningen av en laboration för framtida elever.

2. TEORI

2.1 Tanksystem

Systemet består av två tankar som är ihopkopplade med hjälp av två ventiler som sitter längst ner mellan tankarna. (se figur 1). En pump är kopplad till tank nr 1 (längst till vänster). En annan pump finns också kopplad till tank nummer två, dock används ej denna pump.

Tre stycken magnetventiler är också kopplade till tankarna, två på varje tanks undersida samt en mellan dem. De övriga tre ventilerna i systemet är vanliga kulventiler, där en är kopplad mellan tankarna och övriga två på tankarnas undersidor. Fjärrstyrning av magnetventilerna är möjlig via ett antal ingångar som sitter på switchlådan.

Två nivågivare finns också att tillgå, en per tank. Två återkopplade tuber är kopplade från tankarnas översida ner till vattenreservoaren, detta för att motverka översvämning. En box för datorstyrning finns tillgänglig för att plocka ut data och skriva styrvärden till pumpen. Systemet är också säkrat 6,3A för att skydda pumparna och resten av utrustningen.



Figur 1. Tanksystemet som använts.

2.2 Första ordningens system

Ett system av första ordningen, är ett system vars överföringsfunktion kan beskrivas med hjälp av en tidskonstant. Processer av denna typ har ett stegsvar som har högst lutning i början, vilken sedan avtar för att till slut stabilisera sig vid en slutnivå. Tidskonstanten för en första ordningens process är den tid det tar systemets stegsvar att nå 63% av slutnivå. Det finns många system som kan beskrivas med en tidskonstant, ett exempel på detta är en enkel nivåprocess.

$$G(s) = \frac{K}{1+Ts} \quad (2.1)$$

En första ordningens överföringsfunktion beskrivs i ekvation (2.1).[1]

2.3 Andra ordningens system

Ett system av andra ordningen med reella poler är ett system vars överföringsfunktion kan beskrivas med hjälp av två tidskonstanter. Stegsvaret för en process med två tidskonstanter kryper försiktigt upp i början för att öka likt ett första ordningens system för att slutligen stabilisera sig vid en slutnivå. Tidskonstanterna kan bestämas med hjälp av den tid det tar processen att nå 2/3 respektive 1/3 av slutnivå. De värden som då fås behandlas och slutligen bestäms tidskonstanterna med hjälp av diagram.[1]

$$G(s) = \frac{K}{(1+T_1s)(1+T_2s)} \quad (2.2)$$

En andra ordningens överföringsfunktion med reella poler beskrivs i ekvation (2.2).

Exempel på en process med en andra ordningens överföringsfunktion är två seriekopplade tankar. Om den andra tidskonstanten, T_2 är mycket mindre än den första tidskonstanten T_1 kan systemet ibland approximeras med ett system med endast en tidskonstant utan att modellen är för långt från verkligheten för att vara användbar.

2.4 Dödtid och förstärkning

Dödtid är den tid det tar innan en ändring på insignalen överhuvudtaget märks på utsignalen. Oftast är det olika typer av transportsträckor i systemet som ger upphov till dödtiden, dvs. tar det en viss tid för mediet att röra sig runt i systemet. Om processen som skall regleras är mycket snabb måste även den tidsdiskreta regulatorns cykeltid och omvandlingstider räknas som en form av dödtid.[1]

Med en process förstärkning menas hur mycket processens utsignal ändras då insignalen ändras med en enhet. Förstärkningen betecknas K . Förstärkningen bestäms ur stegsvaret hos en process. Det är viktigt att komma ihåg att om höjden av insignalen till processen som ger stegsvaret inte är lika med 1, så måste slutnivån på stegsvaret divideras med höjden av insignalen för att erhålla korrekt förstärkning. [1]

2.5 Lambda-metoden

Lambda-metoden är ett sätt att ställa in parametrarna i regulatorer. Fördelen med lambda-metoden är att när man använder den metoden för regulatortrimning, så ger den väldigt bra kontroll över hur snabb reglerkretsen blir i slutändan och eftersom man oftast vill att regleringen av en process ska ske så fort och stabilt som möjligt, så är lambda-metoden ett bra val i det fallet.

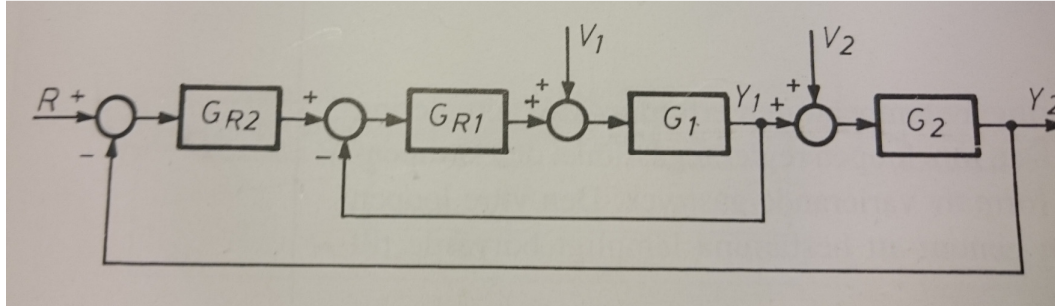
För att reglera med lambda-metoden krävs det ett stegsvar för processen. Med hjälp av stegsvaret kan dödtiden, processförstärkningen, samt tidskonstanten, det vill säga tiden det tar för processen att nå 63%, identifieras.

Därefter bestäms λ genom att välja det största av dödtiden och tidskonstanten. Detta multiplicerar man med en faktor som oftast har värdet 1, 2 eller 3 (men kan vara lägre och högre om så krävs), beroende på hur hög hastighet man vill att reglerkretsen ska ha. Denna faktor betecknas p . Ju lägre faktor man använder desto snabbare reglering får man.[1]

2.6 Kaskadreglering

Kaskadreglering är en reglerform där processerna kan delas upp i flera seriekopplade delar som påverkas av olika störningar. Största fördelen med kaskadreglering, jämfört med vanlig återkoppling, är att den kan motverka ”snabba” störningar på ett tidigare stadium.

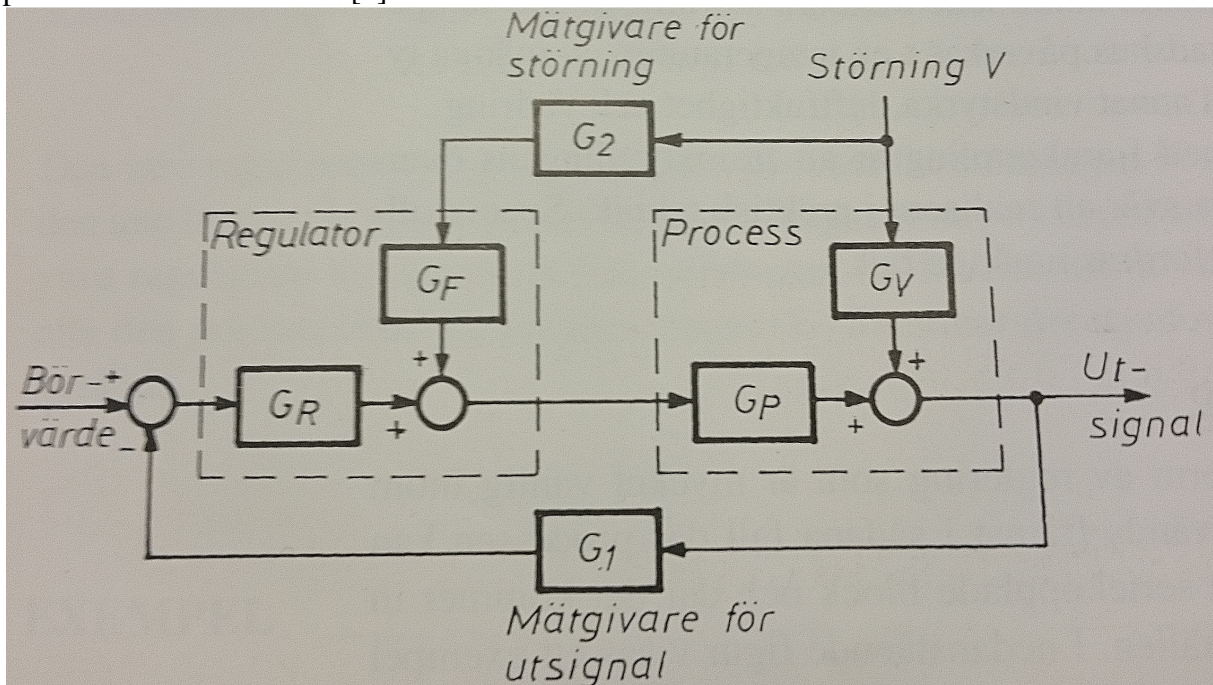
Kaskad-regleringens blockschema består av en inre loop och en yttre loop. Den inre regulatorns uppgift är att ta hand om de ”snabba” störningarna innan de påverkar den yttre loopen. Se figur 2 för ett generellt blockschema. [1]



Figur 2. Generellt blockschema för kaskadreglering.[1]

2.7 Framkoppling

Framkoppling används i ett system där man har mätbara störningar, se figur 3 för ett generellt blockschema. Principen går ut på att man mäter störningen och utnyttjar den informationen för att förbättra regleringen. Teoretiskt kan detta medföra att störningarna som verkar på processen helt kan tas bort.[1]



Figur 3. Generellt blockschema för framkoppling.[1]

2.8 Derivatafilter

Ett derivatafilters uppgift är att trycka ner derivatans inverkan på styrsignalen vid höga frekvenser. Den stora amplitudförstärkningen hos derivatan medför att störningarna förstärks väldigt mycket. Detta resulterar i ett instabilt system. Då derivatafilter används blir regleringen mycket mjukare istället. Brytfrekvensen för filtret betecknas T_F och är approximativt lika med $0.2 \cdot T_D$. [1]

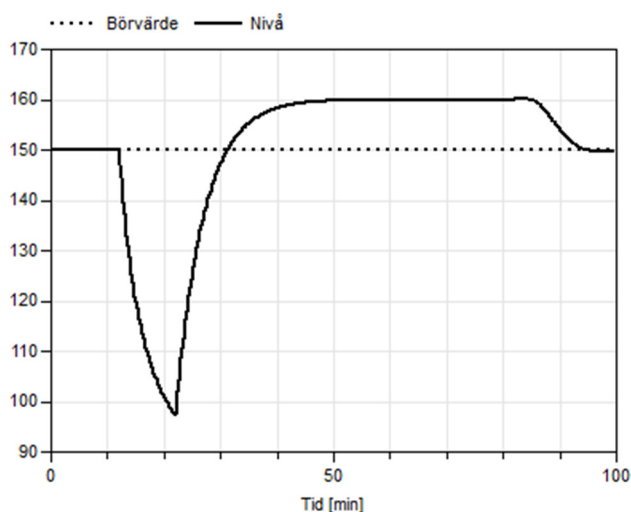
2.9 Anti Wind-Up

Wind-Up eller integratoruppvridning uppstår när styrsignalen är mättad och inte kan styra in styrsystemet till korrekt nivå. Exempel på en mättad styrsignal är en pump som inte kan ge mer än max eller mindre än noll, en ventil som inte kan öppnas mer än max eller stängas mer än helt. Det som händer är att integraldelen i regulatorn fortsätter försöka öka styrsignalen trots att det inte är fysiskt möjligt. När begränsningen sedan släpper på grund av att en störning har försvunnit, har regulatorns integraldel hunnit räkna upp för mycket och då krävs ett stort fel åt motsatt håll innan styrsignalen minskar från sitt mättade värde.

Ett exempel på integratoruppvridning som påminner om tankprocessen finns i figur 4 här nedan. För att hålla nivån 150 i tanken krävs ett flöde som motsvarar 90% av kapaciteten i en ventil i normaldrift och för nivån 160 krävs 100% öppen ventil. Vid 12 minuter kommer en störning på tankens utflöde vilket medför att ventilen öppnar till 100%, men ändå inte kan hålla nivån. Under de 10 minuter som störningen verkar på systemet hinner regulatorns utsignal räkna vidare uppåt från 100% till över 200%, vilket ventilen inte kan ge. När störningen väl släpper ligger ventilen kvar på 100% och tanken börjar fyllas upp. Då tanknivån nått upp till 150 börjar regulatorns utsignal att minska från drygt 200%, vilket medför att ventilen ligger på 100%. Sakta räknar regulatorn ner men ventilen börjar inte stängas förrän regulatorns utsignal nått ner under 100%.

Eftersom utsignaler över 100% i detta fall inte är användbara är det lämpligt att begränsa regulatorns utsignal till 100%. Men det räcker inte, integralens värde måste också begränsas för att undvika uppvridningen.

Det finns flera olika metoder för att förhindra detta. Generellt går de ut på att begränsa integralens värde då styrsignalen är mättad, för att sedan ta bort begränsningen när styrsignalen är inom ett kontrollerbart intervall igen.



Figur 4. Exempel på integratoruppvridning

2.10 Digitala och analoga signaler

En digital signal är uppbyggd av en eller flera bitar. Det vill säga logiska informationsenheter som endast kan anta två värden, en etta eller en nolla. Ettan tolkas som sant och nollan som falskt, om inte inverterad logik används. I projektet används huvudsakligen digitala signaler för att kontrollera magnetventiler och för att tända/släcka lampor samt läsa knapptryckningar. Den typ av variabel som används i dessa fall är en BOOL, det vill säga en bit som kan anta värdena sant eller falskt.

En analog signal är en signal som kan anta oändligt många värden, till skillnad från den digitala signalen som bara kan anta två. Analoga signaler används i projektet för att läsa tankens ärvärde samt skriva styrvärden till pumpen. Eftersom PLC-systemet bara jobbar med digitala signaler internt måste de analoga signalerna konverteras för att de ska kunna behandlas. Konverteringen kallas för A/D-omvandling, för analogt till digitalt, eller för D/A-omvandling som motsvarar motsatsen. För att omvandlingarna skall vara möjliga måste PLC-systemet vara utrustat med A/D- och D/A-moduler. Det som bestämmer noggrannheten på t.ex. mätvärden som kommer in till systemet i form av en analog spänning är upplösningen hos A/D-omvandlaren. En A/D-omvandlare med för låg upplösning genererar en digital signal där en stor del av informationen går förlorad, vilket medför att matningen inte blir exakt. För att undvika detta är det viktigt att tänka på hur stor noggrannhet som krävs av systemet. För reglersystem är A/D- och D/A-omvandlarna särskilt viktiga eftersom mätvärden och styrvärden ofta är analoga signaler. En dålig upplösning på så väl insignalen som utsignalen kan medföra en hackig och dålig reglering oavsett hur bra regulatoren är inställd.

2.11 PLC-moduler

En PLC-modul är en del i ett modulärt PLC-system. Modulerna har olika funktioner, t.ex. finns det ingångs- och utgångs-moduler men också mer specialiserade moduler som kan användas för kommunikation med hjälp av olika industriella standarder. Den stora vinningen med ett modulärt PLC-system är att modulerna enkelt kan bytas ut om en annan funktion önskas. Bytet är mycket enkelt att utföra och tar endast några minuter.

NA-9372

NA-9372 är en programmerbar MODBUS I/O. MODBUS är ett protokoll för kommunikation inom PLC-världen. Den programmeras i CoDeSys och kan maximalt hantera 2 taskar. Den kan programmeras med alla språk inom programspråksstandarden IEC 61131-3.

I/O:t har 4 MB programminne, samt 4 MB dataminne. Man kan ansluta totalt 63 modulenheter till I/O:t. Enheten kan utföra 90 instruktioner på 7 μ s.

För att kommunicera har den 2st Ethernet-portar och en RS232/RS485 port (som kan anslutas till en touch-skärm). [2]

ST-1218

ST-1218 är en digital input modul med 8 digitala ingångar. Datatypen som accepteras är av typen BOOL (1 bit). Modulen tar normalt 24V på ingångarna för att det ska uppfattas som en etta, men kan trigga minimalt på 10,2V och maximalt på 28,8V. Se figur 4 för en detaljerad bild av systemet. [3] [4]

ST-2328

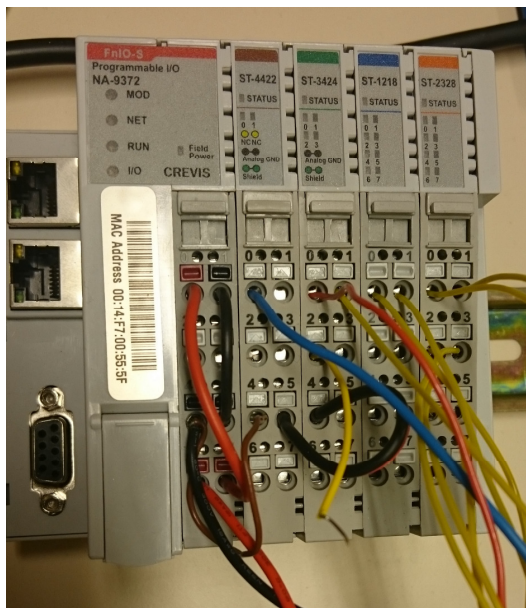
ST-2328 är en digital output modul med 8 digitala utgångar. Datatypen som accepteras är av typen BOOL (1 bit). Modulen skickar normalt 24V ut från utgångarna för en logisk etta, men kan skicka ut minst 11V och mest 28,8V. [5] [6]

ST-4422

ST-4422 är en analog output modul med 2 analoga utgångar. Datatypen som accepteras är av typen WORD (16 bitar), men modulen kan bara ta emot 12 bitar och ge en maximal utspänning på 10V. Modulen har en cykeltid på 2 millisekunder. [7][8]

ST-3424

ST-3424 är en analog input modul med 4 analoga ingångar. Datatypen som accepteras är av typen WORD (16 bitar), men modulen kan bara ta emot 12 bitar och ta emot en maximal inspänning på 10V. Modulens cykeltid är 4 millisekunder. [9][10]



*Figur 5. PLC-modulerna som använts under arbetets gång.
Från vänster till höger: NA9372, ST-4422, ST-3424, ST-1218, ST-2328*

3. METOD

3.1 Stegsvarsanalys

Stegsvarsanalys har utförts på processen, både för enkel och dubbeltank. Målsättningen med detta har varit att ta fram en så exakt överföringsfunktion som möjligt vilket är viktigt för att uppnå en bra reglering. Pumpen har körts på 50% då det har varit lite vatten i tanken, efter en stund när systemet har stabiliserats sig erhålls en slutgiltig högre nivå. Med hjälp av diagram och grafer har därefter en överföringsfunktion bestämts med största noggrannhet. Slutligen plottas överföringsfunktionen i samma MATLAB-fönster som stegsvaret så de enkelt kan jämföras.

