

MALMÖ HÖGSKOLA

Inbyggda system och signaler Styr- och reglerteknik

Labbinlämning 1504e

Utlämning: 20 febr 2017
Deadline inlämning: 14 mars 2017, kl.

16:00

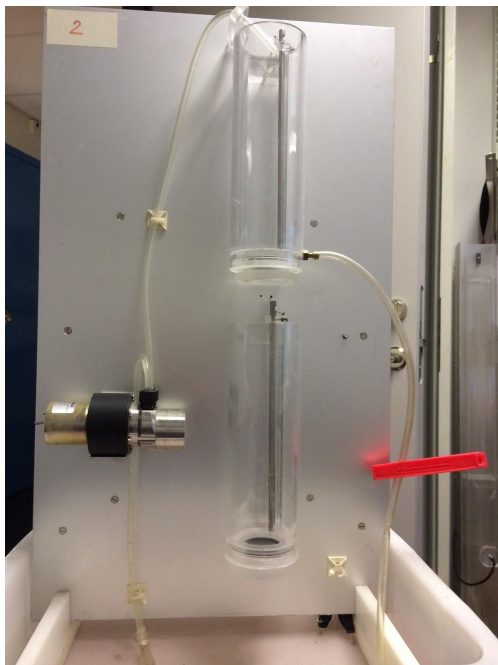
Namn: Leonard Holgersson

Namn: Hadi Deknache

Klassisk reglerteknik och bestämning av tidsdiskreta överföringsfunktioner

Syftet med denna laboration är att praktiskt tillämpa olika klassiska regleralgoritmer på en klassisk vattentankmodell. Genom att få stegsvaret på det fysikaliska systemet kan man använda tumregler för inställningen av regulatorparameter samt genomföra en "black-box" systemidentifikation.

Vi använder oss av en vattentankmodell med två behållare där vattnet pumpas in i den första tank och därifrån rinner genom ett hål i botten in i den andra tanken, se bild 1.



Samma typ av vattenmodell är en klassisk process som används flitigt inom utbildning och forskning inom reglerteknik.

I "Kompendium om reglerteknikutrustningen" på its learning finns en ingående beskrivning av vattentankmodellen.

Bild 1: Kort av vattentankmodellen som används i denna inlämningsuppgift.

I dokumentet "Överblick över reglerteknikdelen" visas vilka vattentankar (övre eller undre) som ska användas till vilka uppgifter. Där finns också referenser till kursboken till varje uppgift. Det är upp till varje grupp hur de planerar genomförandet av uppgifterna. Antal vattentankmodeller är begränsat och man behöver boka tider för att genomföra de praktiska experimenten. Det kräver en viss förberedning för att hinna färdig i tid. Uppgifterna delas upp i olika delar "A-D" : teoretiska förberedande inkl. Matlabprogrammering (A), praktiska och experimentella (B), analyserande (C), och självreflekterande (D) delar.

Innehållsförteckning och översikt

A.1 Vattentankmodell som enkel reglerkrets	6
A.1.1 Rita upp den klassiska reglerkretsen för vattentankmodellen som ska regleras med	6
A.1.2 Analysera processens (vattentankmodellens) in- och utgångssignalerna.	6
A.1.3 Exempel av tillämpningsområden som kan beskrivas med vattentankmodellen	6
B.1 Labbutrustningen och allmänna anvisningar beträffande experimentens genomförande	7
B.1.1 Förberedning av Arduino Due	7
B.1.2 Anslutningen av vattentankmodellen till Arduino Due	8
B.1.2 Test av utgångsspänning mellan 0V-ca.2,5V	8
A.2 Stegsvär, processtyp och typsiffra	8
A.2.1 Stegsvär i öppna regelkretsen	9
A.2.2 Processtypen och typsiffror för nivån i första eller andra behållaren	9
A.2.3 Programmeringen av stegsvarexperiment	9
B.2 Stegsvarexperiment	11
B.2.1. Stegsvarexperiment	12
C.1 Jämförelse av stegsvaren	12
C.1.2 Jämförelse mellan ritning och resultat	12
C.2 Filtrering av mätvärden	12
C.2.1 Filtrera mätningarna från stegsvaren, spara dem i globala variabler och klistra in grafen (plots) här:	13
A.3 Klassiska reglerprinciper	13
A.3.1 Programmering av en tidsdiskret tvålägesreglering	14
A.3.2 Programmering av en tidsdiskret flerstegsreglering	14
A.3.3 Programmering av en tidsdiskret P-reglering	14
A.3.4 Programmering av en tidsdiskret PID-reglering	14
A.3.5 Programmering av en tidsdiskret PD-reglering	14
B.3 Reglering med tidsdiskreta klassiska reglerprinciper	15
B.3.1. Kopiera in grafen från "tvålägesregleringen" här:	16
B.3.2 Kopiera in grafen från "flerstegsregleringen" här:	16
B.3.3 Kopiera in grafen från "P-regleringen" här:	16
B.3.4 Kopiera in grafen från "PID-regleringen" här:	16
B.3.5. Kopiera in grafen från "PD-regleringen" här:	16
C.3 Dimensionering av reglersystem med tumregelmetoder	16

C.3.1 Lambda- och Amigo-metoden	16
C.3.1.1 Vilka värden får ni från Lambda-metoden för:	16
C.3.1.2 Vilka värden får ni från Amigo-metoden för:	16
B.4. Ziegler-Nichols svängningsmetod	16
B.4.1 Klistra in en graf som visar hur ni fick nivån i undre vattentanken att självsvänga enligt svängningsmetoden:	16
B.4.2 Vilka kritiska förstärkning K_0 och periodtid T_0 får ni?	16
B.4.3 Vilka blir PID-regulatorparametrarna enligt Ziegler-Nichols tabell?	17
B.4.4 Vilka blir PID-regulatorparametrarna enligt Åström & Hägglunds föreslag?	17
B.5 PID-regulator med olika parametrar enligt tumregler	17
B.5.1 Kopiera in grafen från PID-regulatorn med Amigo-metoden-parametrarna	17
B.5.2 Kopiera in grafen från PID-regulatorn med Ziegler-Nichols-metoden-parametrarna	17
C.4 Jämförelse mellan olika reglersystem	17
C.4.1 Jämförelse av systemens egenskaper. Fyll i tabellen:	17
A.4 Bestämning av tidsdiskreta överföringsfunktionen	18
A.4.1 Beskriv och visa steg för steg hur ni bestämmer överföringsfunktionen för den övre	18
A.4.2 Beskriv och visa steg för steg hur ni bestämmer överföringsfunktionen för den övre och	18
A.5 Simulation av stegsvaren med Matlab Control-Toolbox	18
A.5.1 Simulera stegsvaren av övre vattentank med Matlab och spara resultatet i en variabel.	18
A.5.2 Simulera stegsvaren av övre och nedre vattentank tillsammans med Matlab och spara	18
C.5 Jämförelse mellan verkliga och simulerade stegsvaren	18
C.5.1 Jämförelse mellan verkliga och simulerade stegsvaren för övre vattentank. Klistra in	18
C.5.2 Jämförelse mellan verkliga och simulerade stegsvaren för övre och nedre vattentank	18
D. Reflektion och utvärdering över det egna lärandet	19
Bilaga	20
Översikt över Matlab instruktioner	20
Exempel av nyckelfrågor i samband med reflektioner och utvärdering av det eget lärande	20

Inlämningen av detta fullständigt ifyllda dokument samt andra filer som ni ska generera för att dokumentera vissa delar av er lösning ska ske på its learning. **Ladda upp varje fil för sig, dvs inte komprimerade.** För videodokumentering kan länkar anges t.ex. till youtube eller andra lämpliga videotjänster.

Laborationen genomförs som vanligt i par dvs. ni jobbar två och två eller ensam. Vid inlämningen på Its learning anges vem som jobbat ihop. Forskningen visar att den mest effektiva inläringen sker när man förklarar något till någon annan! Tillämpa det gärna på varandra i gruppen och i hela klassen för att få hjälp i att förstå vad som ska göras och varför. Själva laborationen blir dock meningslös om ni fuskar och bara kopierar varandras resultat eller formuleringar utan att själv har förstått vad ni skriver! ***Alla svar och alla programkod och mätresultat ska vara gruppens egen!!*** Labbinlämningsuppgifterna dokumenterar er inläring i ämnet och om de genomförs seriöst har man uppnått lärandemålen och kommer att klara sluttentamen!