3.2 Dimensionering av regulatorer

Ett flertal regulatorer har dimensionerats under arbetets gång, både för enkel tank och dubbel tank. Dessutom har ett antal mer avancerade reglerprinciper också använts och dimensionerats. I synnerhet har lambda-metoden använts eftersom den ger en möjlighet till att styra den slutgiltiga regleringens hastighet och stabilitet. Lambda-metoden har också fördelen att den går att använda utan djupare kunskaper i reglerteknik och därmed är lämplig för en laboration. En balanserad reglering har varit målet, med försök att få snabbheten och stabiliteten så bra och balanserad som möjlig.

För att ha något att jämföra med användes Ziegler-Nichols svängningsmetod i början av arbetet, dock förkastades den genast på grund av dåligt resultat. I fallet med störningslogik har regulatorn för dubbeltank använts och då en störning kommer ökas P-verkan kortvarigt för att sedan återgå till det normala. Denna metod skiljer sig mycket från de resterande metoderna.

3.3 Tester och dokumentation

Ett flertal tester har genomförts för att ge en så tydlig bild som möjligt över hur regleringen fungerar med de olika reglerprinciperna samt med och utan dödtid. För att minimera mänsklig påverkan på resultaten samt andra oönskade parametrar har ett standardiserat test tagits fram. Detta test går ut på att först utsätta regulatorn för ett börvärdessteg, låta den stabilisera sig för att därefter testa hur bra systemet klarar av en stor störning samt testa regulatorns förmåga att motverka integral Wind-Up. Slutligen körs ett program som öppnar magnetventilerna slumpat under slumpad tid.

Resultatet har plottats kontinuerligt i MATLAB, vanligtvis har testerna utförts under 1000 sekunder. Detta medför att systemet har haft tid att svänga in sig och återhämta sig efter störningarna. Slumpade störningar användes framförallt för att eliminera mänsklig påverkan av testresultaten, men också eftersom de är en mycket realistisk motsvarighet till de störningar som förekommer i en verklig miljö.

Problem som har uppstått under arbetets gång har självfallet också dokumenterats i ett antal dokument, detta för att underlätta så mycket som möjligt för framtida användare. Även konfigurationen av systemet (PLC m.m.) har dokumenterats i ett särskilt dokument, även detta för att underlätta för framtida användare (se bilaga B och bilaga C).

4. UTFÖRANDE

I arbetet har flera olika reglerprinciper undersökts och implementerats i PLC-systemet i form av kod i structured text. De har sedan genomgått ett antal tester med bland annat dödtid och störningar. Varje test presenteras i detta kapitel i form av grafer och text. Även de överföringsfunktioner som har identifierats med hjälp av stegsvarsanalys presenteras.

Där dimensioneringen har skett med lambdametoden har ett lambda-värde på $p=0,5$ valts. Detta medför en reglering som är väldigt balanserad, ganska snabb med ändå stabil. Högre eller lägre lambdavärde medför att regleringen blir antingen extremt långsam eller instabil. För att möjliggöra dimensionering med lambda-metoden har processens överföringsfunktion approximerats till en första ordningens överföringsfunktion. (Se ekvation 4.2 och 4.4). Denna approximation bidrar till en modellbegränsning, som är nödvändig att genomföra för att kunna använda lambdametoden. Approximationen går huvudsakligen ut på att en tidskonstant uppskattas ur stegsvaret, dvs. att överföringsfunktionen betraktas som en första ordningens överföringsfunktion eventuellt med dödtid.[1]

De olika regulatorerna testades med olika stora pålagda dödtider för att få ett mått på hur robusta systemen är. Mer dödtid hade kunnat hanteras om regulatorerna från början hade dimensionerats för att styra en process med dödtid, dock var det inte syftet i detta fall.

4.1 Implementering i PLC

För att styra processen har en PLC används, som har programmerats enligt de aktuella reglerprinciperna. Till en början användes en färdig PID-regulator ur OSCAT-biblioteket för att utföra tester på systemet samt testa enklare reglering. Den regulatorn hade dock vissa begränsningar som endast kunde överkommas genom att designa en egen regulator från grunden i structured text. Alla tester som presenteras i rapporten är utförda med regulatorer implementerade i structured text.

Det första steget var att implementera en generell algoritm som sedan kunde utvecklas med fler och mer avancerade funktioner. (se ekvation 4.1 och 4.2 samt bilaga A).

$$e = \text{Börvärde} - \text{Ärvärde} \quad (4.1)$$

$$P = K * e \quad (4.2)$$

$$w = w + w_0 + e \quad (4.3)$$

$$I = K * \frac{h}{T_I} * w \quad (4.4)$$

$$D = K * T_D \frac{e - e_0}{h} \quad (4.5)$$

$$y = (P + I + D) \quad (4.6)$$

Där:

e = reglerfelet

w₀ = gammal integralsumma

K = Regulatorns förstärkning

H = Samplingstiden

T_D = Regulatorns derivatid

y = utsignalen

w = integralsumma

P = Regulatorns P-verkan

I = Regulatorns I-verkan

T_I = Regulatorns integraltid

e₀ = Ett sampel gammalt reglerfel

Eftersom PLC-systemet är tidsdiskret och överföringsfunktionerna som används är tidskontinuerliga görs en approximation för att beräkna integralen av felet samt derivatan av felet, se ekvation (4.4) samt (4.5). Approximationen för integral bygger på att integralen av felet kan approximeras med summan av reglerfelet. Det stämmer väldigt bra så länge samplingstiden h inte är för stor. Derivatan kan på liknande sätt approximeras som det aktuella reglerfelet minus det gamla reglerfelet dividerat med samplingstiden. Även den här approximationen stämmer väldigt bra så länge samplingstiden inte är för stor.

Summan av P, I, D bildar slutligen y som skrivs ut till pumpen med hjälp av A/D-modulen, ST-4422. En viktig sak att notera är att ST-4422 kräver en variabel av typen WORD samtidigt som resultatet y är av datatypen REAL. En konvertering från REAL till WORD krävs för att erhålla en utsignal. Samma sak, men omvänt gäller för ärvärdena som läses in med ST-3424.

Den största fördelen med att använda PLC-systemet är att det är väldigt enkelt att ändra på parametrar och regleralgoritmer samt lägga till eller ta bort sidofunktioner. Den grundläggande regulatorn utökades med en funktion som skall förhindra integratoruppvridning samt ett derivatafilter. Integratoruppvridningen förhindras genom att integralsumman inte uppdateras då styrsignalen är bottenad. (för detaljer se bilaga A Program PIDET).

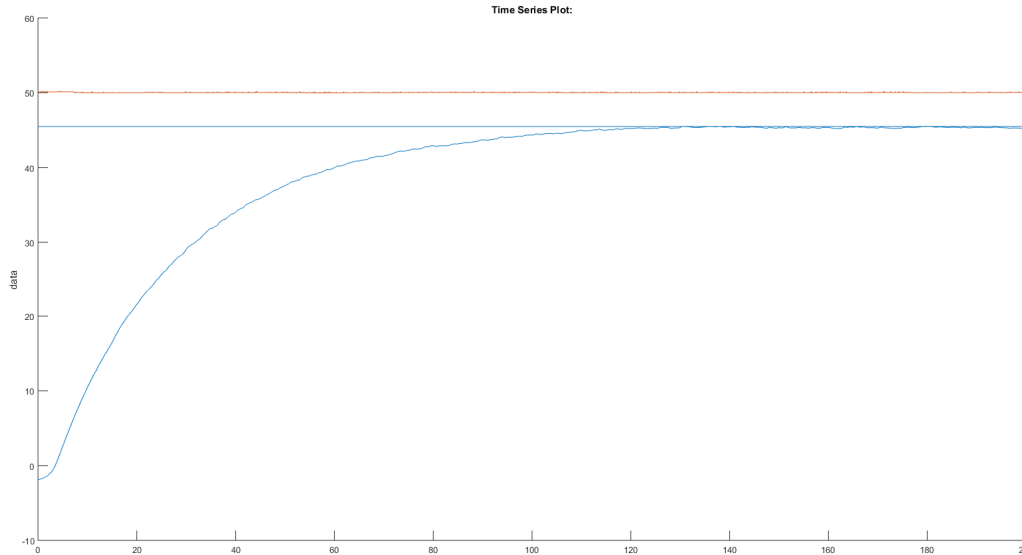
Med hjälp av de två sidofunktionerna var PID-regulatorn komplett. För att enkelt kunna slå på och av systemet samt växla mellan manuellt och automatiskt läge används monostabila tryckknappar samt två lampor. En tryckknapp används för att starta systemet i automatiskt läge, den andra används för att starta systemet i manuellt läge. För att indikera i vilket läge systemet är används två lampor, en som lyser då det är i automatiskt läge samt en som lyser då det är i manuellt läge.

För att kunna utföra tester på hur bra systemet klarar av störningar kodades ett program som genererar slumpade störningar. Störningarna förekommer i form magnetventiler som öppnas och stängs en slumpad tid vid ett slumpat tillfälle. Grunden i programmet är två slumptalsgeneratorer, i form av två block hämtade ur OSCAT-biblioteket. (se bilaga A Program RNDVENT). Ett av blocken används för att generera ett slumptal som används för att avgöra om en ventil ska öppnas eller ej. Det andra blocket genererar ett slumptal som används för att avgöra under hur lång tid ventilen ska vara öppen. Med hjälp av timers styrs sedan magnetventilerna som är kopplade till PLC-systemets digitala utgångsmodul, ST-2328. För att inte sannolikheten för störningar ska bli för stor används ett exekveringsvillkor vilket medför att störningsprogrammet inte exekveras lika ofta som övriga program.

Övriga regulatorer som har används i projektet har varit helt eller till stor del baserade på den grundläggande PID-programmet för enkeltank, men med andra parametrar. Även den dynamiska störningslogiken är baserad på PID-programmet även om den skiljer sig markant eftersom det inte finns ett konstant k -värde.(se bilaga A program FKDBT för detaljer).

4.2 Enkeltank överföringsfunktion

Här presenteras överföringsfunktionen för enkelttank med tillhörande stegsvar. Två tidskonstanter har valts eftersom detta ger en överföringsfunktion med något större noggrannhet än vad en tidskonstant skulle ge. Se figur 6 för stegsvaret.



Figur 6. Stegsvaret enkelttank.

Testet utfördes under 200 sekunder med alla ventiler förutom utloppsventil 4 stängda. Med hjälp av stegvarsidentifiering av en överföringsfunktion med två tidskonstanter, samt grafen ovan togs följande överföringsfunktion fram:

$$G(s) = \frac{K}{(1+T_1s)(1+T_2s)} \quad (4.7)$$

Där:

$$K=96.8$$

$$T_1=27.68$$

$$T_2=2.214$$

Den andra tidskonstanten är mycket mindre än den första, vilket innebär att den i vissa fall kan försummas, dock tas den ändå med eftersom stor akkuratess eftertraktas i detta fall.

För dimensioneringen av en regulator med lambda-metoden approximerades dock den erhållna överföringsfunktionen med en första ordningens överföringsfunktion som har utseende enligt följande:

$$G(s) = \frac{K}{1+Ts} \quad (4.8)$$

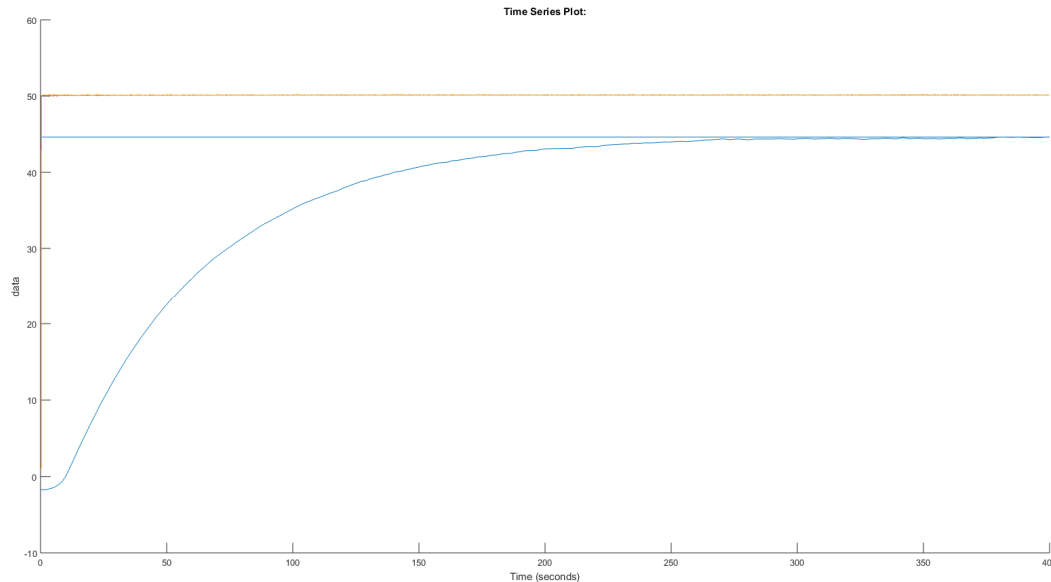
Där:

$$K=96.8$$

$$T=32$$

4.3 Dubbeltank överföringsfunktion

Här presenteras överföringsfunktionen för dubbeltank med tillhörande stegsvar. I detta fall har två tidskonstanter valts eftersom det passar processen bäst. Se figur 7 för stegsvaret.



Figur 7. Stegsvaret dubbeltank.

Testet utfördes under 400 sekunder med ventil 3, 5 och 6 öppna. Med hjälp av stegvarsidentifiering av en överföringsfunktion med två tidskonstanter, samt grafen ovan togs följande överföringsfunktion fram:

$$G(s) = \frac{K}{(1+T_1s)(1+T_2s)} \quad (4.9)$$

Där:

$$K=92.6$$

$$T_1=58.2455$$

$$T_2=4.951$$

I detta fall är tidskonstanterna mycket större, vilket inte var oväntat. En liten dödtid kan observeras, dock är den försumbar eftersom den är väldigt liten i förhållande till stigtiden.

För dimensioneringen av en regulator med lambda-metoden approximerades dock den erhållna överföringsfunktionen med en första ordningens överföringsfunktion som har utseende enligt följande:

$$G(s) = \frac{K}{1+Ts} \quad (4.10)$$

Där:

$$K=96.8$$

$$T=60.4$$

4.4 Enkeltanksreglering med störningar och dödtid

Enkeltanksregleringen har utförts med en PID-regulator som har ställts in enligt lambda-metoden.

PID-regulatorns överföringsfunktion med parametrar framtagna med hjälp av lambdametoden ger följande:

$$G(s) = K \left(1 + \frac{1}{T_I s} + \frac{T_D s}{1 + T_F s} \right) \quad (4.11)$$

Där:

$$K = 2.022$$

$$T_I = 32$$

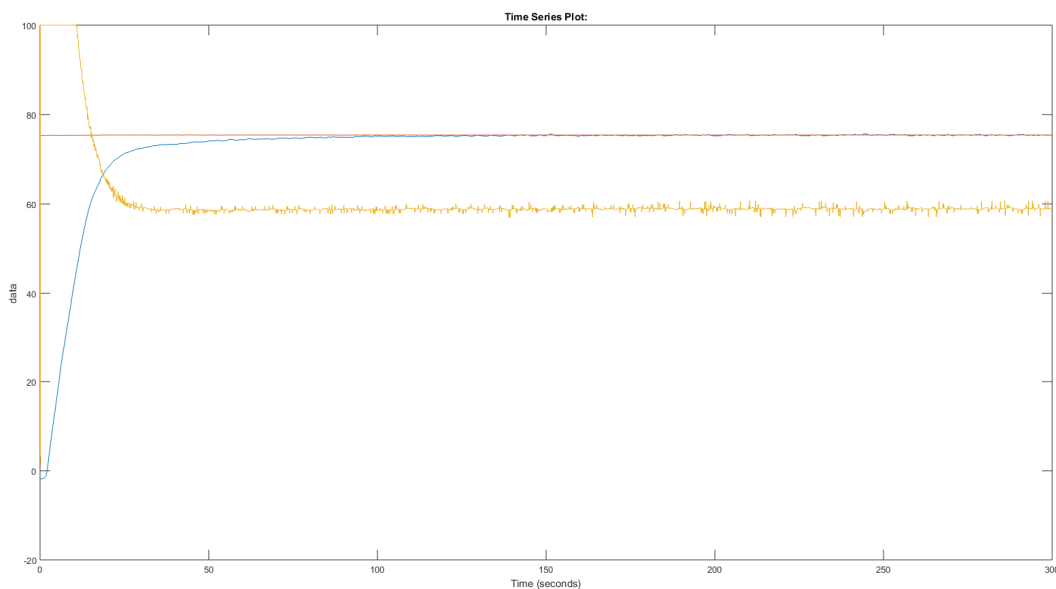
$$T_D = 0.35$$

$$T_F = 0.07$$

Tillämpningen av regleringen i form av kod i structured text finns i bilaga A.

Nedan följer ett antal tester.

Först testades en stegformad börvärdesändring för att se att systemet svängde in sig som förväntat. Stegsvartestet utfördes under 300 sekunder (se figur 8).

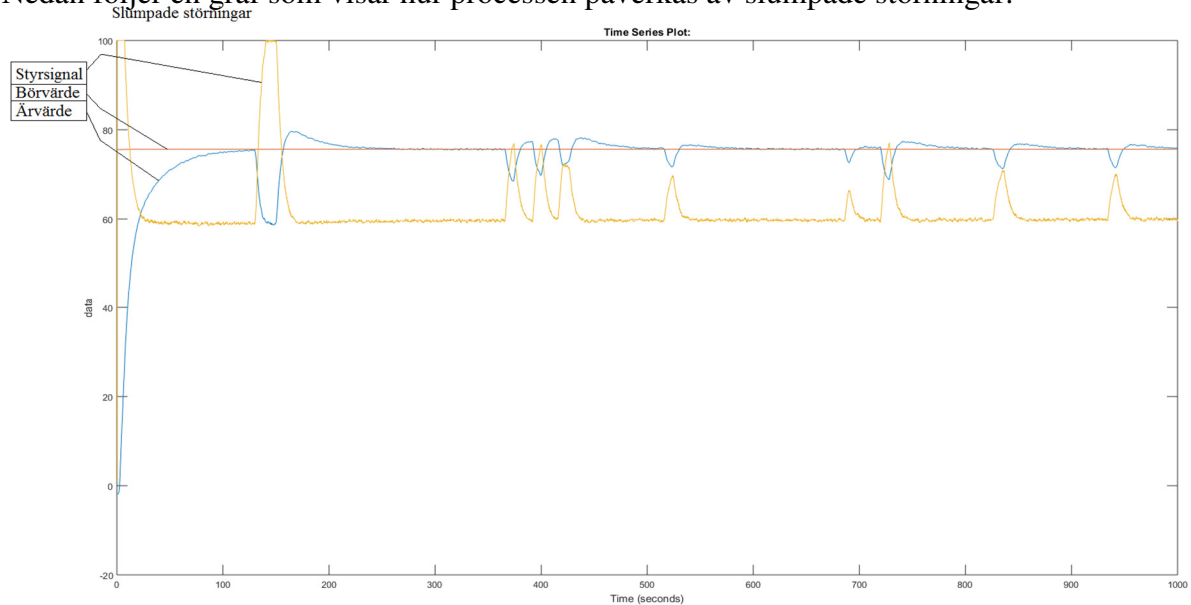


Figur 8. Stegsvart enkelttank med PID-reglering.

Efter börvärdesändringen stabiliserar sig processen snabbt och har ingen översväng. Detta är ett resultat som förväntas med det relativt låga värde på p som valts, $p=0,5$.

Nästa test bestod av både ett börvärdessteg och störningar. Testet utfördes under 1000 sekunder med störningar som kommer slumpat med den ungefärliga sannolikheten för störning på 4 % varje två sekunder (se figur 9). Störningarna kom också med slumpade intervall och verkade en slumpad tid. Regulatorns anti wind-up har också testats när styrsignalen blir mättad under den första stora störningen.

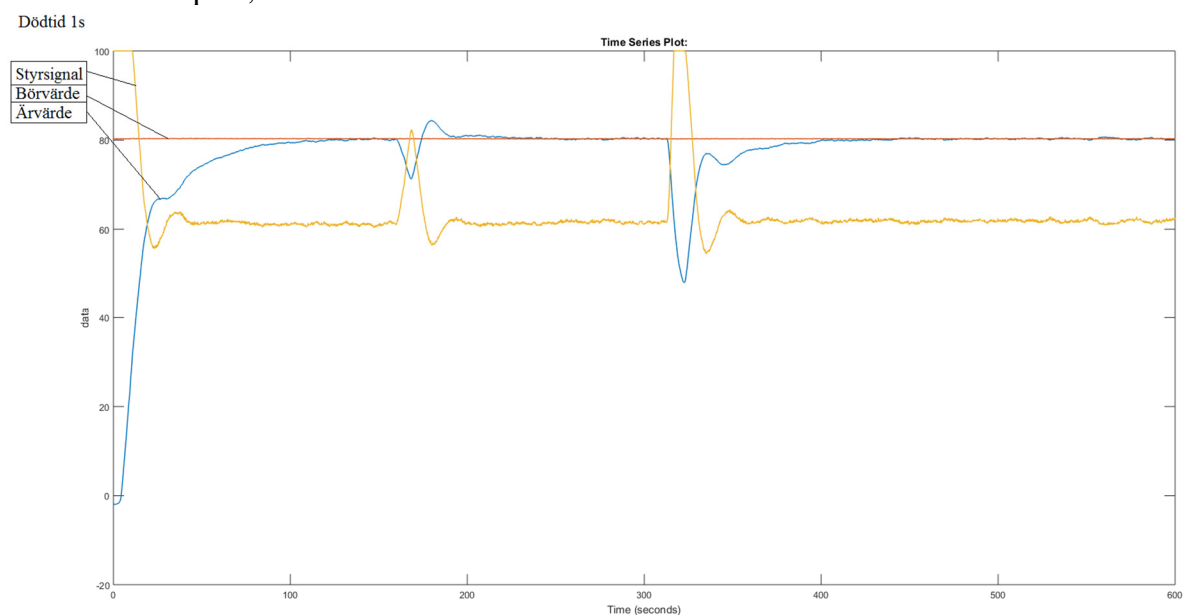
Nedan följer en graf som visar hur processen påverkas av slumpade störningar:



Figur 9. Enkeltanksreglering med slumpade störningar.

Resultatet är mycket tillfredställande, en snabb och stabil reglering med mycket liten översväng efter störningarna som ställer in sig snabbt har uppnåtts. Även anti wind-up-funktionen fungerar som den ska när styrsignalen (gul kurva figur 9) går ner från 100% när tanknivån går tillbaka till börvärdet.

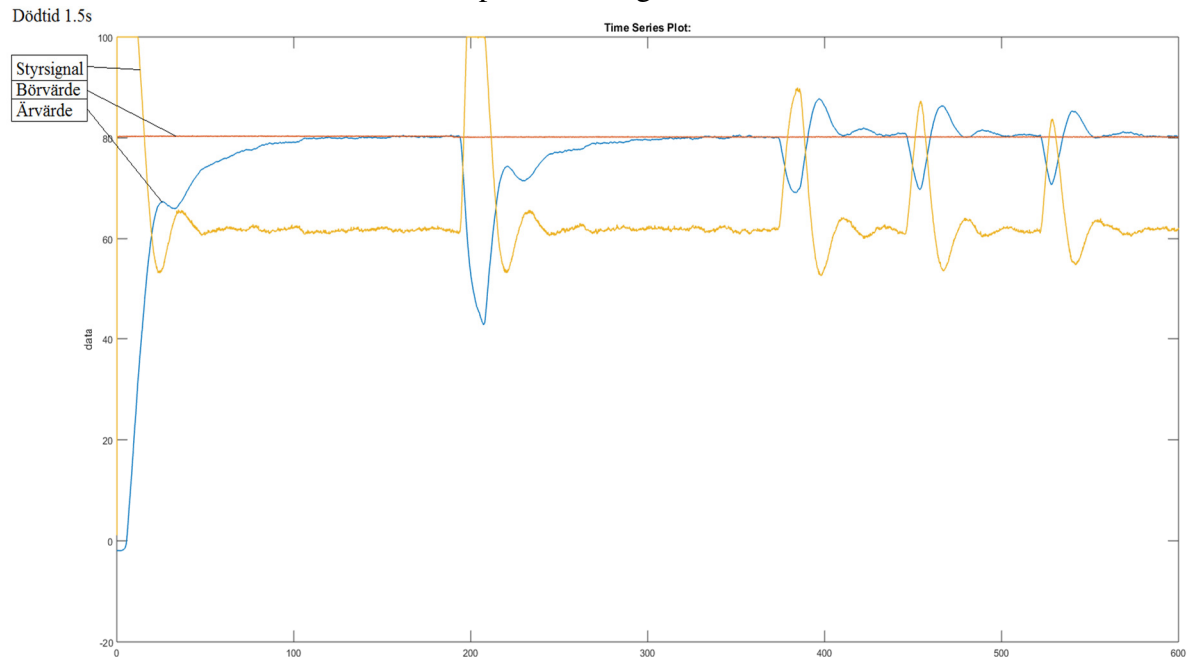
För att få ett mått på regulatorns robusthet och tolerans för modellfel utsattes regulatorn för ett antal tester med olika dödtid. Testet som illustreras i figur 10 har utförts under 600 sekunder med en dödtid på 1,0 sekund.



Figur 10. Enkeltanksreglering med 1 sekunds dödtid samt störningar.

Även under detta försök verkade slumpade störningar. Anti wind-up testades också (se figur 10). Resultatet skiljer sig inte nämnvärt från föregående test.

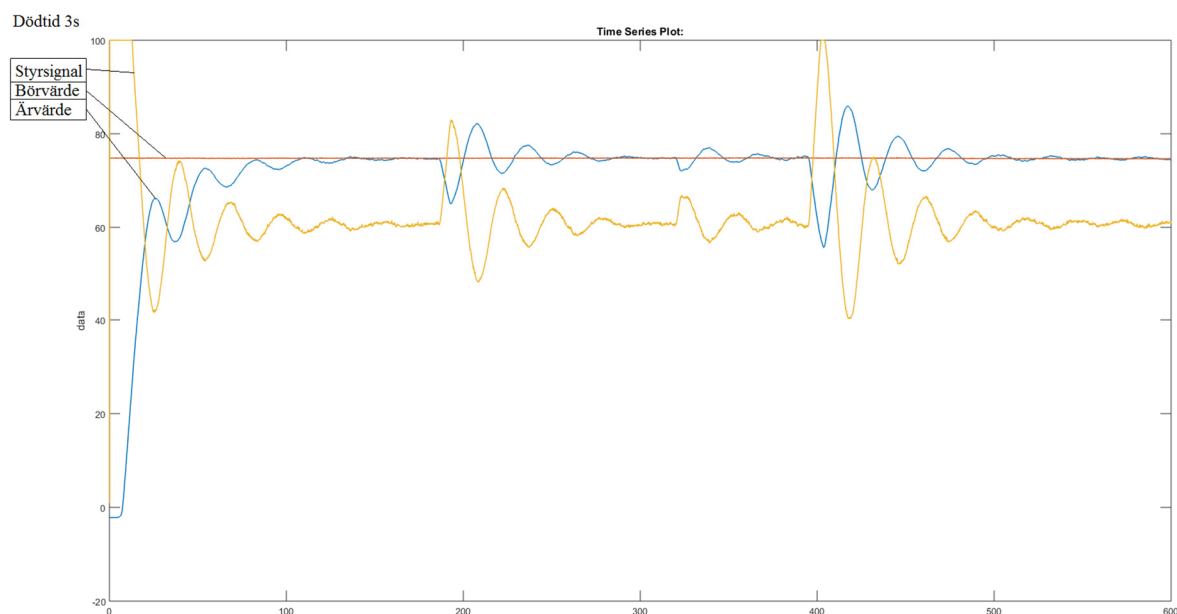
En ytterligare ökning av dödtiden till 1,5 sekunder gav resultat enligt figur 11. Testet har utförts under 600 sekunder med slumpade störningar.



Figur 11. Enkeltanksreglering med 1,5 sekunders dödtid samt störningar.

Till skillnad från det tidigare testet märks nu dödtiden ganska tydligt i form av kraftigare översväng efter störningar som tar längre tid att avta samt en liten undersväng i början av stegsvaret. Även i detta fall ses tydligt att anti wind-up-funktionen har fått arbeta när styrsignalen har nått sitt maxvärde.

I nästa test ökades dödtiden till 3 sekunder (se figur 12).



Figur 12. Enkeltanksreglering med 3 sekunders dödtid samt störningar.

Dödtiden märks nu väldigt tydligt, insvängningen är inte lika jämn som tidigare och störningarna hanteras mycket sämre än tidigare. Systemet är på gränsen till instabilt, vid tester med mer dödtid svänger systemet in efter oacceptabelt lång tid och klarar inte av störningar alls. Därför dras gränsen för vad regulatören klarar av att kontrollera vid 3 sekunders dödtid.

Som nämnt i början på detta kapitel bör dock noteras att mer dödtid kan hanteras av en PID-regulator om den från början dimensioneras för en process som har dödtid. Dock är dödtiden i testerna ingen dödtid som ligger hos processen i sig, det är en pålagd dödtid som endast används för att ge ett mått på robustheten hos systemen.

4.5 Dubbeltanksreglering med störningar och dödtid

Dubbeltanken har reglerats med en PID-regulator som är dimensionerad med hjälp av lambda-metoden. PID-regulatorns överföringsfunktion med parametrar framtagna med hjälp av lambdametoden ger följande:

$$G(s) = K \left(1 + \frac{1}{T_I s} + \frac{T_D s}{1 + T_F s} \right) \quad (4.12)$$

Där:

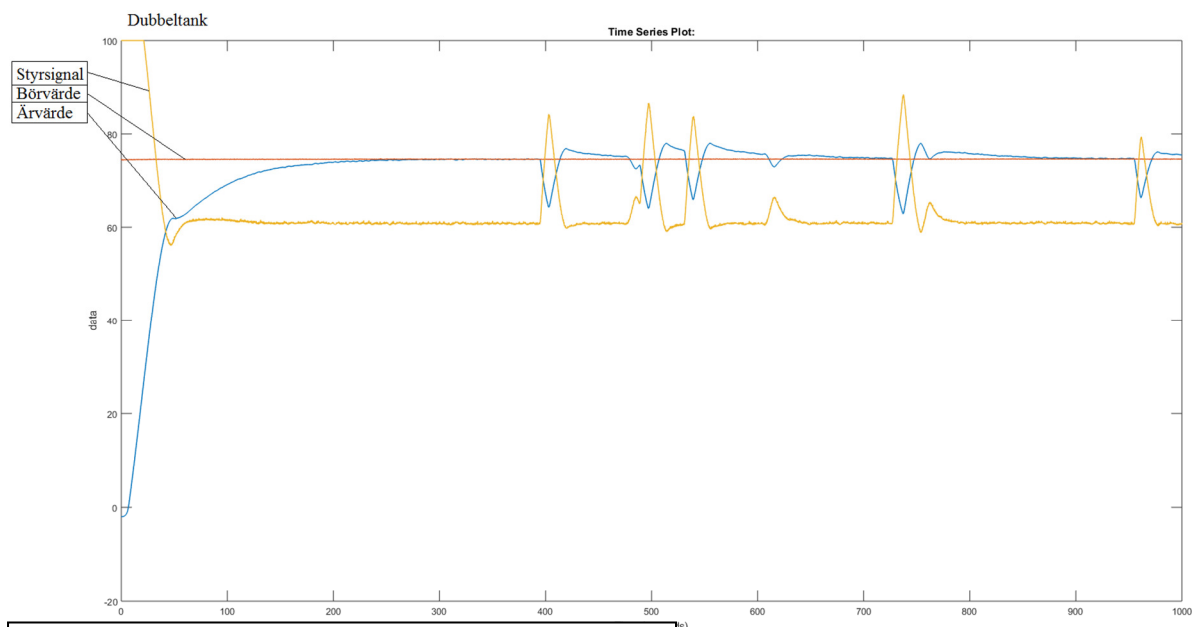
$$K = 2.067$$

$$T_I = 60.4$$

$$T_D = 1.35$$

$$T_F = 0.27$$

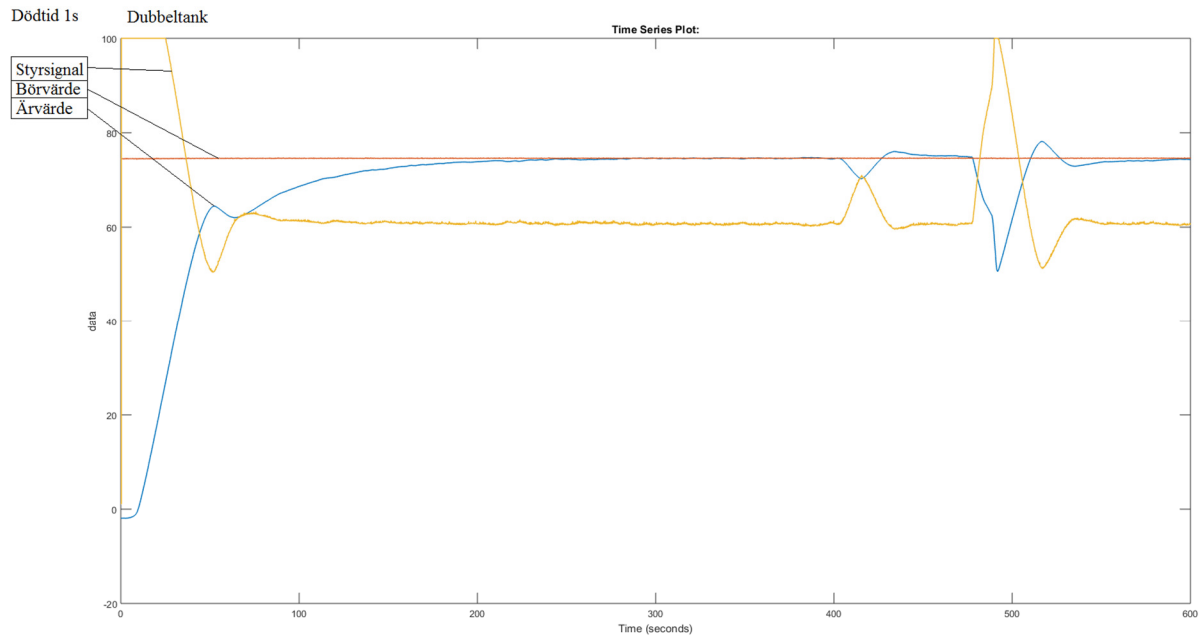
Ventil nummer 4 och 5 har varit öppna under testerna, resten har varit stängda. Testet har utförts under 1000 sekunder med ett initialt börvärdessteg och därefter slumpade störningar för att testa regulatorns stabilitet och snabbhet. Sannolikheten för störning är här precis som tidigare på 4 % varje två sekunder (se figur 13). Tillämpningen av regleringen i form av kod i structured text finns i bilaga A. Här nedan redovisas resultatet i form av flertalet grafer.