Dokument som ni behöver för att kunna lösa uppgifterna är kursboken "Modern Reglerteknik" av Bertil Thomas, Matlabguiden, Matlabs "help" och dokumentation samt material som finns upplagda på its learning.

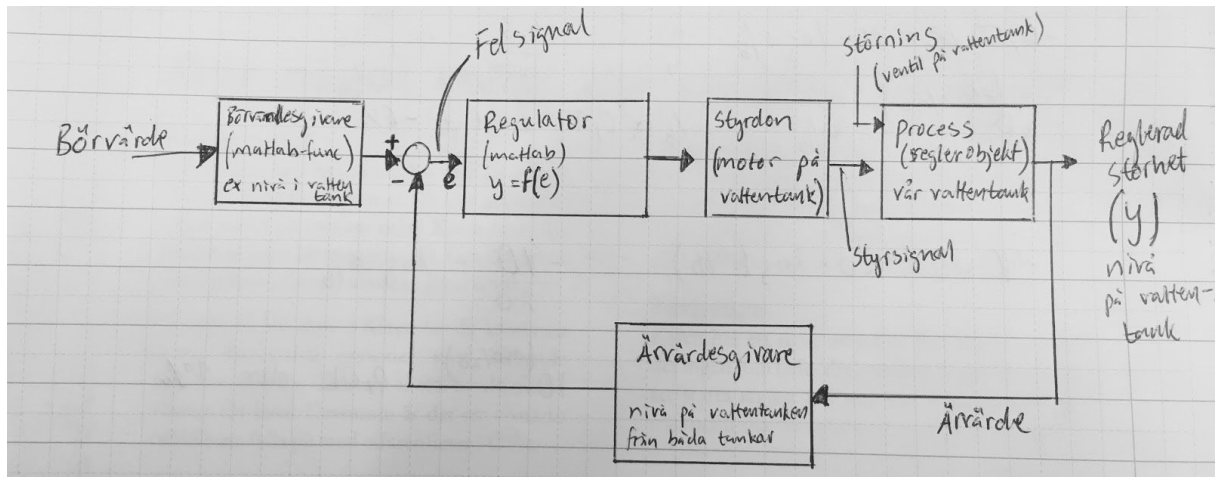
Krav för godkänd

- Fullständigt ifyllt dokument (inkl namn på titelsida) med korrekta svar till alla frågor, uppladdad till its learning som word eller pdf-fil, (okomprimerad).
- Binär fil "labb1504e.mat" med alla variabler från reglerexperimenten, uppladdad på its learning.
-

A.1 Vattentankmodell som enkel reglerkrets

Börja med att läsa igenom "Kompendium om reglerteknikutrustningen", åtminstone kap. 1 & 2. Du behöver få en överblick över processen som ska regleras. Gör därför följande uppgifter:

A.1.1 Rita upp den klassiska reglerkretsen för vattentankmodellen som ska regleras med Matlab och Arduino. Ange de olika begrepp till de olika signalerna.



A.1.2 Analysera processens (vattentankmodellens) in- och utgångssignalerna.

a) I vilket spänningsområde befinner sig signalerna?

Spänningen på vattentankens nivåmätare ligger mellan 0-10 V

b) Hur behöver den anpassas för att kunna hanteras av Arduino?

Genom att sätta en spänningsdelare för att skala ner spänningen så insignalen helst ligger under 3 V samt en op amp ut från vattentanken till in till arduino duens A0/A1 ingång, för att skydda ingången på duen.

c) Vad är sambandet mellan den fysikaliska enheten och Arduinos motsvarande digitala representation? vattennivå → 0-10V → 0-1023 i arduino dvs 1024 steg. Vattennivån bestämmer analoga signalen in för hur högt det ligger 0-1023.

A.1.3 Exempel av tillämpningsområden som kan beskrivas med vattentankmodellen

Vattentankmodellen är en mycket användbar modell i helt olika sammanhang. Testa detta genom att ange tre olika exempel, en från området "ekonomi", en från "samhälle" och en från "industri" som kan beskrivas med vattentankmodellen. Beskriv vad i dina exempel som motsvarar motorstyrningen, pumpen, in- och utflöden, störningar, nivån i första och andra behållaren samt börvärden.

1. "Ekonomi"-exempel:
skatter

2. "Samhälle"-exempel:

Arbete

3. "Industri"-exempel: Godisfabrik

Översikt och jämförelse

	Ekonomi-exempel:	Samhälle-exempel :	Industri-exempel:
motorstyrning	Skattereglering	Jobbannonser	Chef säger vad som behöver göras efter kundens förfrågan.
pumpen	Arbetare som genererar pengar till företag.	Kallelse till intervju	Arbetare producerar godis i stora mängder.
inflöde till första behållare	Lön, skatteåterbäring	Andel som intervjuas	Godis till sortering och kvalitetskontroll.
utflöde ur första behållaren	Skatt på lön	Folk som anställs	Sorterat/kontrollerat godis
inflöde till andra behållaren	Skattepengar till staten	Nyanställda på arbetsplats	Nysorterat/-kontrollerat godis
utflöde ur andra behållaren	Statens utgifter	Folk som slutar på arbetsplats	Godis som ska levereras
störningar	Skattefusk	Ingen matchar kriterierna för jobbet	Exempelvis maskinfel
nivån i första behållaren	Privatpersonens kapital	Folk som matchar kriterierna för arbetet.	Godis som ska sorteras
nivån i andra behållaren	Statens kapital	Anställda på arbetsplats.	Färdigsorterat godis som ska levereras till kund
börvärde	Statens önskade kapital	Bestäms av chef på arbetsplats.	Godisproduktion enligt kundens efterfrågan och i fas tidsmässigt.

B.1 Labbutrustningen och allmänna anvisningar beträffande experimentens genomförande

Vi har ställt ihop ett "Kompendium om reglerteknikutrustningen" som finns på its learning. Där beskrivs de olika delarna som ingår i utrustningen. **Vänligen läs igenom kompendiet först, innan ni fortsätter!!** Fig B.1 nedan ger en överblick över de olika delarna som man behöver ha koll på innan man kan påbörja experimenten.

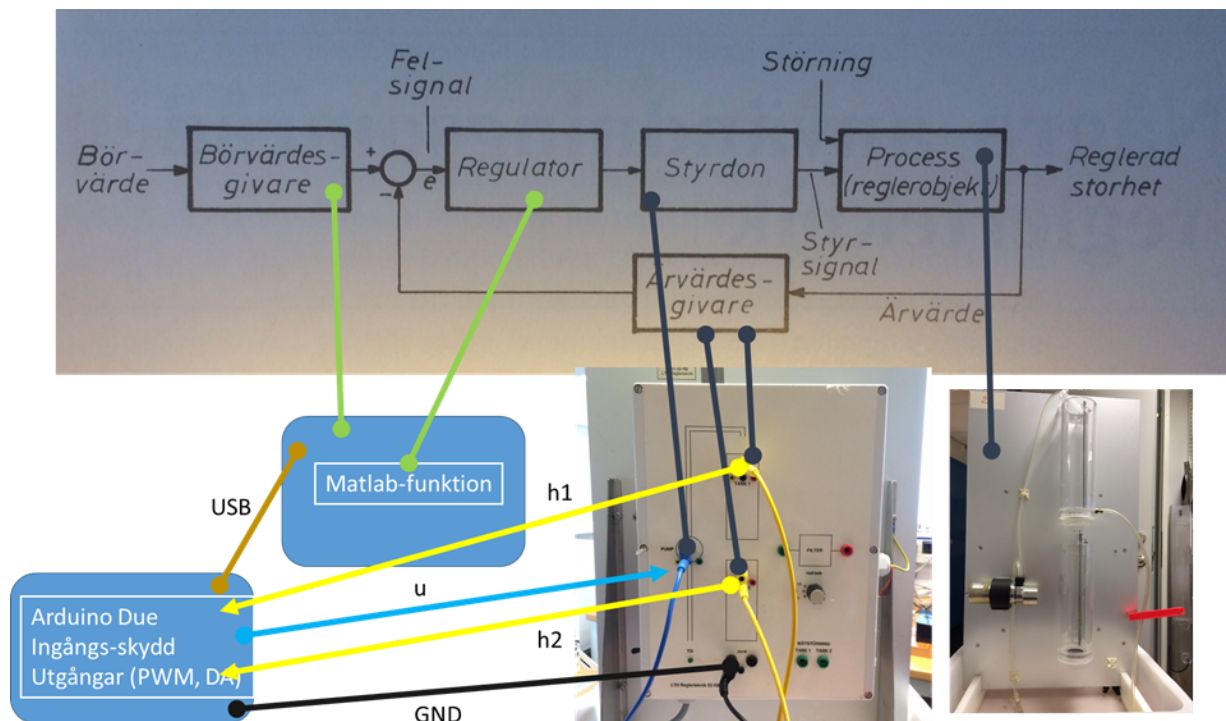


Bild B.1: Översikt över de olika delarna som ingår i utrustningen.