Figur 13. Dubbeltanksreglering med störningar.

Regulatorns hastighet är mycket tillfredställande samtidigt som ingen översväng uppstår. Störningarna hanteras också bra, ingen nämnvärd översväng där heller.

För att testa regulatorns robusthet och hur väl den klarar modellfel gjordes tester med en extra dödtid i systemet. Det första testet har utförts under 600 sekunder med slumpade störningar. Dödtiden är i detta fall på 1,0 sekund (se figur 14). Nedan följer flertalet grafer som påvisar hur dödtid påverkar processen:

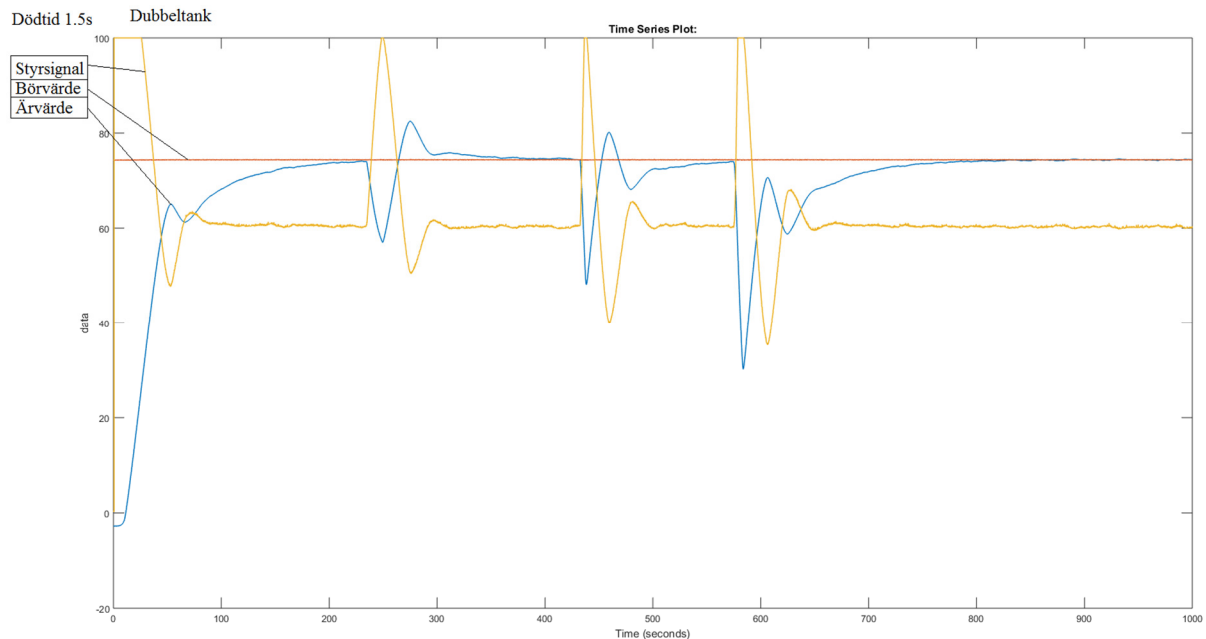


Figur 14. Dubbeltanksreglering med störningar samt dödtid på 1,0 s.

Påverkan av dödtiden är här något tydligare än vad den var för enkeltanken, en liten undersväng kan genast observeras vid insvängningsförloppet. Skillnaden mot reglering utan dödtid är framförallt att systemet självklart blir långsammare.

Regleringen fungerar för övrigt bra, störningarna hanteras bra och systemet är aldrig nära instabilitet. Även en mycket stor störning hanteras bra. Efter en störning återhämtar sig systemet väldigt snabbt och översvängen är mycket liten. En snabb och stabil reglering har uppnåtts.

Testet har utförts under 1000 sekunder med slumpade störningar. Dödtiden är i detta fall på 1.5 sekunder (se figur 15).

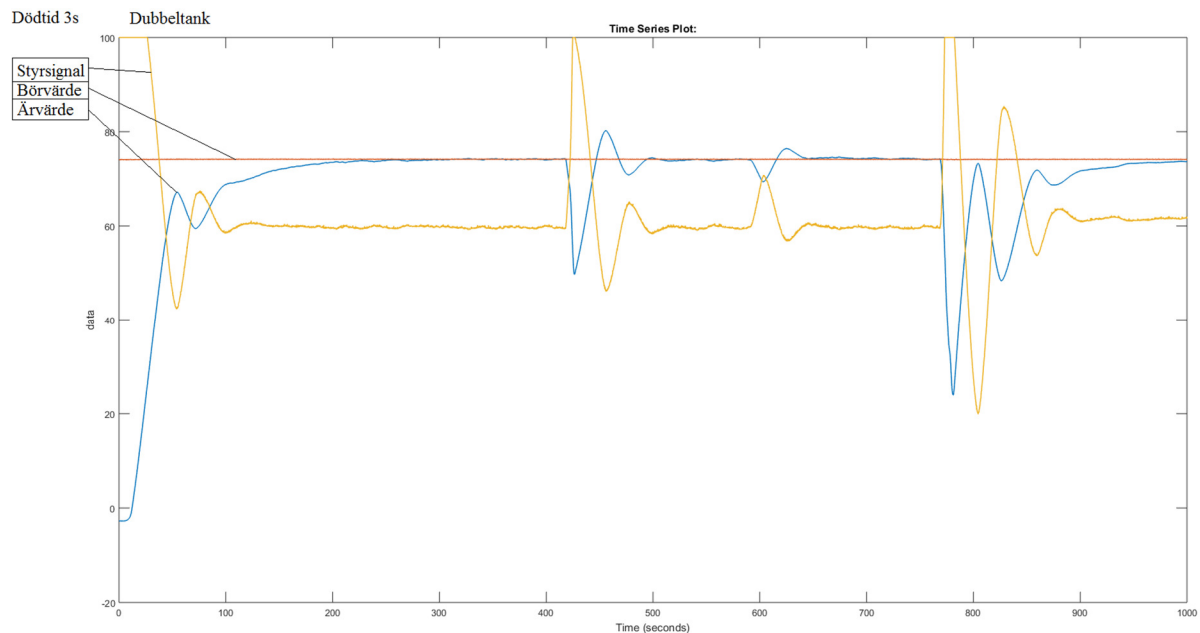


Figur 15. Dubbeltanksreglering med störningar samt dödtid på 1,5 s.

Påverkan av dödtiden blir självfallet mer uppenbar jämfört med det tidigare testet. En något större undersväng kan nu iakttas vid insvängningen. Systemet stabiliserar sig något långsammare än tidigare på grund av dödtiden, och störningskänsligheten blir också större.

En något större översväng kan observeras, men framförallt kan nu även en undersväng observeras tydligt efter den andra störningen. Systemet återhämtar sig dock fortfarande relativt snabbt och är aldrig i närheten av självsvängning eller instabilitet. Den tredje störningen är extra svår för systemet att hantera eftersom den egentligen är två störningar som kommer med en väldigt kort tid emellan. Resultatet blir att systemet tar ganska lång tid på sig att nå börvärdet igen, dock kan här ingen översväng observeras.

Testet här under har utförts under 1000 sekunder med slumpade störningar då systemet uppnår börvärdet. Dödtiden är på 3 sekunder (se figur 16).

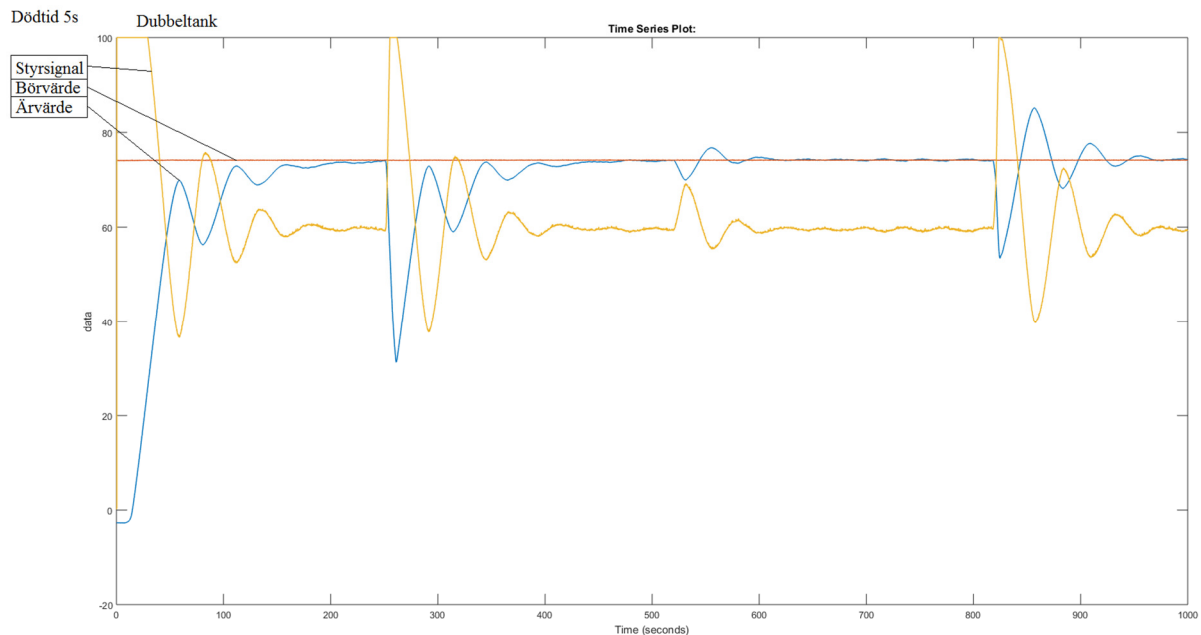


Figur 16. Dubbeltanksreglering med störningar samt dödtid på 3 s.

Systemet blir märkbart mer påverkat med ökande dödtid vilket tydligt kan observeras vid insvängningsförloppet; undersvängen är nu tydligt märkbar. Systemet har självklart också blivit långsammare. Känsligheten för störningar har också ökat.

Efter en störning observeras nu inte bara en översväng utan även en undersväng om störningen är stor. I de flesta fall dör inte svängningen heller ut efter en period utan fortsätter ett tag till, dock med en amplitud som avtar efter varje svängning. Systemet tar märkbart längre tid på sig att stabilisera sig. I fallet med den tredje störningen kommer också en störning till innan systemet har hunnit återhämta sig helt. Här observeras att systemet tar väldigt lång tid på sig att återhämta sig, över 100 sekunder. Dock uppkommer ingen översväng. Instabilitet är också långt bort, även om systemet är märkbart pressat av testet kommer det aldrig i någon självsvängning.

Testet är utfört under 1000 sekunder med en dödtid på 5 sekunder samt slumpade störningar då systemet har svängt in sig (se figur 17).



Figur 17. Dubbeltanksreglering med störningar samt dödtid på 5 s.

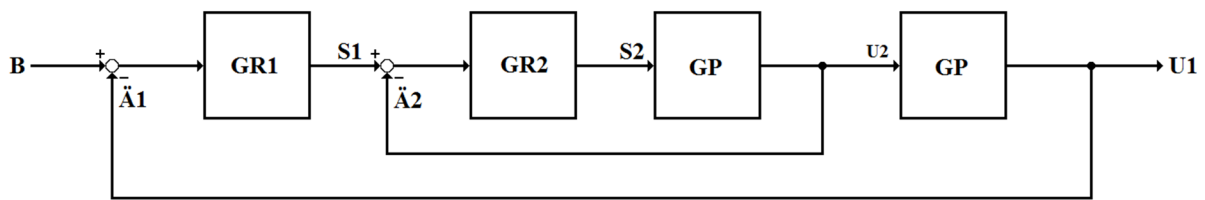
Påverkan av dödtiden är här mycket tydlig, tre undersvängar kan till exempel observeras under insvängningsförloppet. Resultatet blir självklart ett långsammare system. Störningskänsligheten har ökat tydligt.

Efter en störning tar nu systemet längre tid på sig att stabilisera sig och systemet är mycket nära att hamna i självsvängning då det försöker återhämta sig från störningen. Dock minskar fortfarande amplituden på svängningarna med tiden. Resultatet blir att systemet tar längre tid på sig att stabilisera sig efter en störning. Dock svänger systemet fortfarande in efteråt. Översvängen hålls dock fortfarande nere tack vare den försiktiga regleringen, men resultatet av det blir såklart en långsammare reglering. Instabilitet har inte uppnåtts dock är regleringen knappast bra, men den får anses vara acceptabel vid 5,0 sekunders dödtid. Gränsen dras därför här, vid större dödtid uppstår instabilitet och systemet klarar aldrig av att stabilisera sig. Känsligheten för störningar är inte möjlig att testa med större dödtid på grund av att systemet som tidigare nämnt aldrig riktigt svänger in.

Som nämnt i början på detta kapitel bör dock noteras att mer dödtid kan hanteras av en PID-regulator om den från början dimensioneras för en process som har dödtid. Dock är dödtiden i testerna ingen dödtid som ligger hos processen i sig, det är en pålagd dödtid som endast används för att ge ett mått på robustheten hos systemen.

4.6 Kaskadreglering med störningar och dödtid

Kaskadreglering har testats på systemet med alla ventiler stängda förutom ventil 3, 4 och 5. Regulatorn, som består av en inre regulator och en yttre regulator har båda dimensionerats med hjälp av lambda-metoden. I figur 17 kan den kaskadreglering som används i projektet studeras. Den inre regulatorn benämns GR2 och den yttre regulatorn benämns GR1. Den yttre regulatorn är en master som styr den inre regulatorn som agerar slav i systemet. Styrsignalen till pumpen är benämnd u . Den inre regulatorn är vanligtvis väldigt snabb, dock inte i detta fall eftersom båda regulatorerna är dimensionerade med samma p -värde, då målet var att testa och utvärdera reglerprincipen, inte optimera den. Tillämpningen av regleringen i form av kod i structured text finns i bilaga A.



Figur 18. Använd form av kaskadreglering i arbetet.

Nedan följer regleralgoritm med parametrar:

$$GR1(s) = K_1 \left(1 + \frac{1}{T_{I1}s} + \frac{T_{D1}s}{1+T_{F1}s} \right) \quad (4.13)$$

Där:

$$K_1 = 2.022 \quad T_{I1} = 32 \quad T_{D1} = 0.35 \quad T_{F1} = 0.07$$

$$GR2(s) = K_2 \left(1 + \frac{1}{T_{I2}s} + \frac{T_{D2}s}{1+T_{F2}s} \right) \quad (4.14)$$

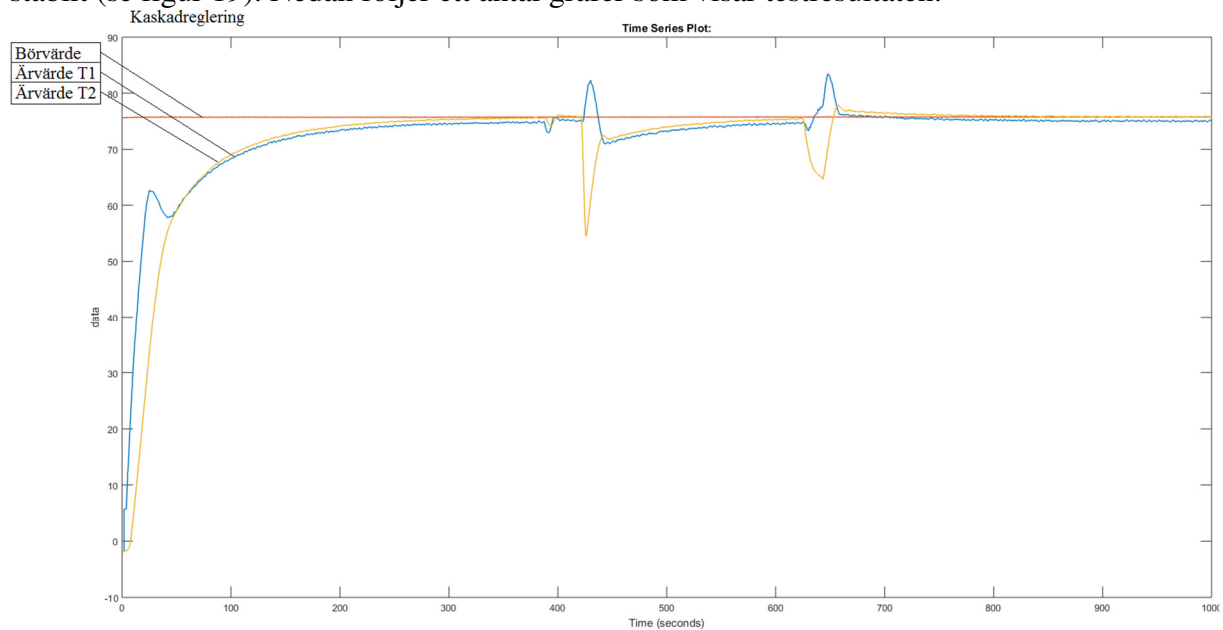
Där:

$$K_2 = 2.067 \quad T_{I2} = 60.4 \quad T_{D2} = 1.35 \quad T_{F2} = 0.27$$

Regulatorn GR1 är här den inre regulatorn och regulatorn GR2 är den yttre regulatorn. Eftersom syftet med dessa tester var att utvärdera hur kaskadreglering kan tillämpas på tanksystemet så är inte regleringen optimal. Då den inre regulatorn tas fram kunde ett annat lambda-värde ha använts för att ge en snabbare inre loop, vilket är önskvärt i kaskadreglering. Även en ren P-regulator i den inre loopen skulle kunna testas. Dock har inte något av detta genomförts eftersom syftet ej var att ta fram en optimal reglering, utan snarare att testa om reglerprincipen kan tillämpas med det aktuella systemet.

Även kaskadregleringen utsattes för samma typ av automatiserade tester som det tidigare dubbeltankssystemet med en regulator.

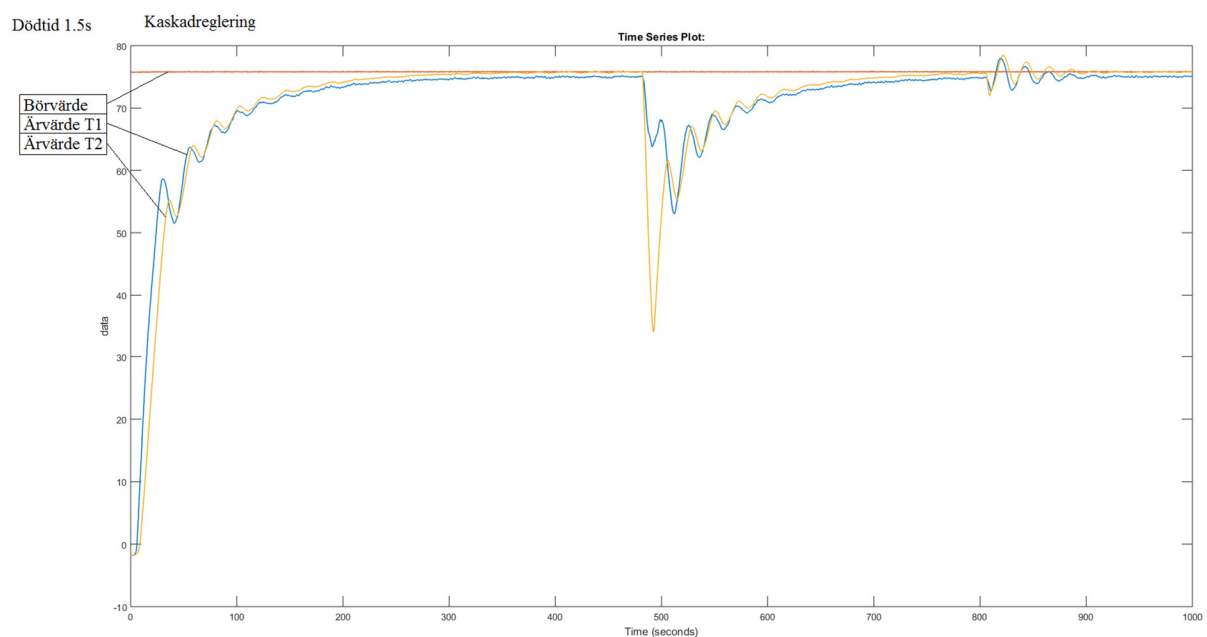
Det första testet har utförts under 1000 sekunder med två störningar då systemet var stabilt (se figur 19). Nedan följer ett antal grafer som visar testresultaten:



Figur 19. Kaskadreglering med störningar.

Regleringen är mycket jämn, dock något långsammare än den vanliga dubbeltanksregleringen. Störningar hanteras dock mycket bra av systemet, eftersom störningen i en tank bekämpas innan den hinner påverka nästa tank.

Testet nedanför utfördes under 1000 sekunder med slumpade störningar samt en dödtid på 1,5 sekunder (se figur 20). Här nedan redovisas resultaten av kaskadreglering med dödtid:

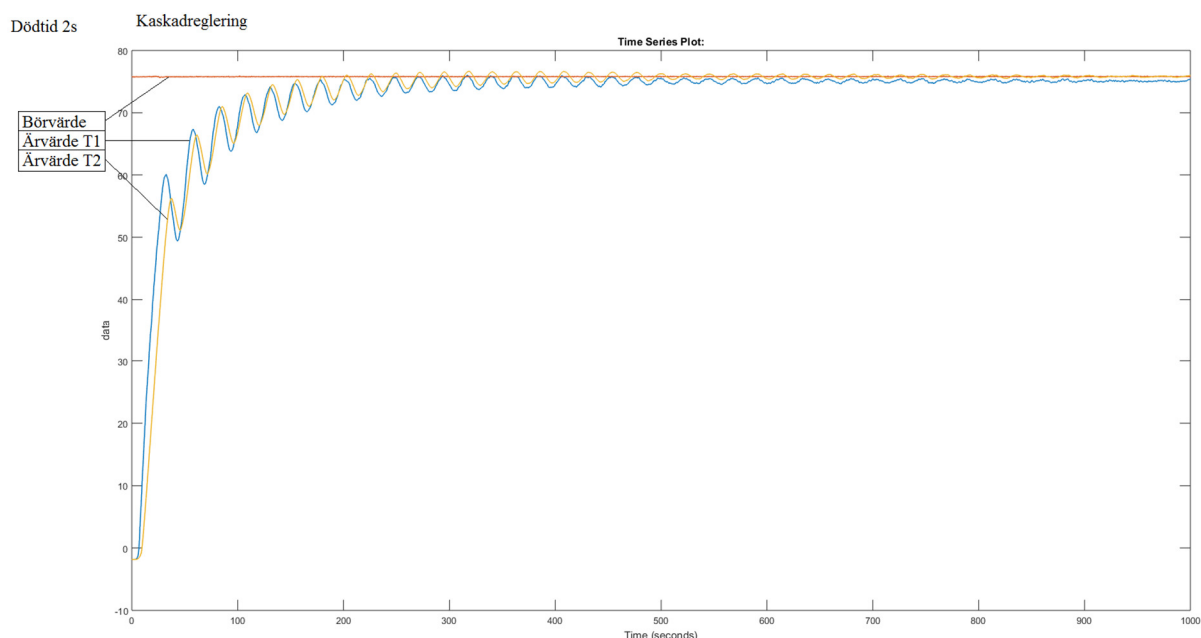


Figur 20. Kaskadreglering med störningar samt dödtid på 1,5 sekunder.

Testet med dödtid på 1,0 sekund redovisas inte här eftersom resultatet är identiskt med det tidigare resultatet utan dödtid. Insvängningen är ganska svängig, undersväng kan observeras på flera ställen under insvängningstiden. Detta i kombination med dödtiden medför att det tar ganska lång tid innan systemet stabiliserar sig.

Störningshanteringen är både bra och dålig, störningarna bekämpas snabbt dock tar det lång tid innan systemet börjar svänga in och det blir rejäla svängningar innan dess. Dock svänger systemet fortfarande in sig och ingen självsvängning uppstår någonsin hur mycket systemet än pressas med stora störningar. Resultatet får anses vara bra även om systemet som tidigare nämnts svänger ganska mycket. Detta tyder på behov av ytterligare trimning.

Testet nedan har utförts under 1000 sekunder med en dödtid på 2.0 sekunder, dock utan störningar eftersom systemet tar så lång tid på sig att stabiliseras (se figur 21).



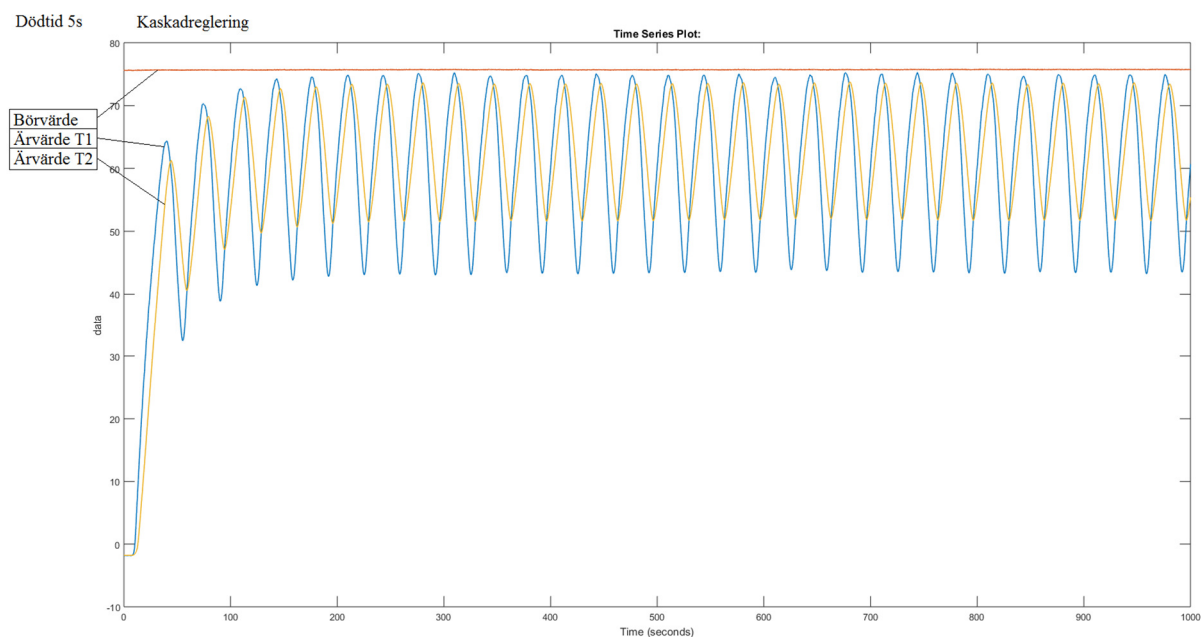
Figur 21. Kaskadreglering med dödtid på 2.0 sekunder.

Insvängningen är nu mycket sämre än tidigare, systemet tar runt 900 sekunder på sig att svänga in, detta är betydligt mycket mer än något tidigare test har varit i närheten av. Dock stabiliserar sig systemet till slut, och slutsatsen som dras av detta är att systemet trots allt är stabilt.

Som nämnts i början på detta kapitel bör dock noteras att mer dödtid kan hanteras av en PID-regulator om den från början dimensioneras för en process som har dödtid. Dock är dödtiden i testerna ingen dödtid som ligger hos processen i sig, det är en pålagd dödtid som endast används för att ge ett mått på robustheten hos systemen.

Systemet med en dödtid på 2 sekunder utsattes aldrig för någon störning eftersom det skulle ta extremt lång tid för systemet att återhämta sig efter störningen. Regleringen är inte bra, och inte snabb heller men den får anses vara acceptabel om man kan acceptera en väldigt långsam insvängning. Det är mycket möjligt att systemet hade stabiliserat sig snabbare med en annan inställning av regulatorerna, men syftet med testet var inte att ta fram en optimal kaskadreglering. Här dras gränsen för vad regulatorn klarar av med aktuella inställningar, nästa test är ett exempel på hur det ser ut när systemet är instabilt.

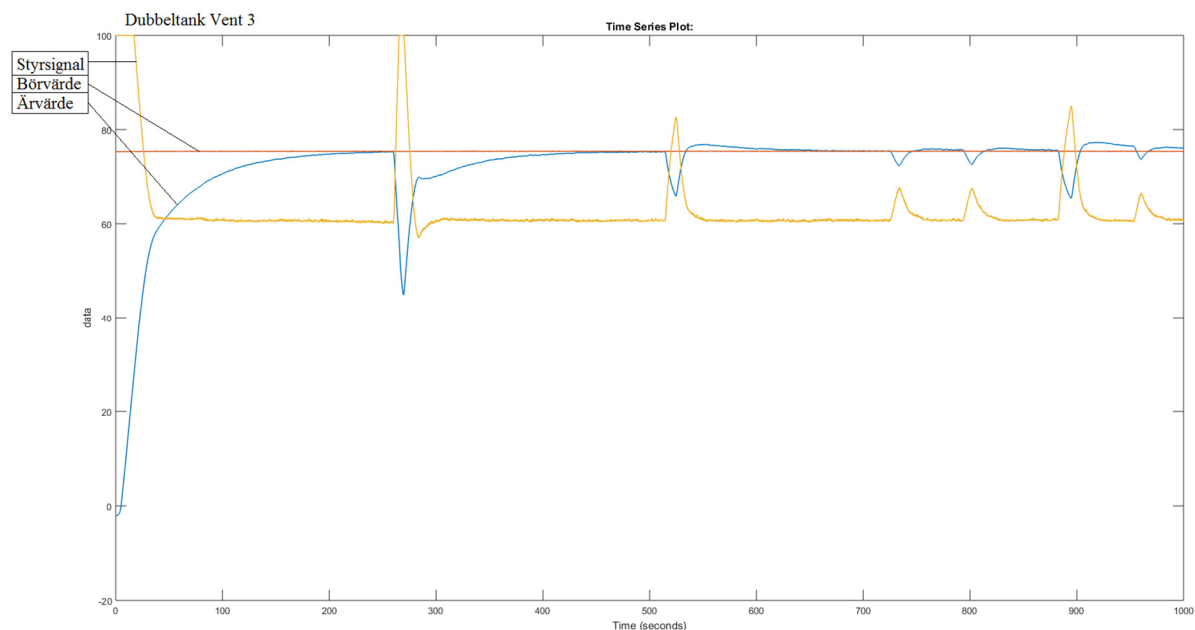
Detta är ett fint exempel på ett instabilt system, där dödtiden ställdes in till 5,0 sekunder. Stabilitet uppnås inte under 1000 sekunders test, och svängningarna avtar inte heller märkbart (se figur 22).



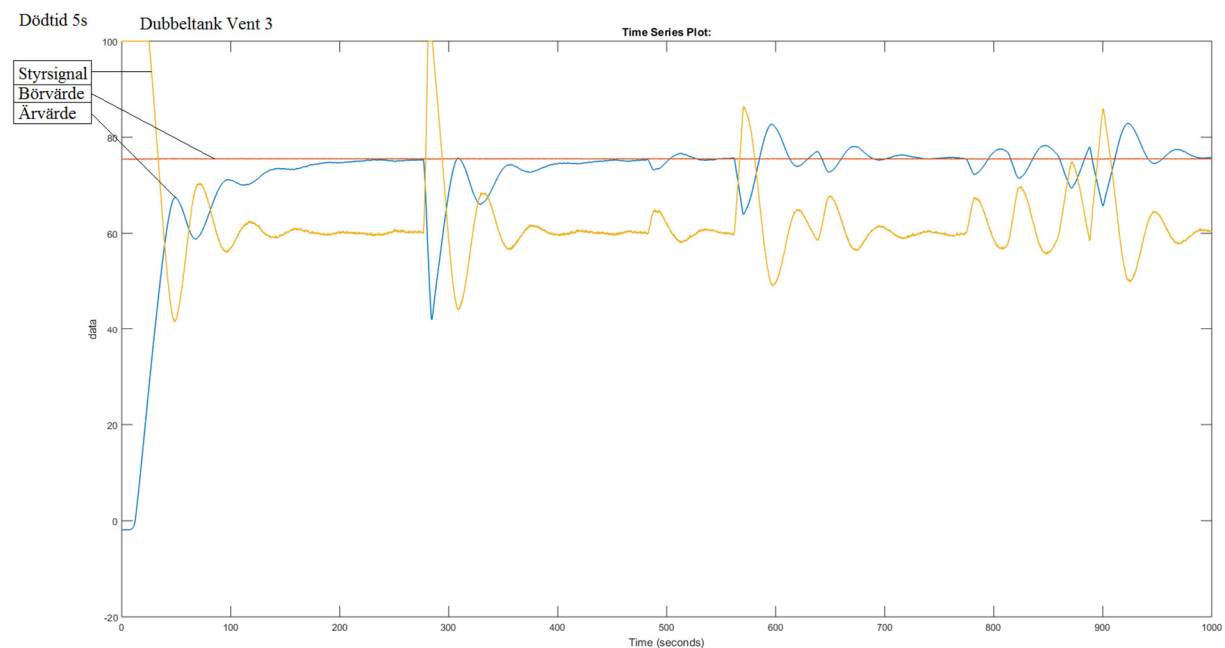
Figur 22. Kaskadreglering med dödtid på 5.0 sekunder.

Regleringen är underkänd, vilket som tidigare nämnts visar att gränsen för vad regulatorn klarar med den aktuella inställningen är korrekt dragen.

Här presenteras två försök med ventil 3 öppen men med vanlig PID-reglering på dubbeltanken, se figur 23.



Figur 23. Dubbeltanksreglering, ventil 3 öppen och slumpade störningar.

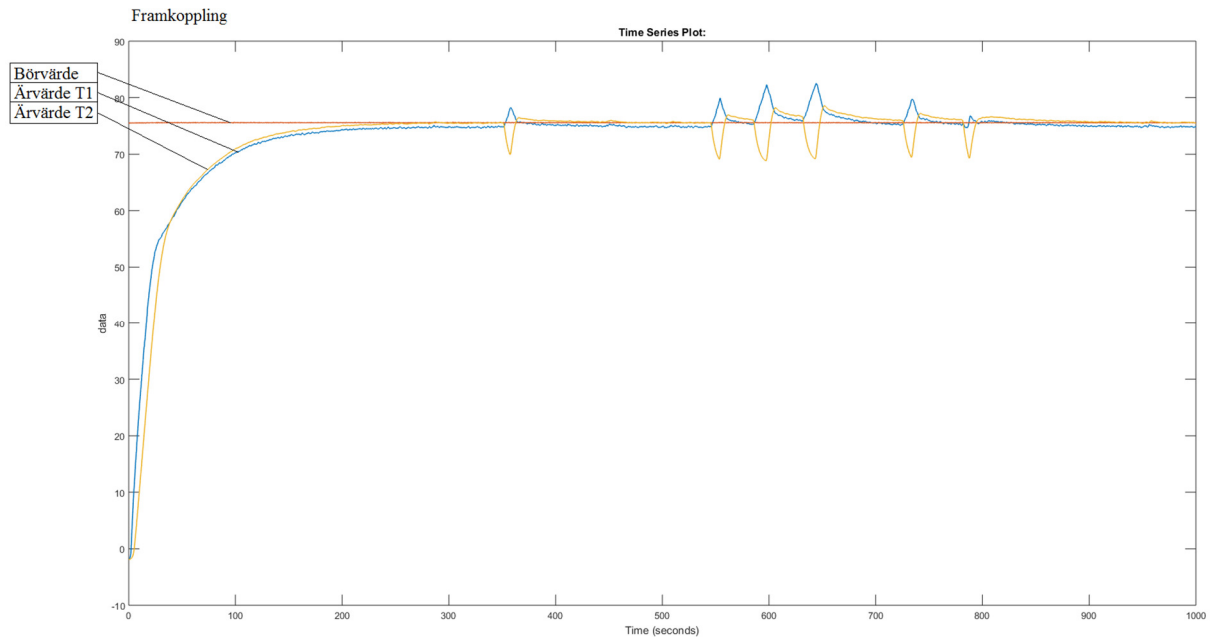


Figur 24. Dubbeltanksreglering, ventil 3 öppen, dödtid 5.0 s och slumpade störningar.