OBS: Det är Matlab 2015b som ska användas!

B.1.1 Förberedning av Arduino Due

Börja med att anpassa arduino due till reglertekniklabben. Detta innebär, att enligt beskrivningen i kompendium:

- Skydda och anpassa analoga ingångarna 'A0' och 'A1', genom elektroniska kretsen med op-amp CA3240, beskriven i kompendiums kap. 2.1.
- Bygga upp en elektronisk krets med en op-amp TS921 för anpassningen av PWM-utgång ('DAC1') och digital/analog utgången ('DAC0'), enligt beskrivningen i kompendiums kap. 2.2.

B.1.2 Anslutningen av vattentankmodellen till Arduino Due

Anslut nu kablarna från vattentankmodellen till Arduino Due:

- i) Jordning från vattentankmodellen till Arduinos GND
- ii) Nivåmätning av övre tanken (h1) till Arduino, t.ex. 'A0'
- iii) Nivåmätning av nedre tanken (h2) till Arduino, t.ex. 'A1'
- iv) Pumpstyrning från Arduino (TS921 eller motor shield) till vattentanken

Om banankontaktanslutningarna har tagit slut igen i labbet så behöver ni eventuellt löda ihop några igen för att underlätta experimenterandet!

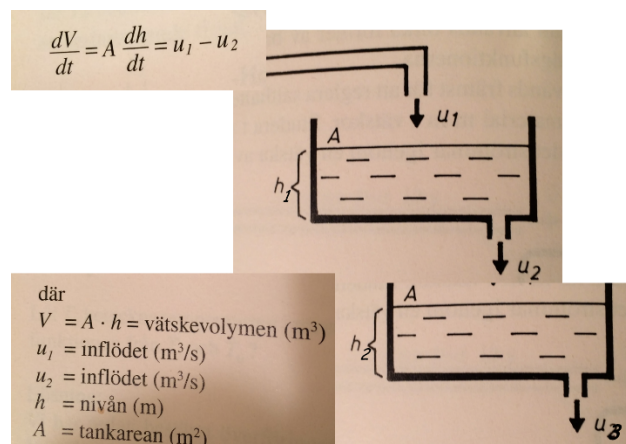
B.1.2 Test av utgångsspänning mellan 0V-ca.2,5V

Utgången ska producera 0-ca 2.2-2.7V för pumpens styrspänning. Välj ingångsspänningar med en extern powerbox mellan 0V-3,3V och mät spänningen vid utgången (pin 6).

Ingångsspänning TS921-krets	TS921-pin 6
0V	1mV
0,5V	54mV
1V	0.45V
1,5V	0.93V
2V	1.4V
3V	2.38V

A.2 Stegsvvar, processtyp och typsiffr

Olika system beter sig annorlunda. Beroende på deras statiska och dynamiska egenskaper kan man dela in dem i olika processtyper. Ett sätt att få systemet att avslöja sina egenskaper är att ta upp stegsvaret. Stegsvaret är förloppet av utgångsvärdet när ingångssignalen är ett steg.



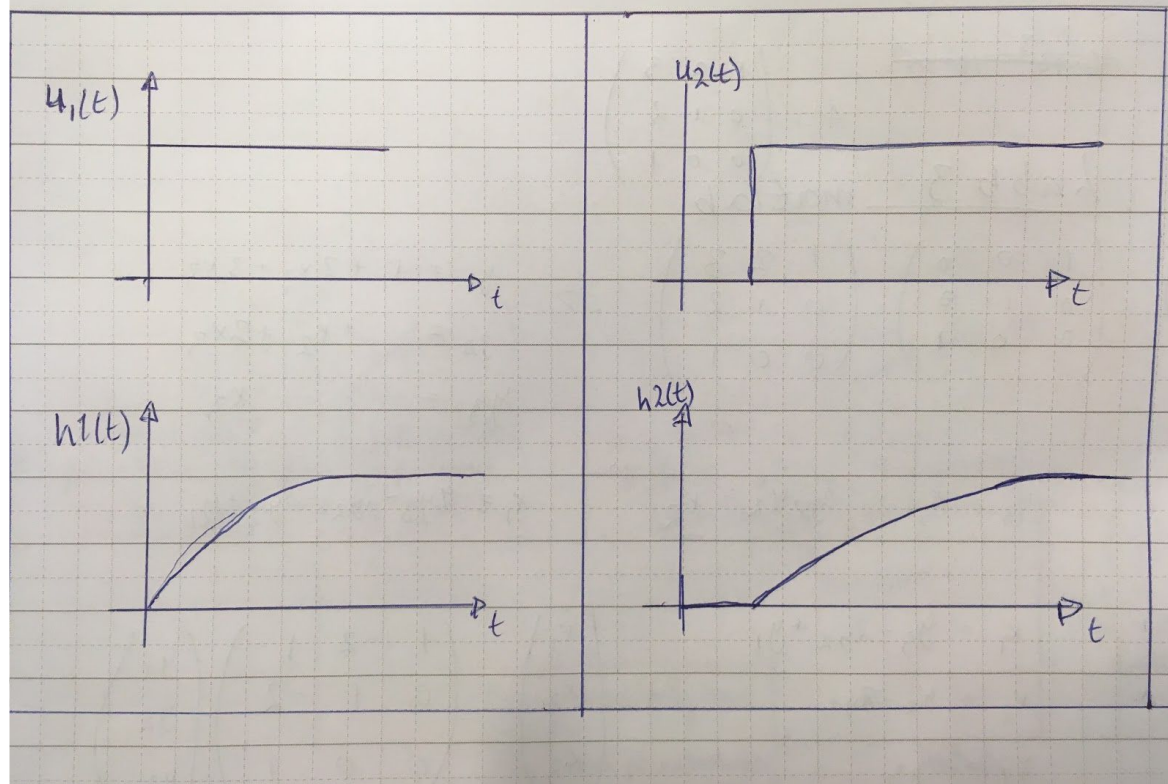
Figur A.2.1 kopplade vattentankar

I följande uppgift fokuserar vi oss på vattenmodellen enligt figur A.2.1 ovan.

A.2.1 Stegsvvar i öppna regelkretsen

Rita upp hur vattennivåerna h_1 och h_2 förändrar sig som följd av ett stegsvvar, dvs om man pumpar in ett konstant flöde u_1 i den första vattentanken och vattnet därifrån flyter ut in i den andra tanken. Vi antar att tankarna är helt tomma i början och att tillflödet u_1 är lagom så att vattnet inte rinner över, dvs det finns ett nivå h_1 där tillflödet u_1 är lika stor som utflödet u_2 .

(Rita på tavlan eller på pappret och tar kort och klistra in här:)



A.2.2 Processtypen och typsiffror för nivån i första eller andra behållaren

a) Utifrån dina funderingar om stegsvaret av $h_1(t)$ och $h_2(t)$ i förgående uppgiften: hur skiljer sig dynamiken åt av de två nivåförloppen?