Vid jämförelse av dessa två grafer med motsvarande grafer för kaskadregleringen inses att kaskadregleringen ger en jämnare med något långsammare reglering för dubbeltanken då ingen dödtid förekommer. Dock inses också att vanliga dubbeltanksregleringen med ventil 3 öppen (se figur 23 och figur 24) klarar av 5 sekunders dödtid till skillnad från kaskadregleringen, detta på grund av att regulatorerna kaskadregleringen använder har samma p -värde.

4.7 Dynamisk störningslogik med dödtid

Tester med en dynamisk störningslogik har också utförts. Det som händer är att förstärkningen hos regulatören ändras då en störning uppstår, för att kompensera för störningen. Detta är möjligt eftersom störningarna orsakas av magnetventiler som är kopplade till PLC:t. Den nya förstärkningen används tills ventilen stängs. Tillämpningen av regleringen i form av kod i structured text finns i bilaga A. Testet har utförts under 1000 sekunder med slumpade störningar (se figur 25). Här nedan följer resultaten:

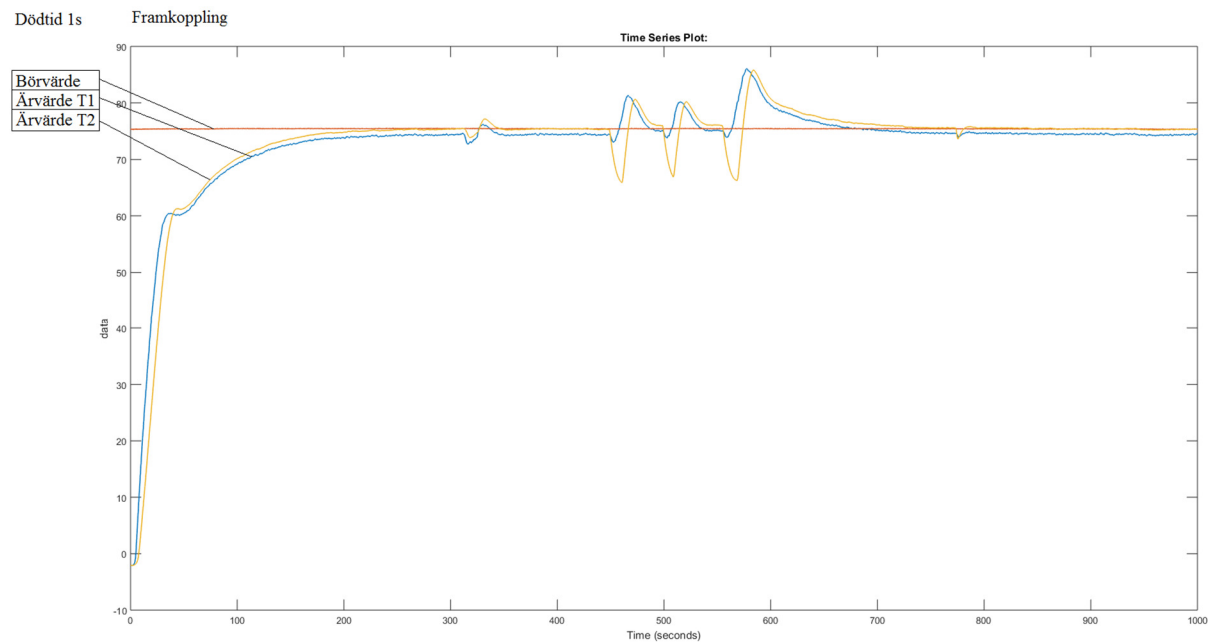


Figur 25. Dynamisk störningslogik dubbeltank.

Insvängningen är snabb, och är identisk med dubbeltankens insvängning eftersom det i grund och botten är samma regulator.

Störningarna hanteras väldigt bra, ärvärdena i de båda tankarna är en direkt spegling av varandra vid en störning eftersom K-verkan ökas då störningen uppstår. Systemet återhämtar sig extremt snabbt efter störningen vilket är en mycket eftersträvad egenskap. Stabiliteten är också mycket bra, eftersom det kraftigare K-värdet kopplas bort så fort störningen är över. K-värdet ökas under en störning till 2,5 vilket ger en snabb men samtidigt inte instabil reglering. En liten översväng kan dock noteras efter att störningen har slutat att verka på systemet. Den beror på att regulatorns I-verkan fortfarande ger ett stort bidrag till styrsignalen.

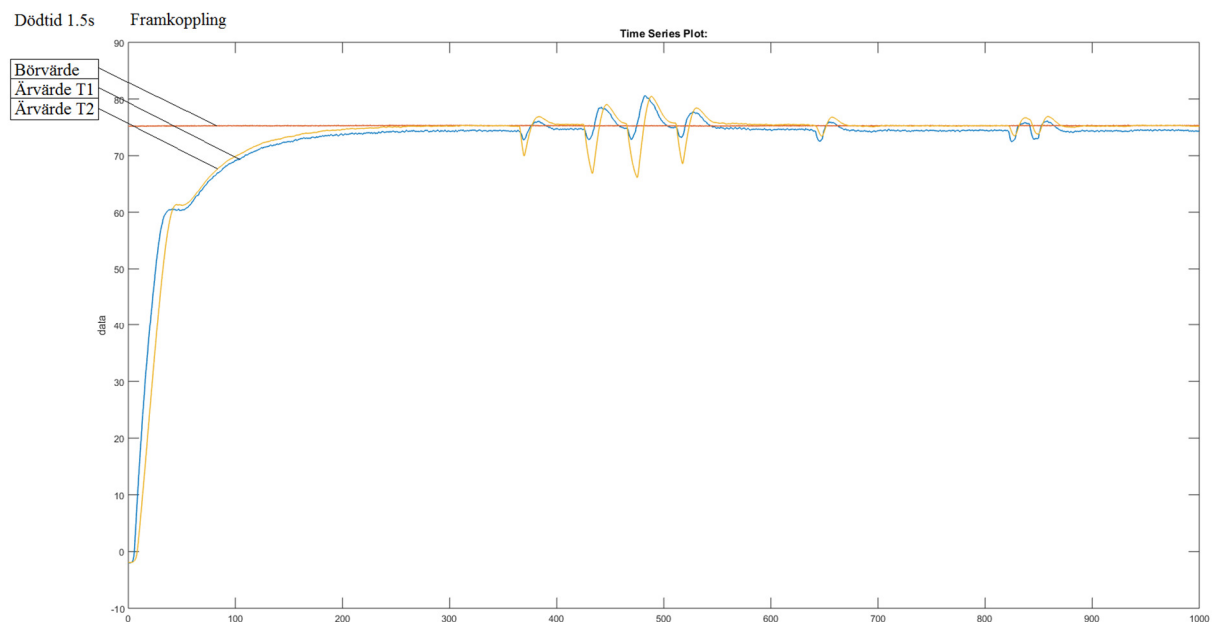
Testet har utförts under 1000 sekunder samt med dödtiden 1.0 sekund och med slumpade störningar (se figur 26).



Figur 26. Dynamisk störningslogik dubbeltank med dödtid 1,0 sekund.

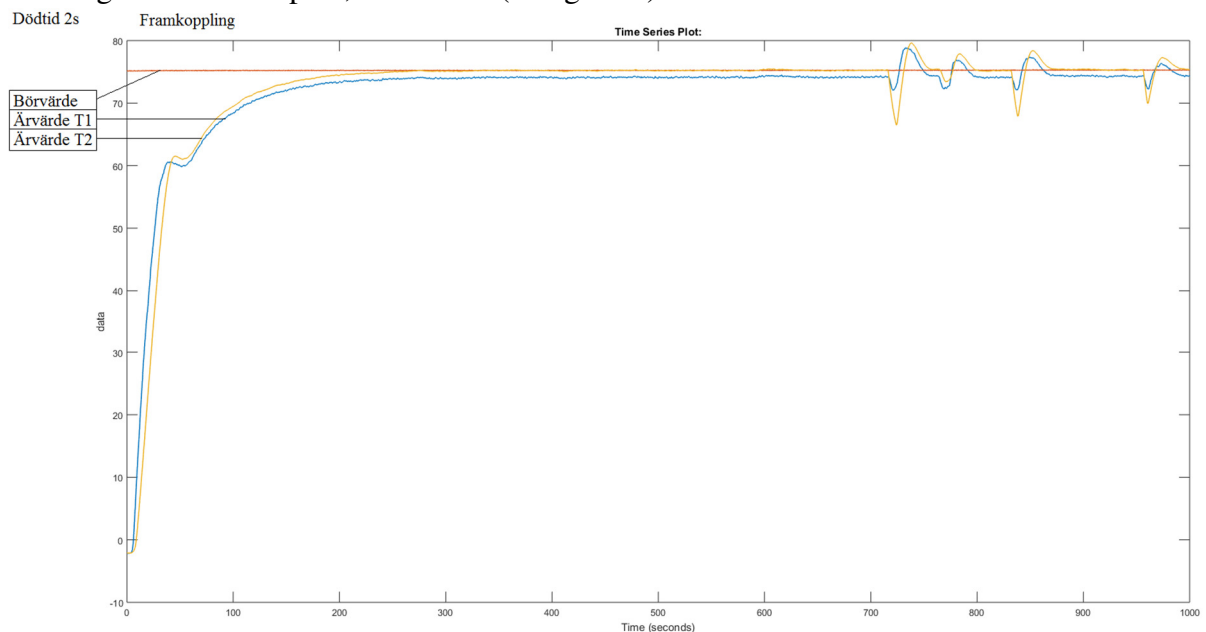
Störningarna hanteras fortfarande väldigt bra, dock märks dödtiden av både vid insvängningen och när systemet ska återhämta sig efter en störning. Instabilitet är dock fortfarande långt bort, och återhämtningen sker fortfarande hyfsat snabbt. Det framgår tydligt att störningen släcks ut efter en översväng, vilket också eftersträvas.

Motsvarande test med dödtid på 1,5 s ses i figur 27. Precis som vid testet med 1.0 sekunder klarar systemet av störningarna bra. En liten tendens till en andra svängning kan dock observeras efter de större störningarna.



Figur 27. Dynamisk störningslogik dubbeltank med dödtid 1,5 sekunder.

Testet här under har precis som tidigare tester utförts under 1000 sekunder med slumpade störningar och dödtid på 2,0 sekunder (se figur 28).

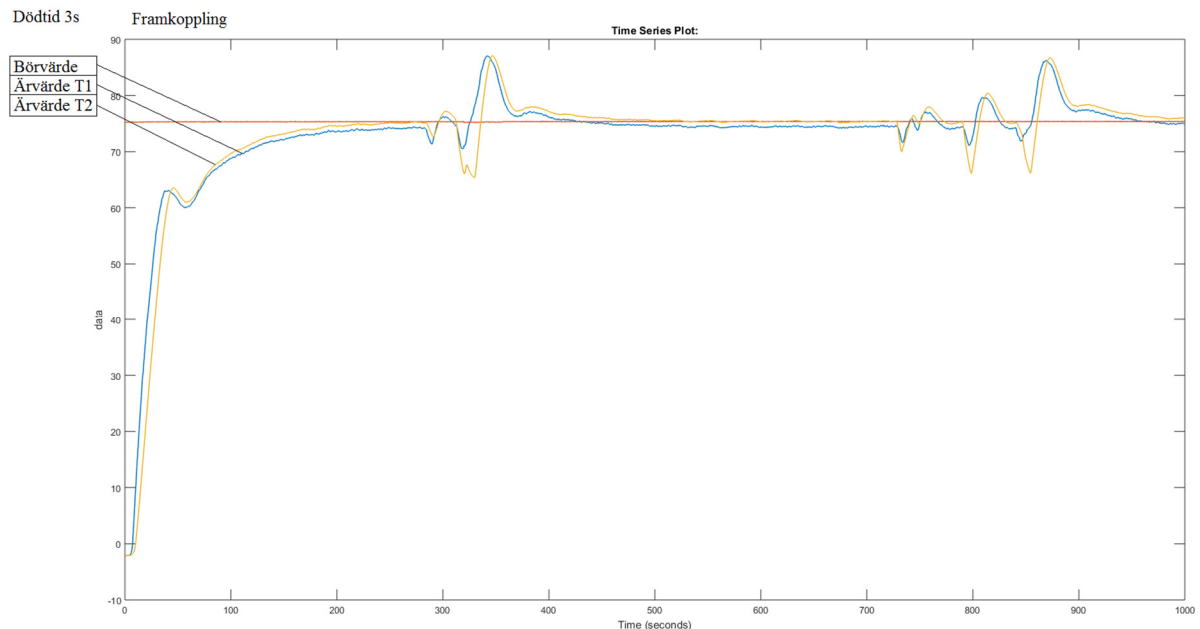


Figur 28. Dynamisk störningslogik dubbeltank med dödtid 2,0 sekunder.

Resultatet är väldigt likt det för dubbeltank med samma dödtid förutom att störningarna hanteras bättre tack vare den dynamiska störningslogiken.

I testerna med en lägre dödtid kunde en tendens till en andra svängning efter störningarna observeras, nu syns den mycket tydligare. Systemet är dock fortfarande stabilt och släcker snabbt ut störningarna samt svänger in. Instabiliteten är inte uppnådd än även om den närmar sig.

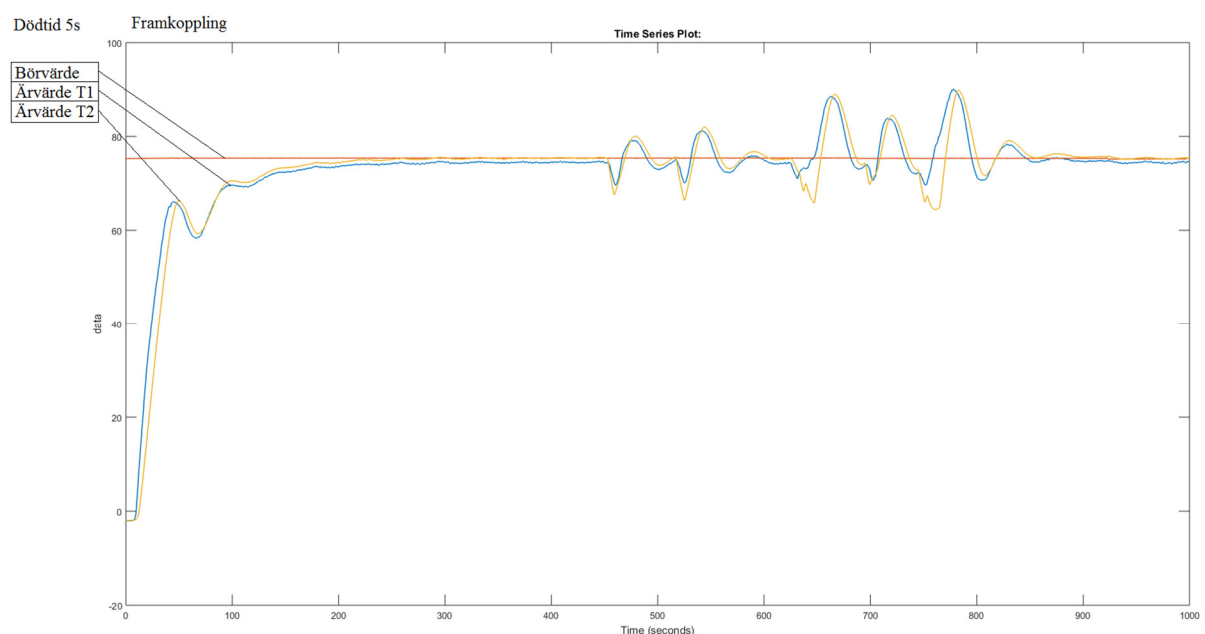
Testet utfördes med en dödtid på 3,0 sekunder under en period på 1000 sekunder. Precis som tidigare observeras att svängningarna tar längre tid att dö ut med ökad dödtid (se figur 29).



Figur 29. Dynamisk störningslogik dubbeltank med dödtid 3,0 sekunder.

Jämfört med den vanliga dubbeltanksregleringen svänger systemet in snabbare men får en relativt kraftig översväng. Instabilitet är dock fortfarande ej ett problem.

Testet utfördes med en dödtid på 5,0 sekunder under en period på 1000 sekunder.