Övre tanken stiger direkt vilket inte den undre tanken gör eftersom den tar lite tid(dötid) tills den börjar höjas i nivå. Samt att "signalen/inflödet" tar tid för den undre medan den övre sker direkt och inte behöver vänta tills det finns tillräckligt med tryck från övre behållaren.

b) Vilken processtyp och typsiffror har första behållaren? Förklara varför.

typsiffran "0", eftersom vårt stegsvar går mot ett fixt börvärde dvs inte går mot oändligheten. Processortypen är av tidskonstant 1, då det sticker upp direkt och inte har någon dötid.

c) Vilken processtyp och typsiffror har andra behållaren? Förklara varför.

typsiffran "0", eftersom vårt stegsvar går mot ett fixt börvärde dvs inte går mot oändligheten. Processortypen är av tidskonstant 2, då det sticker upp direkt och har en liten dötid.

A.2.3 Programmeringen av stegsvarsexperiment

I uppgiften A.2.1 ritade ni upp förloppet av de förmodade stegsvaren för h1 och h2 av den öppna reglerkretsen. Nu blir det dags att köra stegsvarexperimentet på vattenmodellen och att jämföra resultatet från experimentet med era ritningar.

För detta behöver ni först programmera en Matlabfunktion. Ni kan utgå från en gemensam stomme som ni sedan också kan använda för att programmera de olika regulatorerna.

Stommen av ett matlabprogram visas i figuren nedan och finns förklarad i

Kompendium om reglerteknikutrustningen:

```
function [e, u, y, t]=vm_stomme(a, N, Ts, v)
% Stomme för regulator-block. Kan användas för att lägga till och anpassa till olika
% klassiska, tidsdiskreta regulatorer

% Argument (anpassas efter ändamål)
% a: arduino-objektet som Matlab använder för att kommunicera med Arduino
% N: antal samplingar
% Ts: samplingstiden mellan samplingar
% v: börvärde i digitala enheter (0..1023)

% Resultat (anpassas efter ändamål)
% e: vektor med N mätningar av felsignalen
% u: vektor med N mätningar av styrsignalen
% y: vektor med N mätningar av process-svaren == ärvärden
% t: tidsdiskret tidsvektor 1:N

% Initialisering av variablerna -----
e=zeros(1, N);
u=zeros(1, N);
y=zeros(1, N);
t=zeros(1,N);

start=0; elapsed=0; ok=0; % används för att upptäcka för korta samplingstider
k=0; % samplingsindex

% Konfigurering av in- och utgångar -----
% Arduino ingångar
% analoga ingångar 'A0' och 'A1' behöver inte konfigureras. 0..1023
% 'A0': y
% analoga utgångar behöver inte heller konfigureras. DAC1-> PWM, DAC0 -> DAC,
% 0..255
% 'DAC0': u

% cyklisk exekvering av samplingar
for k=1:N % slinga kommer att köras N-gångar, varje gång tar exakt Ts-sekunder

    start = cputime; %startar en timer för att kunna mäta tiden för en loop
    if ok <0 % testar om samplingen är för kort
        k
        disp('samplingstiden är för lite! Ök värdet för Ts');
        return
    end

    % uppdatera tidsvektorn
```

```

t(k)=k;

% läs ingångsvärde sensorvärden
y(k)= analogRead(a, 'A0'); % mät ärvärdet

% beräkna felvärdet som skillnad mellan ärvärdet och börvärdet
e(k)=v-y(k);

% Regulatorblock
% beräkna styrvärdet, t.ex p-regulator med förstärkning Kp=1
u(k)=e(k); % p-regulator, Kp=1

% begränsa styrvärdet till lämpliga värden, vattenmodellen t.ex. u >=0 och u <255, samt
%   heltal
u(k)=min(max(0, round(u(k)), 255);

% skriva ut styrvärdet
analogWrite(a,u(k), 'DAC0'); %DAC-utgång

%online-plot
plot(t,y,'k-',t,u,'m:',t,e,'b');

elapsed=cputime-start; % räknar åtgången tid i sekunder
ok=(Ts-elapsed); % sparar tidsmarginalen i ok

pause(ok); %pausar resterande samplingstid

end % slut av samplingarna -----

% plotta en fin slutbild,
    plot(t,y,'k-',t,u,'m:',t,e,'b');
xlabel('samples k')
ylabel('y, u ,e')
title('xxx-regulator')
legend('y ', 'u ', 'e ')

% -----

```

Fig A.3.1: Stomme av matlabprogrammet som kan användas för programmeringen.

Anpassa stommen med de rader som krävs för att kunna ta upp stegsvaret av övre och undre vattentank. **Sätt pumpen från noll till maxhastighet i tidpunkten "t=0"**

Tänk på att resultatet, dvs vektorerna med mätvärden, ska senare analyseras, dvs de ska kunna sparas som globala matlabvariabler.

A.2.3.1 Klistra in Matlabprogrammen för stegsvaret här:

```

analogRead(a, 'A0');%Tar bort ful spik i b?rjan av plottningen.
% cyklisk exekvering av samplingar

```

```
for k=1:N % slinga kommer att k?ras N-gångar, varje gång tar exakt Ts-sekunder
```

```
start = cputime; %startar en timer för att kunna mäta tiden för en loop
```

```
if ok < 0 % testar om samplingen är för kort
```

```
    k
```

```
    disp('samplingstiden är för lite! öka värdet för Ts');
```

```
    return
```

```
end
```

```
t(k)=k;
```

```
% l?s ing?ngsv?rde sensorv?rden
```

```
y(k)= analogRead(a, 'A0'); % m?t ?rv?rdet
```

```
analogWrite(a,40, 'DAC1'); %DAC-utg?ng
```

```
end % slut av samplingarna -----
```

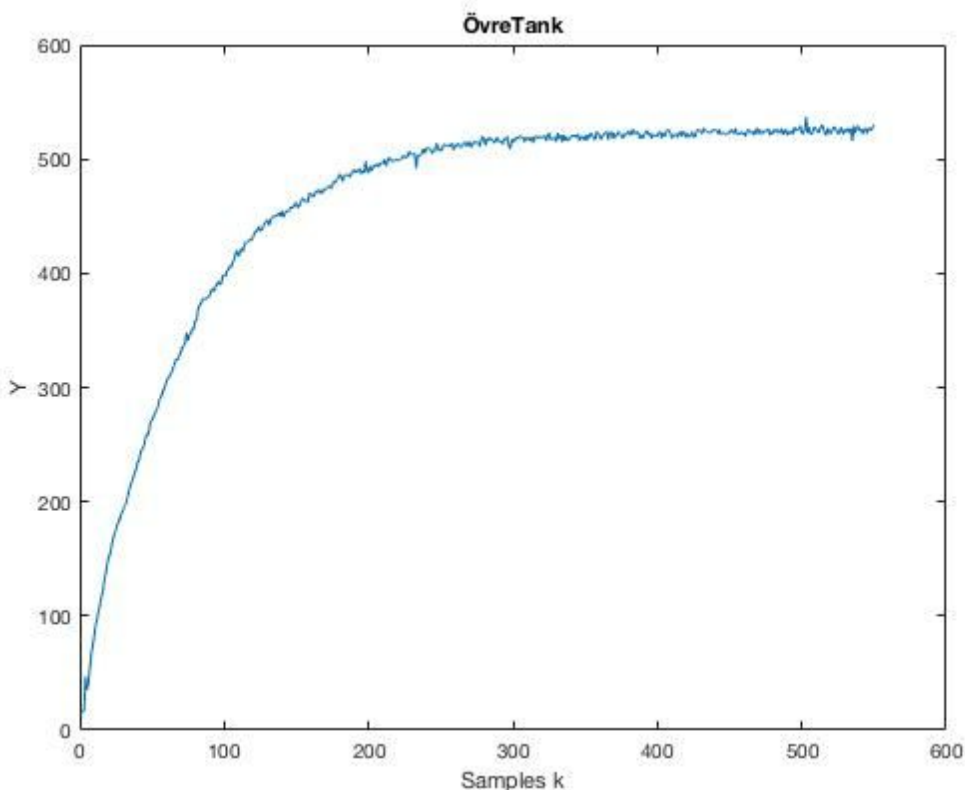
B.2 Stegvarsexperiment

Genomför stegvarsexperiment med den övre och den nedre vattentanken. (För den nedre betyder att den fylls via utloppet av den övre men att man bara mäter nivån i den nedre tanken.)