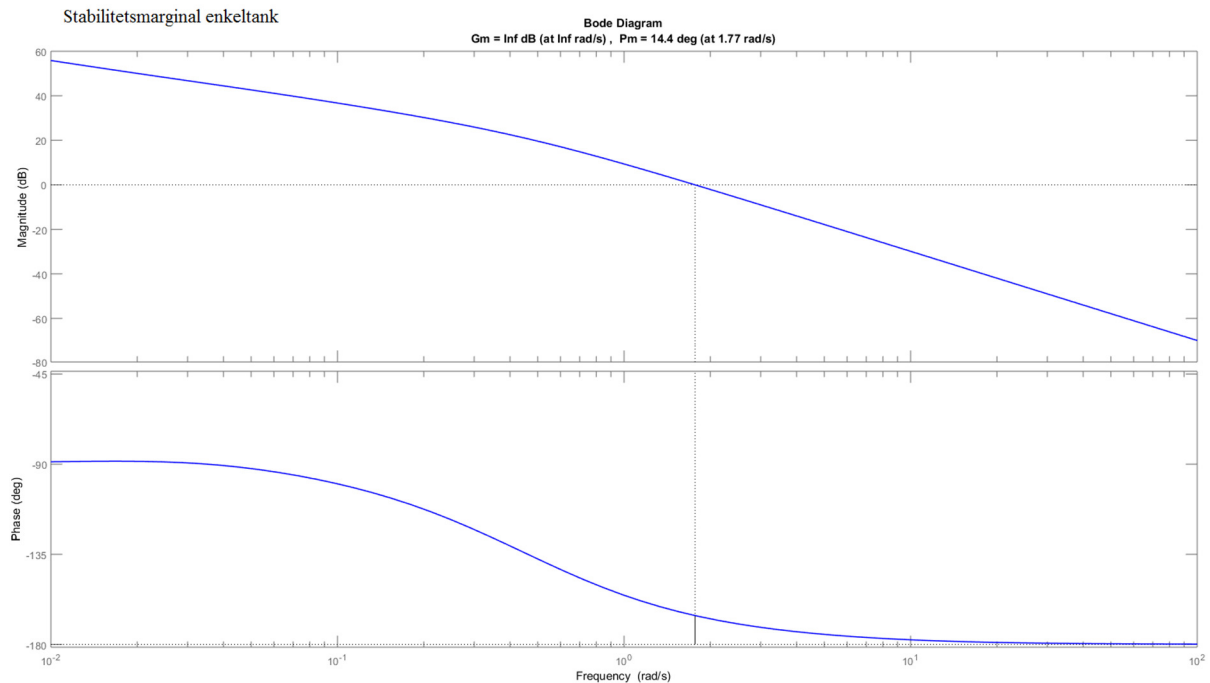
Figur 30. Dynamisk störningslogik dubbeltank med dödtid 5,0 sekunder.

Här observeras tydligt att störningslogiken ger en stor översväng, men är snabb på att svänga in sig efteråt (se figur 30). Anledningen till detta är som nämnts i början av detta underkapitel är eftersom I-verkan ger ett relativt stort bidrag till styrsignalen då störningen har uppstått. Jämfört med den vanliga dubbeltanksregleringen som har mindre översväng men tar längre tid att stabilisera sig. Insvängningsförloppet i början är dock identiskt med det för dubbeltanken.

Här dras också gränsen, vid högre dödtid uppstår instabilitet, framförallt vid insvängningen men också vid störningarna som inte alls kan hanteras. Systemet är dock stabilt vid dödtid på 5,0 sekunder. Jämfört med den vanliga dubbeltanksregleringen med samma dödtid kan man se att den dynamiska störningslogiken orsakar större översväng men ger en snabbare återhämtning efter störningar. Vilken metod man ska använda beror helt på vilka egenskaper som är önskade hos det slutgiltiga systemet. Eftersom regulatorn är baserad på den vanliga dubbeltanksregulatorn hanteras dödtid på precis samma sätt, vilket innebär att en bättre reglering skulle kunna uppnås om regulatorn från början hade dimensionerats för dödtid.

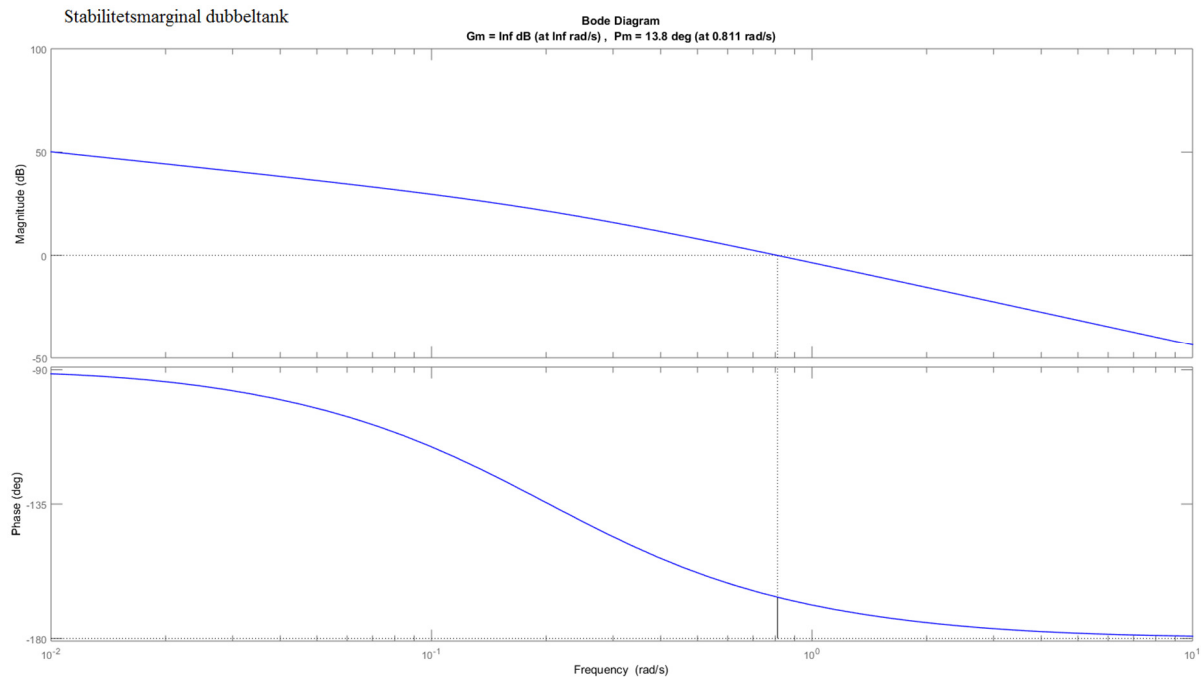
4.8 Stabilitetsmarginaler för de olika regulatorerna

Stabilitetsmarginaler hos de olika systemen presenteras här för att ge en återkoppling till de resultat som uppnått i de olika försök som har genomförts.



Figur 31. Stabilitetsmarginaler för enkeltank.

För enkeltanken blir stabiliteten hyfsad (se figur 31), med tanke på att ett så lågt lambda-värde har används för att dimensionera regulatorn. Fasmarginalen som är 14,4 grader, är en bit under för vad som brukar önskas av industriella reglersystem, det som eftersträvas av detta reglersystem är dock en balans mellan snabbhet och stabilitet. Tanksystemets överföringsfunktion har approximerats till en första ordningens överföringsfunktion för att lambda-metoden skulle kunna appliceras, detta bidrar till resultatet. Marginalerna är dock tillräckliga för att ge ett system som fungerar bra under normala driftförhållanden samt hanterar störningar bra.



Figur 32. Stabilitetsmarginaler för dubbeltank.

För dubbeltanken blir stabiliteten något sämre än för enkeltanken (se figur 32), vilket inte heller var helt oväntat. Även här har samma låga lambda-värde används för dimensioneringen. Tankens överföringsfunktion har också som tidigare nämnts approximerats som en första ordningens för att lambda-metoden skall kunna appliceras, detta bidrar självklart en hel del till resultatet. Fasmarginalen som är 13,8 grader är även här låg, men den är tillräcklig för att ge ett snabbt system som kan fungera under normala driftförhållanden utan problem. Systemen klarar som tidigare visat även störningar av olika storlek.

Stabilitetsmarginaler för framkopplingen och kaskadregleringen presenteras ej här eftersom de regulatorerna till stor del eller fullt ut bygger på regulatorerna för enkel och dubbeltank. Resultatet skulle med andra ord bli näst intill identiskt eller extremt likt.

5. RESULTAT OCH DISKUSSION

I projektet har flera olika reglerstrukturer analyserats. Dessutom har möjligheten att kontrollera digitala utgångar samtidigt som tank-systemet styrs, testats i form av lampor som tänds och släcks samt magnetventiler som öppnas och stängs. Implementering av de olika funktionerna har skett i strukturerad text vilket medför att det har varit enkelt att utföra ändringar och tillägg under arbetets gång. Många sidofunktioner och förbättringar av tidigare skapade funktioner har växt fram, tillsammans har det bidragit till ett väldigt flexibelt system. Den största fördelen med att koda funktionerna i strukturerad text är att det blir väldigt lätt även för den som aldrig har jobbat med PLC innan att komma in i systemet och skapa funktioner och kod.

Även om strukturerad text skiljer sig en del från ett programspråk som C finns det många likheter samt väldigt många funktioner att välja på. Det är också användningen av strukturerad text som möjliggör det för PLC-systemet att utföra samma uppgifter som en vanlig regulator. Även om det går att få liknande resultat genom att använda färdiga block i andra PLC-programmeringsspråk kvarstår dock den stora fördelen med att i detalj bygga upp regulatorn och därmed lätt få in önskade funktioner utan problem. Dessutom är det möjligt att packa koden till ett funktionsblock när programutvecklingen i structured text är slutförd, om så skulle önskas vilket är ytterligare ett starkt argument för att koda i structured text.

Att kunna binda samman reglertekniken med styrtekniken och automation på detta sätt är unikt och bidrar till att studenter som arbetar med den typ av system snabbt får en förståelse för vad reglerteknik och automation är, detta visar framförallt på den stora potentialen hos PLC-styrda reglersystem i utbildningssyfte.

Under projektets gång har de problem och utmaningar som stötts på dokumenterats mycket grundligt för att underlätta för framtida användare och eller vidareutveckling av systemet. Särskilda dokument har också tagits fram för att underlätta konfigurerings av systemet samt förslag på hur en labb kan se ut (se bilaga B och bilaga D).

Resultaten av de tester som har utförts kan också användas i framtida laborationer för att visa på hur stor påverkan vissa reglertekniska funktioner har på regleringen, ett exempel på detta är användandet av anti wind-up jämfört med ingen anti wind-up. Det program som har skapats som genererar slumpade störningar kan direkt användas i en laboration för att ge en realistisk testmiljö. Det medför att studenterna får en mer realistisk feedback på hur deras regulatorer fungerar. Slutligen har de erfarenheter som projektets deltagare samlat på sig under projektets gång bidragit till att ett dokument har skapats som skall vara till stöd för upprättandet av framtida laborationer (se bilaga D).

Överlag är resultatet mycket tillfredställande för både dubbel och enkeltank. Både snabb och stabil reglering har tagits fram och testats i detalj med olika grader av störningar och dödtid. Stabilitetsmarginalerna är dock för de olika regulatorerna väldigt låga, det beror som tidigare nämnt på att låga lambda-värden har använts då regulatorerna tagits fram. Användandet av lambda-metoden för att kontrollera en process av andra ordningen är också en förenkling, eftersom processen då approximeras med en första ordningens process. Resultatet blev tillräckligt bra, även om stabilitetsmarginalerna inte är imponerande. Om man önskar att undvika detta kan en annan metod än lambda-metoden användas för att ta fram regulatorn för tvåtanksystemet.

Slutsatsen som kan dras av resultaten är att det går bra att låta studenter genomföra laborationer med en liknande utrustning som har används i projektet. PLC:n är som visat snabb nog för att möjliggöra styrning av tank-processen på många olika sätt enligt många olika reglerprinciper. Samtidigt som reglering av processen utförs kan styrning av digitala komponenter utföras utan att detta ger någon negativ påverkan av regleringen som pågår parallellt. Detta har bevisats genom styrningen av magnetventiler som har utförts under pågående reglering.

De olika reglerprinciperna har visat olika styrkor och svagheter, därför är det omöjligt att direkt avgöra vilken som är bäst. Det beror väldigt mycket på vilket tillämpning det är samt vilka egenskaper som anses vara viktigast i systemet. Under arbetets gång har regulatorns snabbhet haft en väldigt hög prioritet, samt förmågan att hantera processtörningar. Dock är det värt att nämna att regleringen med framkoppling är den som har störst utvecklingspotential, eftersom störningarna i det aktuella systemet är mätbara. Om mer tid läggs på att ta fram en algoritm för hur regulatorn ska kompensera vid störningar kan teoretiskt en näst intill perfekt reglering tas fram.

REFERENSER

- [1] B. Thomas: *Modern Reglerteknik*, 4:e uppl., Stockholm, Sverige: Liber, 2013
- [2] Remote I/O NA-9372, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,
http://www.crevis.co.kr/eng/product/view01.php?str_bcode=030120001&str_no=383
(acc 2016-05-12)
- [3] Remote I/O ST-1218, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,
http://www.crevis.co.kr/eng/product/view01.php?str_bcode=030120002&str_no=50
(acc 2016-05-12)
- [4] FnIO S-Series Digital Input ST-1214, ST-1218, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,
http://ftc.beijer.se/files/C125728B003AF839/5524E3ADE47C03D7C1257C2100487DC1/ST-1214_ST-1218_201605BE.pdf
(acc 2016-05-12)
- [5] Remote I/O ST-2328, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,
http://www.crevis.co.kr/eng/product/view01.php?str_bcode=030120003&str_no=71
(acc 2016-05-12)
- [6] FnIO S-Series Digital Output ST-2324, ST-2328, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,
http://ftc.beijer.se/files/C125728B003AF839/7B715B79E6282EA2C1257C2100501BA1/ST-2324_ST-2328_201605BE.pdf
(acc 2016-05-12)
- [7] Remote I/O ST-4422, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,
http://www.crevis.co.kr/eng/product/view01.php?str_bcode=030120005&str_no=114
(acc 2016-05-13)
- [8] FnIO S-Series Analog Output/ST-4422, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,
http://ftc.beijer.se/files/C125728B003AF839/D09DE04E99BB4179C1257C220032882F/st_4422.pdf
(acc 2016-05-13)
- [9] Remote I/O ST-3424, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,
http://www.crevis.co.kr/eng/product/view01.php?str_bcode=030120004&str_no=91
(acc 2016-05-13)
- [10] FnIO S-Series Analog Input/ST-3424, Crevis CO, 29-4, Gigok-Ro, Gihueng-Gu, Yong-in, Kyounggi, KOREA 2014, Produktdatablad,

http://ftc.beijer.se/files/C125728B003AF839/963ED1FDC5CCB987C1257C22002BA406/st_3424.pdf
(acc 2016-05-13)

BILAGOR

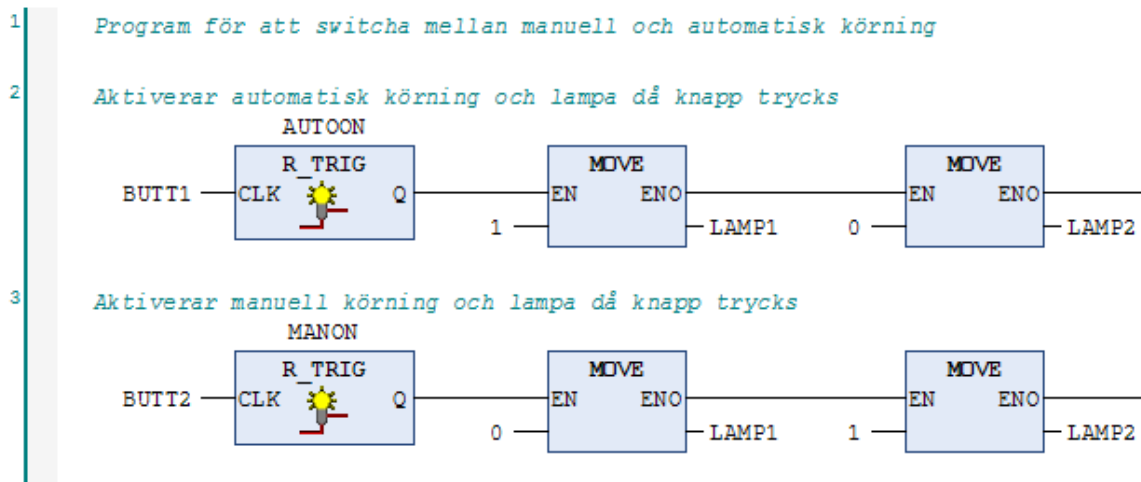
A. Kod i CoDeSys

Globala Variabel Lista

```
1 VAR_GLOBAL
2 SUPVAL1: WORD; //Börvärde tank 1
3 ACTVAL1: WORD; //Ärvärde tank 1
4 ACTVAL2: WORD; //Ärvärde tank 2
5 CONVAL1: WORD; //Styrvärde tank 1
6 //Ventiler
7 VENT1: BOOL; //Magnetventil 1
8 VENT2: BOOL; //Magnetventil 2
9 VENT3: BOOL; //Magnetventil 3
10 //Lampor
11 LAMP1: BOOL; //Lampa för automatisk körning
12 LAMP2: BOOL; //Lampa för manuell körning
13 //Knappar
14 BUTT1: BOOL; //Knapp för automatisk körning
15 BUTT2: BOOL; //Knapp för manuell körning
16 END_VAR
```

Manuell-Auto Switch

```
1 PROGRAM MANAUTO
2 VAR
3 AUTOON: R_TRIG;
4 MANON: R_TRIG;
5 END_VAR
```