B.2.1. Stegvarsexperiment

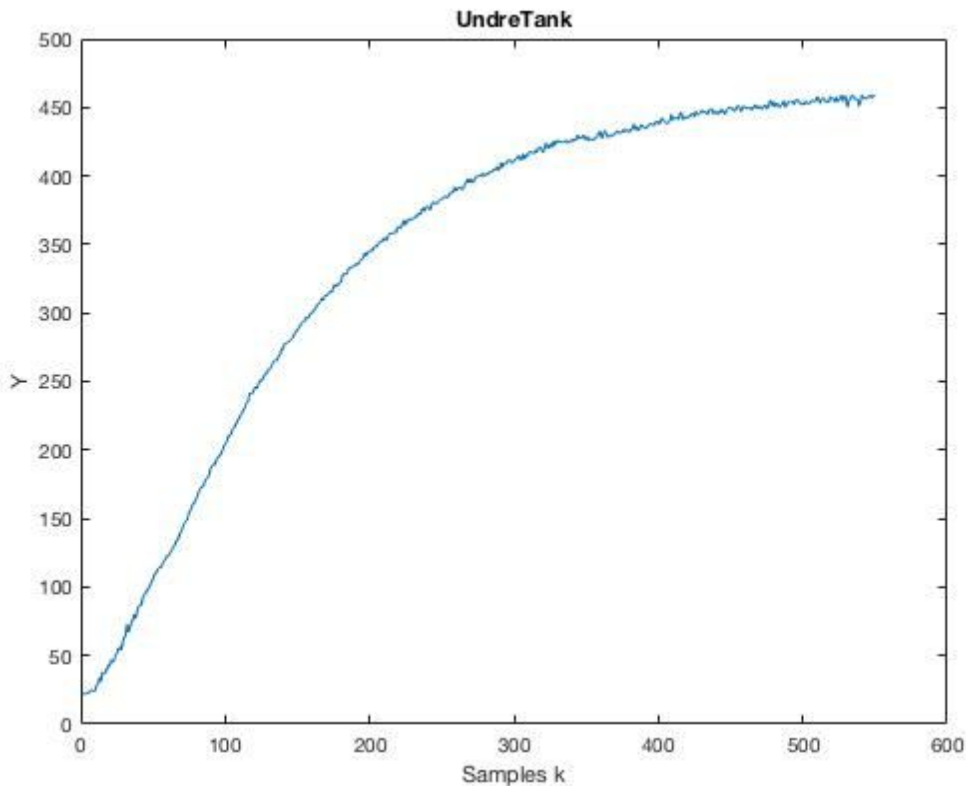
Genomför experimentet och klistra in grafen med stegsvaret av den öppna reglerkretsen här:

a) För övre vattentank



Tankarna får en spänning på $\approx 0,4V$ vilket inte stämmer då den skall börja på 0V detta pga att vattentanken måste kalibreras. Vi har diskuterat detta med dig.

b) För nedre vattentank



Tankarna får en spänning på $\approx 0,2V$ vilket inte stämmer då den skall börja på 0V detta pga att vattentanken måste kalibreras. Vi har diskuterat detta med dig.

- c) Mätningen av stegsvaren har sparats i variablerna `yStegsvar` (övre tank) och `y2Stegsvar` (nedre tank)

C.1 Jämförelse av stegsvaren

Analysera era förväntningar av stegsvaren med era experimentella mätningar.

C.1.2 Jämförelse mellan ritning och resultat

Jämför nu resultatet från experimentet med din ritning i A.2.1:

- a) Vad stämmer bra överens?

Den generella formen på graferna och att den nedre tanken är mer fördröjd än den övre.

- b) Vad är mest annorlunda och varför? Hur förklarar du skillnaden om det finns några?

Den nedre tanken stiger inte lika högt som den övre i det verkliga fallet.

Resultatet från stegsvaret för h_1 och h_2 kommer ni att använda för att identifiera systemet i

form av en differensekvation samt för att få fram parameterinställningar för en rad regulatorer med hjälp av tumreglerna. Men först behöver dem filtreras något.

C.2 Filtrering av mätvärden

Mätvärden för nivån i övre och i nedre vattentanken från stegsvarsexperimentet behöver filtreras något för att dämpa störningarna innan vi kan använda dem vidare. Ni kan använda vilket lågpasfilter som helst som ni kanske redan har använt i Tommys del av kursen. Beskriv i så fall hur filtren ser ut och vad den gör:

Alternativet är att du bygger en enkel "moving average filter", se också utdraget ur Matlabs dokumentation nedan

Examples

Moving-Average Filter of Vector Data

Find the moving-average of a vector without using a `for` loop.

A moving-average filter is represented by the following difference equation,

$$y(n) = \frac{1}{windowSize} (x(n) + x(n-1) + \dots + x(n - (windowSize - 1))).$$

Define the numerator coefficients of the rational transfer function. Use a window size of 5.

```
windowSize = 5;

b = (1/windowSize)*ones(1,windowSize)

b =

    0.2000    0.2000    0.2000    0.2000    0.2000
```

Define the denominator coefficients of the rational transfer function.

```
a = 1;
```

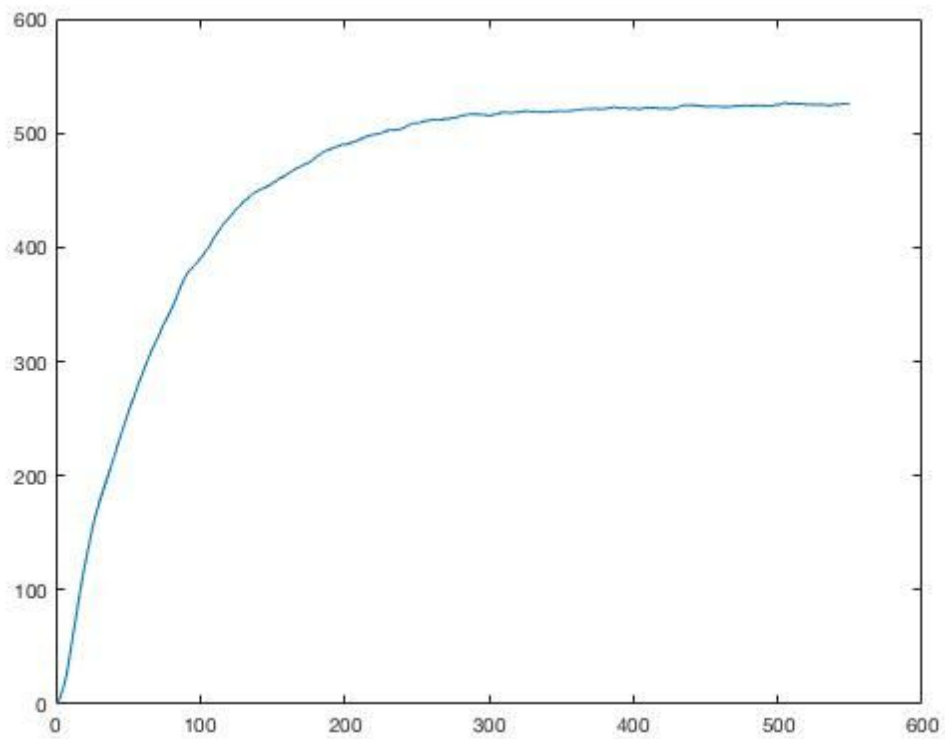
Find the moving-average of the data with a window size of 5.

```
y = filter(b,a,x);
```