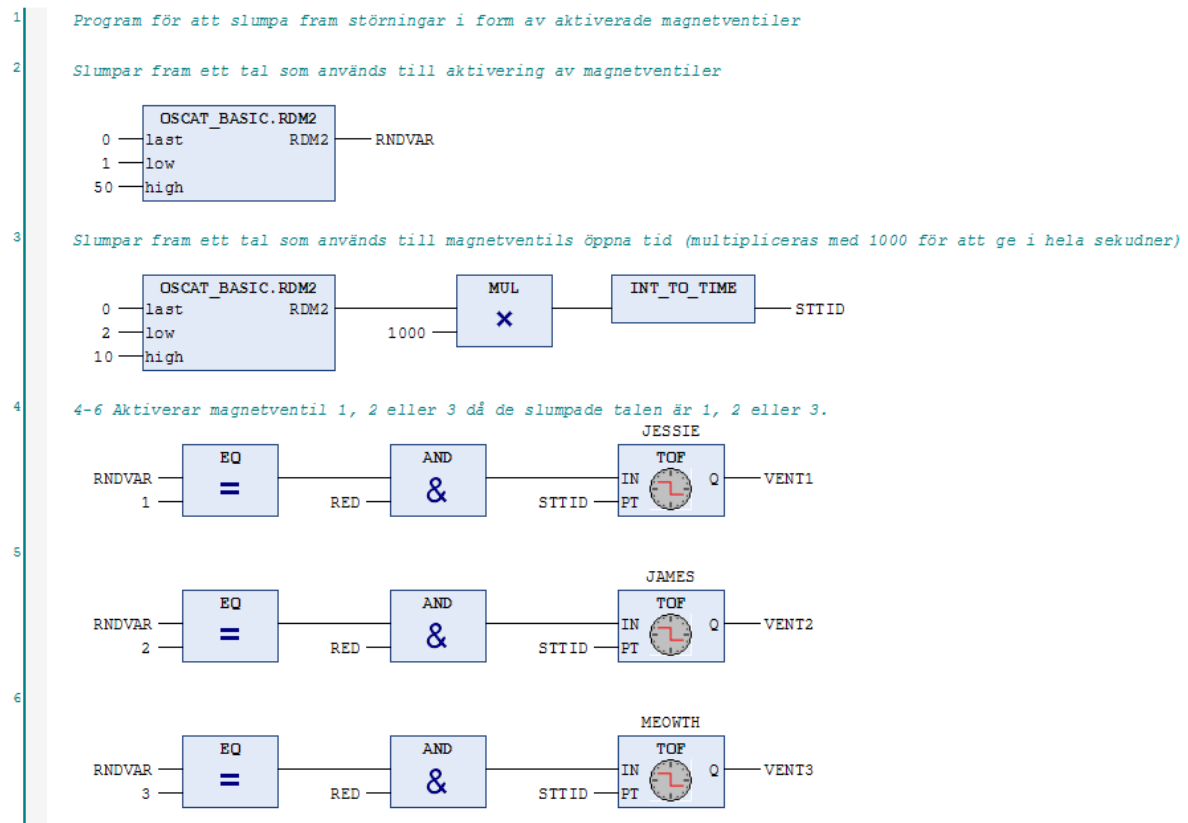


Slumpade Störningar

```

1 PROGRAM RNDVENT
2 VAR
3 RNDVAR: INT; //Variabel för slumpad tal som används till ventilernas aktivering
4 JESSIE: TOF;
5 JAMES: TOF;
6 MEOWTH: TOF;
7 STTID: TIME; //Variabel för ventilernas öppettid
8 RED: BOOL; //Variabel för att aktivera slumpade störningar
9 END_VAR

```



Enkeltanksreglering

```
1 PROGRAM PIDET
2 VAR
3 w: REAL; //Aktuell integralsumma
4 wo: REAL; //Ett sampel gammal integralsumma
5 u: REAL; //Styrsignal
6 e: INT; //Aktuell avvikelse
7 eo: INT; //Ett sampel gammal avvikelse
8 h: REAL; //Samplingsintervall
9 P: REAL; //P-verkan
10 I: REAL; //I-verkan
11 D: REAL; //D-verkan
12 uN: REAL; //Styrsignal mellanvariabel för Integral Wind-Up
13 K: REAL; //Förstärkning
14 TI: REAL; //Integraltid
15 TD: REAL; //Derivatatid
16 TF: REAL; //Derivata-filter konstant
17 END_VAR

1 //Reglering av enkeltank
2
3 //PID-regulator automatisk
4 IF LAMP1=TRUE THEN
5
6 h:=0.0255;
7 K:=2.022;
8 TI:=32;
9 TD:=0.35;
10 TF:=0.07;
11 e:=(SUPVAL1-ACTVAL1); //Beräknar avvikelsen mellan bör- och ärvärdet
12
13 //Integral Wind-Up
14 IF u>=4095 THEN
15
16 un:=4095;
17 w:=ABS( ((un-P-D)*TI) / (K*h) ); //Beräknar integralsumma för Integral Wind-Up
18
19 ELSE
20
21 w:=wo+e; //Beräknar nuvarande integralsumma
22
23 END_IF
24
25 P:=(K*e); //Beräknar P-verkan
26 I:=(K*(h/TI)*w); //Beräknar I-verkan
27 D:=(K*(TD*(e-eo)) / (h+TF*(e-eo))); //Beräknar D-verkan
28 eo:=e; //Sätter gammal avvikelse
29 wo:=w; //Sätter gammal integralsumma
30 u:=(P+I+D); //Beräknar styrsignalen
31 CONVAL1:=REAL_TO_WORD(u); //Skriverutstyrsignal till pumpen.
32
33 //PID-regulator manuell
34 ELSIF LAMP2=TRUE THEN
35
36 CONVAL1:=SUPVAL1; //Skriver direkt till pumpen från potentiometern
37
38 END_IF
```

Dubbeltanksreglering

```
1 PROGRAM PIDDBT
2 VAR
3 w: REAL; //Aktuell integralsumma
4 wo: REAL; //Ett sampel gammal integralsumma
5 u: REAL; //Styrsignal
6 e: INT; //Aktuell avvikelse
7 eo: INT; //Ett sampel gammal avvikelse
8 h: REAL; //Samplingsintervall
9 P: REAL; //P-verkan
10 I: REAL; //I-verkan
11 D: REAL; //D-verkan
12 uN: REAL; //Styrsignal mellanvariabel för Integral Wind-Up
13 K: REAL; //Förstärkning
14 TI: REAL; //Integraltid
15 TD: REAL; //Derivatatid
16 TF: REAL; //Derivata-filter konstant
17 END_VAR

1 //Reglering av dubbeltank
2
3 //PID-regulator automatisk
4 IF LAMP1=TRUE THEN
5
6 h:=0.0255;
7 K:=2.067;
8 TI:=60.4;
9 TD:=1.35;
10 TF:=0.27;
11 e:=(SUPVAL1-ACTVAL2); //Beräknar avvikelsen mellan bör- och ärvärdet
12
13 //Integral Wind-Up
14 IF u>=4095 THEN
15
16 un:=4095;
17 w:=ABS( ((un-P-D)*TI) / (K*h) ); //Beräknar integralsumma för Integral Wind-Up
18
19 ELSE
20
21 w:=wo+e; //Beräknar nuvarande integralsumma
22
23 END_IF
24
25 P:=(K*e); //Beräknar P-verkan
26 I:=(K*(h/TI)*w); //Beräknar I-verkan
27 D:=(K*(TD*(e-eo))/(h+TF*(e-eo))); //Beräknar D-verkan
28 eo:=e; //Sätter gammal avvikelse
29 wo:=w; //Sätter gammal integralsumma
30 u:=(P+I+D); //Beräknar styrsignalen
31 CONVAL1:=REAL_TO_WORD(u); //Skriverutstyrsignal till pumpen.
32
33 //PID-regulator manuell
34 ELSIF LAMP2=TRUE THEN
35
36 CONVAL1:=SUPVAL1; //Skriver direkt till pumpen från potentiometern
37
38 END_IF
```

Kaskadreglering

```
1 PROGRAM KSKDB
2 VAR
```

```

3 h: REAL; //Samplingsintervall
4 s2: WORD; //Styrsignal mellanvariabel för kaskadreglering
5 //Variabler tank 2
6 u2: REAL; //Styrsignal tank 2
7 un2: REAL; //Styrsignal mellanvariabel för Integral Wind-Up tank 2
8 P2: REAL; //P-verkan tank 2
9 I2: REAL; //I-verkan tank 2
10 D2: REAL; //D-verkan tank 2
11 w2: REAL; //Aktuell integralsumma tank 2
12 wo2: REAL; //Ett sampel gammal integralsumma tank 2
13 e2: INT; //Aktuell avvikelse tank 2
14 eo2: INT; //Ett sampel gammal avvikelse tank 2
15 K2: REAL; //Förstärkning tank 2
16 TI2: REAL; //Integraltid tank 2
17 TD2: REAL; //Derivatatid tank 2
18 TF2: REAL; //Derivata-filter konstant tank 2
19 //Variabler tank 1
20 u1: REAL; //Styrsignal tank 1
21 un1: REAL; //Styrsignal mellanvariabel för Integral Wind-Up tank 1
22 P1: REAL; //P-verkan tank 1
23 I1: REAL; //I-verkan tank 1
24 D1: REAL; //D-verkan tank 1
25 w1: REAL; //Aktuell integralsumma tank 1
26 wo1: REAL; //Ett sampel gammal integralsumma tank 1
27 e1: INT; //Aktuell avvikelse tank 1
28 eo1: INT; //Ett sampel gammal avvikelse tank 1
29 K1: REAL; //Förstärkning tank 1
30 TI1: REAL; //Integraltid tank 1
31 TD1: REAL; //Derivatatid tank 1
32 TF1: REAL; //derivata-filter konstant tank 1
33 END_VAR

1 //Kaskadreglering
2
3 //PID-regulator automatisk
4 IF LAMP1=TRUE THEN
5
6 h:=0.0255;
7 K1:=2.022;
8 K2:=2.067;
9 TI1:=32;
10 TI2:=60.4;
11 TD1:=0.35;
12 TD2:=1.35;
13 TF1:=0.07;
14 TF2:=0.27;
15 e2:=(SUPVAL1-ACTVAL2); //Beräkna avvikelse mellan börvärde och ärvärde för tank 2
16
17 //Integral Wind-Up tank 2
18 IF u2>=4095 THEN
19
20 un2:=4095 ;
21 w2:=ABS( ((un2-P2-D2)*TI2)/(K2*h) ); //Beräknar integralsumma för Integral Wind-Up
tank 2
22
23 ELSE
24
25 w2:=wo2+e2; //Beräknar nuvarande integralsumma tank 2
26
27 END_IF
28

```

```

29 P2:=(K2*e2); //Beräknar P-verkan tank 2
30 I2:=(K2*(h/TI2)*w2); //Beräknar I-verkan tank 2
31 D2:=(K2*(TD2*(e2-eo2))/(h+TF2*(e2-eo2))); //Beräknar D-verkan tank 2
32 eo2:=e2; //Sätter gammal avvikelse tank 2
33 wo2:=w2; //Sätter gammal integralsumma tank 2
34 u2:=(P2+I2+D2); //Beräknar styrsignalen tank 2
35 s2:=REAL_TO_WORD(u2); //Styrsignal type-conversion för kaskadreglering
36
37 e1:=(s2-ACTVAL1); //Beräkna avvikelse mellan styrsignal tank 2 och ärvärde tank 1
38
39 //Integral Wind-Up tank 1
40 IF u1>=4095 THEN
41
42 un1:=4095 ;
43 w1:=ABS(((un1-P1-D1)*TI1)/(K1*h)); //Beräknar integralsumma för Integral Wind-Up
tank 1
44
45 ELSE
46
47 w1:=w1+e1; //Beräknar nuvarande integralsumma tank 1
48
49 END_IF
50
51 P1:=(K1*e1); //Beräknar P-verkan tank 1
52 I1:=(K1*(h/TI1)*w1); //Beräknar I-verkan tank 1
53 D1:=(K1*(TD1*(e1-eo1))/(h+TF1*(e1-eo1))); //Beräknar D-verkan tank 1
54 eo1:=e1; //Sätter gammal avvikelse tank 1
55 wo1:=w1; //Sätter gammal integralsumma tank 1
56 u1:=(P1+I1+D1); //Beräknar styrsignalen tank 1
57 CONVAL1:=REAL_TO_WORD(u1); //Skriverutstyrsignal till pump
58
59 //PID-regulator manuell
60 ELSIF LAMP2=TRUE THEN
61
62 CONVAL1:=SUPVAL1; //Skriver direkt till pumpen från potentiometern
63
64 END_IF

```

Dynamisk störningslogik

```
1 PROGRAM FKDBT
2 VAR
3 w:REAL; //Aktuell integralsumma
4 wo:REAL; //Ett sampel gammal integralsumma
5 u:REAL; //Styrsignal
6 e:INT; //Aktuell avvikelse
7 eo:INT; //Ett sampel gammal avvikelse
8 h:REAL; //Samplingsintervall
9 P:REAL; //P-verkan
10 I:REAL; //I-verkan
11 D:REAL; //D-verkan
12 uN:REAL; //Styrsignal mellanvariabel för Integral Wind-Up
13 K:REAL; //Förstärkning
14 TI:REAL; //Integraltid
15 TD:REAL; //Derivatid
16 TF:REAL; //Derivata-filter konstant
17 END_VAR

1 //Störningslogik reglering
2
3 //PID-regulator automatisk
4 IF LAMP1=TRUE THEN
5
6 h:=0.0255;
7 TI:=60.4;
8 TD:=1.35;
9 TF:=0.27;
10 e:=(SUPVAL1-ACTVAL2); //Beräknar avvikelsen mellan bör- och ärvärdet
11
12 //Ökad förstärkning vid störning
13 IF VENT2=1 OR VENT1=1 THEN
14
15 K:=2.5; //Ny förstärkning
16
17 ELSE
18
19 K:=2.067; //Normal förstärkning
20
21 END_IF
22
23 //Integral Wind-Up
24 IF u>=4095 THEN
25
26 un:=4095;
27 w:=ABS((un-P-D)*TI)/(K*h); //Beräknar integralsumma för Integral Wind-Up
28
29 ELSE
30
31 w:=wo+e; //Beräknar nuvarande integralsumma
32
33 END_IF
34
35 P:=(K*e); //Beräknar P-verkan
36 I:=(K*(h/TI)*w); //Beräknar I-verkan
37 D:=(K*(TD*(e-eo))/(h+TF*(e-eo))); //Beräknar D-verkan
38 eo:=e; //Sätter gammal avvikelse
39 wo:=w; //Sätter gammal integralsumma
40 u:=(P+I+D); //Beräknar styrsignalen
41 CONVAL1:=REAL_TO_WORD(u); //Skriverutstyrtsignal till pumpen.
42
```

```
43 //PID-regulator manuell
44 ELSIF LAMP2=TRUE THEN
45
46 CONVAL1:=SUPVAL1; //Skriver direkt till pumpen från potentiometern
47
48 END_IF
```

B. Konfigurering

B.1 Konfigurering av PLC och CoDeSys

1. Öppna CoDeSys V3.5 SP3 Patch 1
2. Välj New Project – Standard Project och ge ett namn samt plats på datorn där projektet sparas.
3. Välj CODESYS Control Win V3 under ”Device” för tillfället och välj valfritt språk under ”PLC_PRG in”.
4. Välj Tools – Options – Features – Predefined Feature Sets och ändra från ”Standard” till ”Professional”.
5. Välj Tools – Device Repository – Install – välj aktuella xml-filer.
6. Högerklicka på Device (CODESYS Control Win V3) och välj ”Update Device”. Välj PIO beroende på vad som används.
7. Högerklicka på en ”<Empty>(<Empty>)” och välj ”Plug Device”. Välj sedan modulerna som används. EXTREMT VIKTIGT! Lägg till modulerna i ordningen som de sitter fysiskt (vänster till höger), annars funkar det inte att starta (RUN på PLC:n blinkar rött efter login).
8. Dubbelklicka på ”Device (CVS NA-93xx(PIO))”. Under ”Communication Settings” välj Add Gateway – OK. Markera Gateway-1 och välj ”Scan Network” och dubbelklicka på rätt PIO för att sätta den som ”Active”.

B.2 Lägg till bibliotek i CoDeSys

1. Välj Tools – Package Manager – Install och välj ”OSCATBasic”.
2. Välj Tools – Library Repository – Install och välj ”oscat_basic_333”.
3. Dubbelklicka på ”Library Repository” i vänstra spalten under ”Application” och välj ”Add library”. Välj sedan Application – Utils – BASIC.

B.3 Infoga block från OSCAT Basic (FBD)

1. Högerklicka i ett ”Network” i ett öppet FBD.
2. Välj ”Insert Box”.
3. Klicka sedan på ”Text Search” och skriv in namnet på block som önskas.

C. Problem som kan uppstå

PID-regulatorn fastnar då utsignalen från tanken går över 10V och fortsätter öka med det dåvarande styrvärdet utan att reglera sig.

Detta beror på att modulen ST-3424 bara kan ta en inspänning på 10V och tanken ger ut knappt över detta vid 100 %.

Spänningsdelning av insignalen löste problemet, men processen blev något mer instabil och en faktor måste multipliceras med är-värdet för att ge korrekt reglering. Denna faktor är beroende av tankens egenskaper, dvs. något som måste ställas in för varje tank-system.

En annan, mycket enklare och bättre, lösning är att man bara får köra till 90% under labben, detta löser problemet eftersom utspänningen då aldrig överstiger 10V.

D. Förslag inför labb

Eftersom endast en tank skall regleras, och det inte finns tid för studenterna att konfigurera upp PLC:t mm ges följande förslag:

Fysisk utrustning som behövs:

- 2 st Tryckknappar (för att kunna växla mellan manuell och automatisk körning).
- 2 st lampor för att indikera vilket läge systemet är i.
- Potentiometer för att ställa in börvärde/kontrollera pumpens varvtal, eventuellt två potentiometrar med endast en funktion var.
- Eventuellt ett nödstopp.
- A/D låda.

Förslag på uppgifter:

En bra början är att studenterna kopplar upp knappar, potentiometrar mm till de fysiska utgångarna på PLC:t. Eventuellt kan studenterna till en början använda funktionsblocket OSCAT_BASIC.CTRL_PID för att komma in i uppgiften snabbt.

Efter stegsvarsanalys kan de ta fram lämpliga parametrar och ställa in regulatorn med valfri metod. Detta bör ta mellan 1 och 1½ timma.

Därefter är det dags för studenterna att själva ta fram en regulator i structured text, och jämföra resultatet med den tidigare regulatorn (de behöver inte nödvändigtvis ta fram en tidsdiskret PID-regulator). Eventuellt kan här frågeställningar som anti wind-up behandlas, om studenterna har fått lite information om dessa begrepp tidigare.

Om magnetventilerna skall användas kan den funktion för slumpade störningar som vi har tagit fram användas.