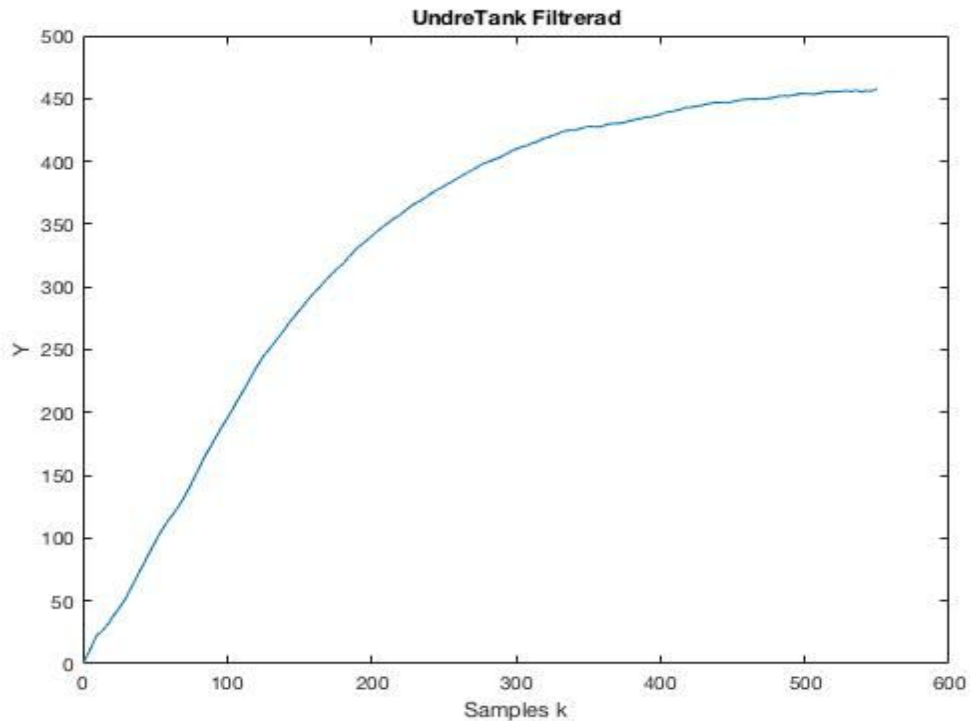
Testa gärna med olika värden för `windowSize`. Istället för "x" ska ni ange er variabel med mätvärden från nivåmätningen. Resultatet y blir då det filtrerade signal.

C.2.1 Filtrera mätningarna från stegsvaren, spara dem i globala variabler och klistra in grafen (plots) här:

a) Filtrerad stegsvarsmätning från övre vattentank



b) Filtrerad stegsvarsmätning från nedre vattentank



c) Val av "windowSize": 10

d) Global variabel för filterad stegsvar, övre vattentank: yfiltstegsvar

e) Global variabel för filterad stegsvar, nedre vattentank: y2filtstegsvar

A.3 Klassiska reglerprinciper

Nu ska ni testa olika reglerprinciper som används i "regulatorblocken" inom den klassiska, enkla reglerkretsen. Regulatorn är den del i blockdiagrammet som har felsignalen $e(t)$ som ingång och styrsignalen $u(t)$ som utgång.

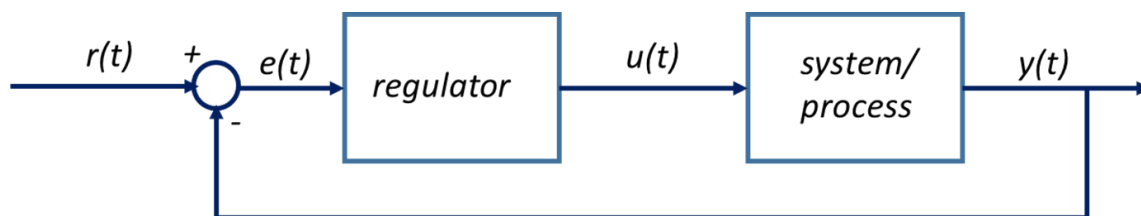


Fig A.3: blockdiagram av en enkel regelkrets

Först måste de olika regulatorerna programmeras. Ni kan utgå från samma stomme

som ni redan har använt för programmeringen av stegsvaren.

A.3.1 Programmering av en tidsdiskret tvålägesreglering

Anpassa programstommen till en tidsdiskret tvålägesreglering och spara programmet som .m-fil.

A.3.1.1 Kopiera in Matlabkoden för regulatorblocken som implementerar en "tvålägesreglering":

```
y(k)= analogRead(a, 'A0'); % m?t ?rv?rdet
% beräkna felvärdet som skillnad mellan ärvärdet och börvärdet
e(k)=v-y(k);
if(e(k)<0)

    u(k)=0;

elseif(e(k)>0)
    u(k)=150;
end
```

A.3.2 Programmering av en tidsdiskret flerstegsreglering

Anpassa programstommen till en tidsdiskret flerstegsreglering och spara programmet som .m-fil.

A.3.2.1 Kopiera in Matlabkoden för regulatorblocken som implementerar en "flerstegsreglering":

```
e(k)=v-y(k);
if(e(k)<=0)
    if(e(k)>=-30)
        u(k)=0;
    else
        u(k)=50;
    end

elseif(e(k)>0)

    if(e(k)>30)
        u(k)=130;
    else
        u(k)=80;
    end
end

% skriva ut styrvärdet
analogWrite(a,u(k), 'DAC1'); %DAC-utgång
```

A.3.3 Programmering av en tidsdiskret P-reglering

Anpassa programstommen till en tidsdiskret P-reglering och spara programmet som .m-fil.

A.3.3.1 Kopiera in Matlabkoden för regulatorblocken som implementerar en "P-reglering":

```

y(k)= analogRead(a, 'A0'); % mät ärvärdet
% beräkna felvärdet som skillnad mellan ärvärdet och börvärdet
e(k)=v-y(k);

% Regulatorblock
% beräkna styrvärdet, t.ex p-regulator med förstärkning Kp=1
u(k)=e(k); % p-regulator, Kp=1

% skriva ut styrvärdet
analogWrite(a,u(k), 'DAC1'); %DAC-utgång

```

A.3.4 Programmering av en tidsdiskret PID-reglering

Anpassa programstommen till en tidsdiskret PID-reglering och spara programmet som .m-fil.

A.3.4.1 Kopiera in Matlabkoden för regulatorblocken som implementerar en "PID-reglering":

```

y2=zeros(1,N);
K=1;
Td=1;
Ti=10;
% I?s ingångsvärde sensorvärden
y(k) = analogRead(a, 'A0'); % mät ärvärdet
y2(k) = analogRead(a, 'A1');
% beräkna felvärdet som skillnad mellan ärvärdet och börvärdet
e(k) = v-y2(k);
w(k) = w(k-1)+e(k);
u(k) = K*(e(k)+Td*((e(k)-e(k-1))/Ts)+(Ts/Ti)*w(k));

u(k)=min(max(0, round(u(k))), 255);

% skriva ut styrvärdet
analogWrite(a,u(k), 'DAC1'); %DAC-utgång

```

A.3.5 Programmering av en tidsdiskret PD-reglering

Anpassa programstommen till en tidsdiskret PD-reglering och spara programmet som .m-fil.

A.3.5.1 Kopiera in Matlabkoden för regulatorblocken som implementerar en "PD-reglering":

```

y2=zeros(1,N);
Kp=1;
Td=1;

y(k)= analogRead(a, 'A0'); % mät ärvärdet tank1
y2(k)=analogRead(a, 'A1'); % mät ärvärdet tank2

e(k)=v-y2(k);
u(k)=Kp.*e(k)+Kp*(Td/Ts).*(e(k)-e(k-1));

```

B.3 Reglering med tidsdiskreta klassiska reglerprinciper

Dags att testa era olika regulatorer på riktigt vid vattentankmodellen. Följ informationen från tabellen i dokumentet "Överblick över reglerteknikdelen" angående vilka regulatorer som ska testas med övre och vilka med undre vattentanken.

Vänligen läs noggrant igenom följande anvisningar innan ni fortsätter med experimenten:

1. Resultaten (dvs mätvärden av vattennivåerna) av alla genomförda experiment ska sparas för att kunna analyseras och jämföras senare. Det enklaste sättet att göra det i Matlab är att ange specifika variabelnamn i samband med funktionens upprop i kommandofönster, exempelvis:

```
>> [h1o,h2o,to,uo] = vm_openstep(a,N,dT,v)
>> [h1P,h2P,tP,uP,eP] = vm_P(a,N,dT,bv, Kp)
osv
```

Det behövs inga ändringar i själva programmet, Matlab överger variablerna i samma ordning som står i funktionsdeklarationen efter nyckelordet "function".
2. Alla variabler som finns i arbetsminne kan sedan sparas tillsammans i en binär fil (extension ".mat"). Föreslaget är att ni använder filnamn "labb1504e". Kommandon i kommandofönster för att spara är:

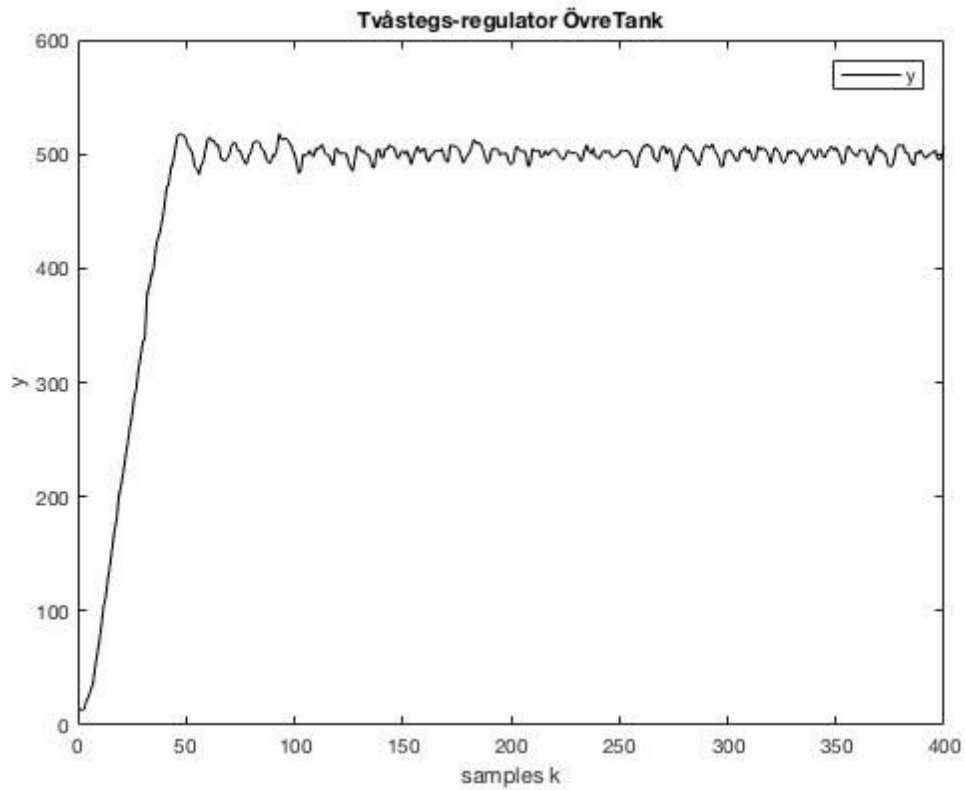
```
>> save labb1504e
```
3. Spara era resultat på detta viset i samma fil efter varje experiment, så att ingenting tappas bort.
4. Om ni genomför den praktiska delen i olika omgångar så kan ni börja om där ni slutade genom att ladda in alla variabler från filen genom kommandon:

```
>> load labb1504e
```

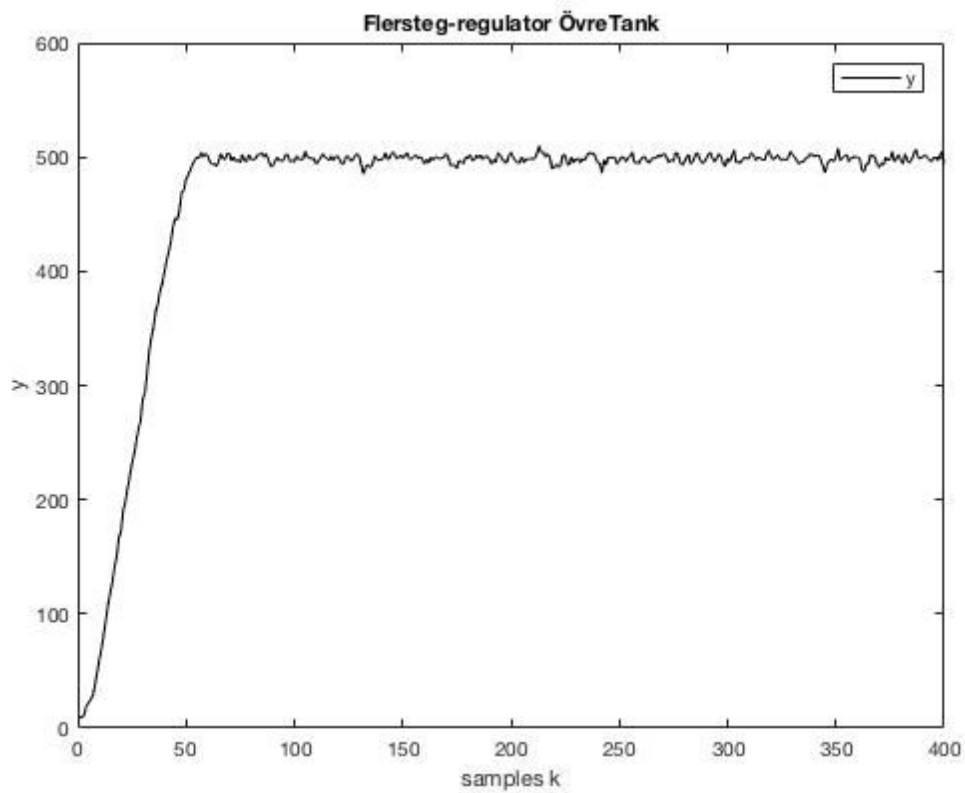
(Och sedan sparar ni enligt pkt 2 och 3)
5. Angående de olika experimenten med olika regulatorer:
 - Testa er fram angående olika börvärden. Om börvärdet är för högt kan det ta längre tid för att uppnå värdet. Om det är för lågt kan fallhöjden av vattnet leda till för mycket turbulens i behållaren som gör det svårt att tolka resultatet.
 - När ni har hittat ett bra värde för börvärdet ska ni helst använda detsamma i de olika experimenten för att få en bättre jämförbarhet.
 - Minimalkrav för alla experiment är att de genomförs så att man ser hur systemet svänger in sig eller försöker svänga in sig mot börvärdet eller ändringen i börvärdet. Välj därför inte en för kort tid för era experiment!! (Minst 3 minuter kan vara lagom).
6. För ambitiösa studenter: Syftet med regleringen är att kunna följa börvärdesändringar och att kompensera för störningar. I minimalkraven enligt pkt5 tester vi reglersystemets stegsvar från noll till börvärdet. Vad man därför dessutom skulle vilja testa är ändringar i stegsvaret (till olika nivåer eller att följa en ramp) och compensation av störningar. Genom att skriva ett litet skript (te.ex. som egen funktion) skulle man för varje regulator kunna köra samma sekvens, exempelvis.
 - Vanlig stegsvar till börvärdet första n-samplingar, sedan
 - ändring i börvärdet (som steg eller ramp) för m-samplingar, sedan

- uppmana användaren att initiera en störning (genom att öppna extra utlopp) och köra för m-samplingar
- det vore också bra att i slutplotten ange var del 2 och 3 började

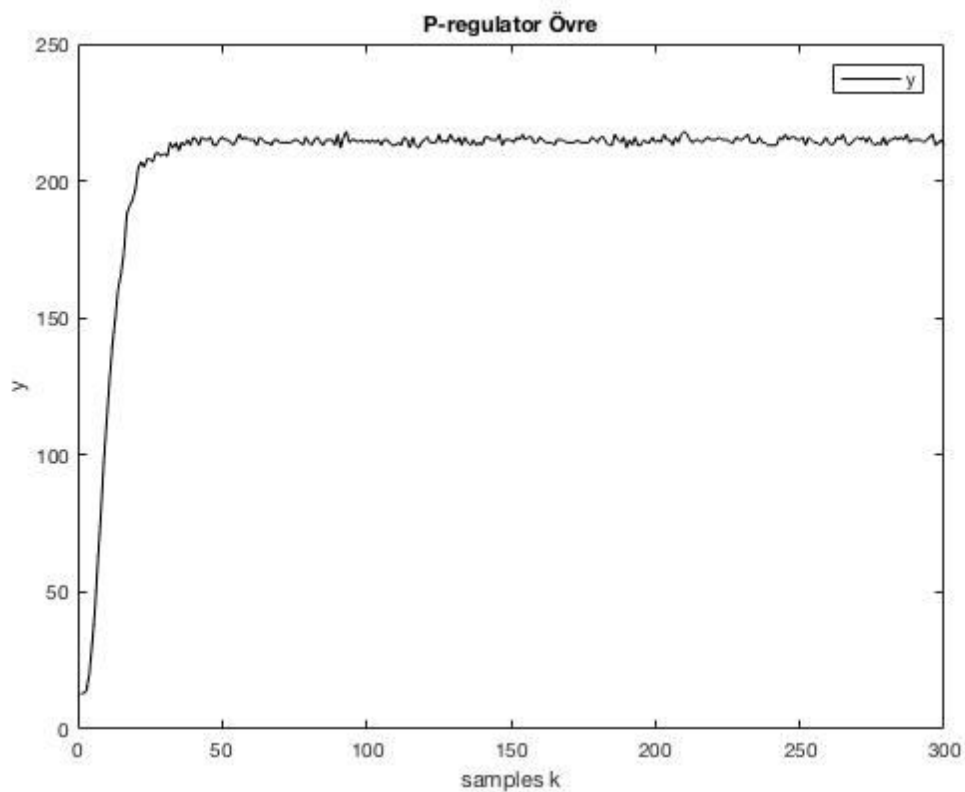
B.3.1. Kopiera in grafen från "tvålägesregleringen" här:



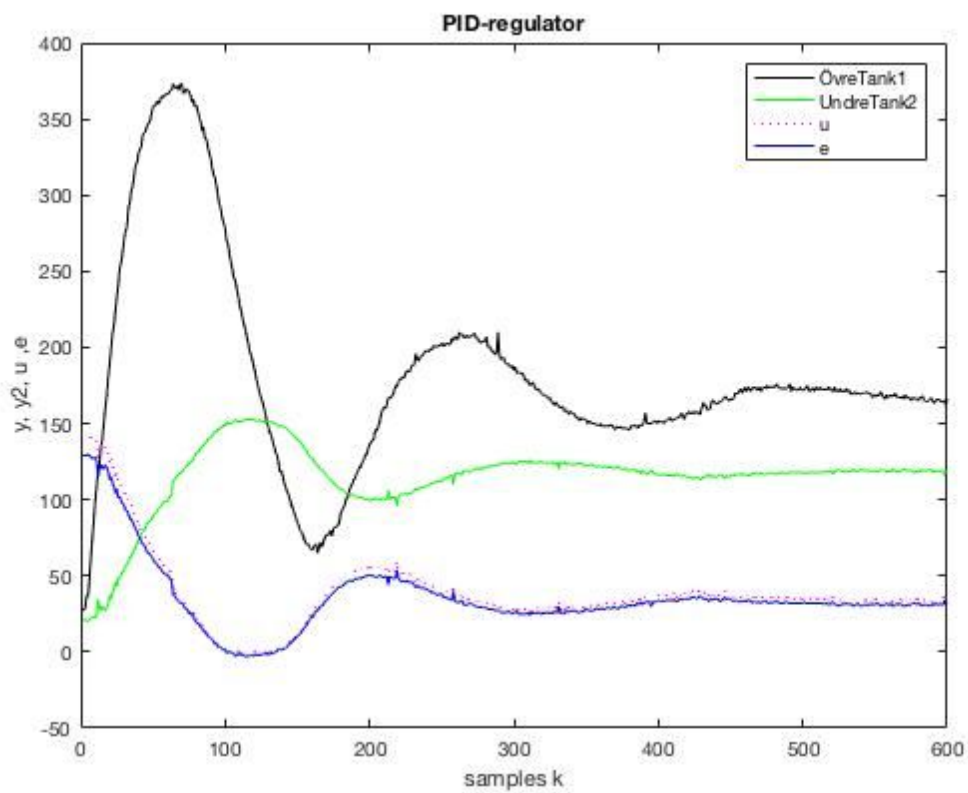
B.3.2 Kopiera in grafen från "flerstegsregleringen" här:



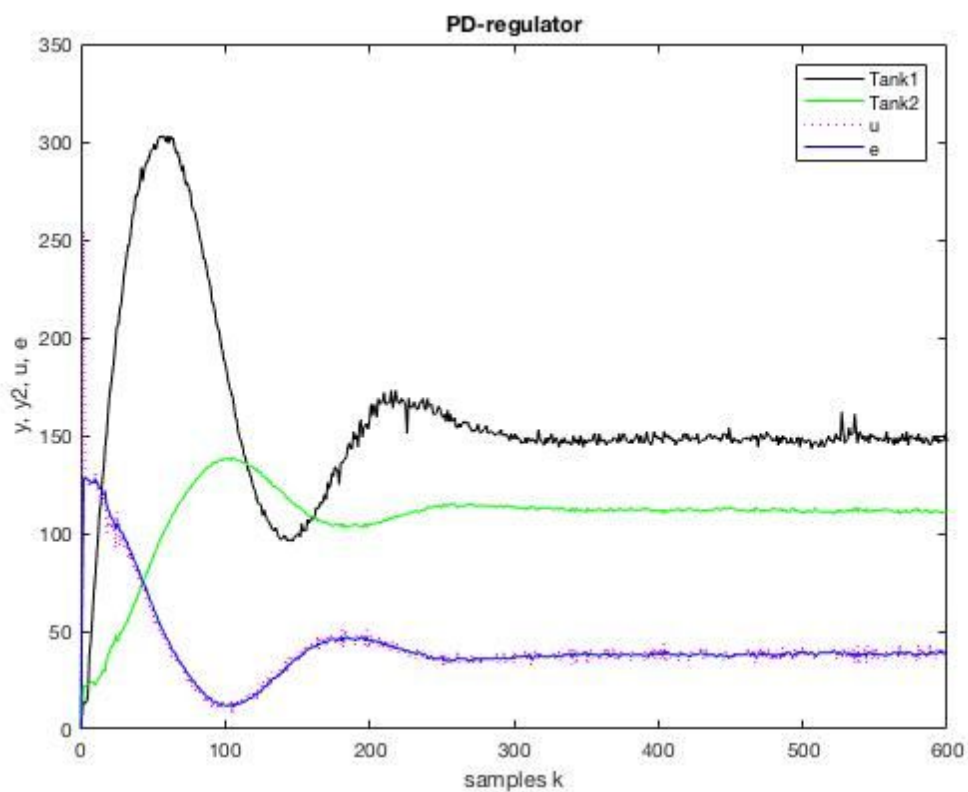
B.3.3 Kopiera in grafen från "P-regleringen" här:



B.3.4 Kopiera in grafen från "PID-regleringen" här:



B.3.5. Kopiera in grafen från "PD-regleringen" här:



C.3 Dimensionering av reglersystem med tumregelmetoder

Att välja reglerparametrarna, t.ex. för en PID-regulator kan vara tidskrävande och besvärlig. Istället för att använda sig av "trial and error" finns det tumregelmetoder. Ni ska få testa Ziegler-Nichols svängningsmetoden samt Amigo-metoden (som i sin tur baserar på Lambdametoden).

C.3.1 Lambda- och Amigo-metoden

Utgå från stegsvaret för undre vattentanken, dvs övre- och undre vattentank tillsammans. Analysera stegsvaret enligt beskrivningen till Lambda-metoden. Bestäm K_s , L och T . Använd sedan Amigo-metoden för att räkna ut K , TI och TD för PID-regulatorn.

C.3.1.1 Vilka värden får ni från Lambda-metoden för:

a) $K_s = 612/70 \approx 8,742857143$

b) $L = 2$

c) T ger $0,63 \cdot 612 = \text{Sampling } 386,56$ dvs $T = 386,56 - 3 = 383,56$

C.3.1.2 Vilka värden får ni från Amigo-metoden för:

a) $K = \frac{1}{K_s} \left(0,2 + 0,45 \cdot \frac{T}{L} \right) = \frac{1}{8,742857143} \left(0,2 + 0,45 \cdot \frac{612}{2} \right) = 9,8939052286$

b) $TI = \frac{0,4 \cdot L + 0,8 \cdot T}{L + 0,1 \cdot T} \cdot L = \frac{0,4 \cdot 2 + 0,8 \cdot 383,56}{2 + 0,1 \cdot 383,56} \cdot 2 = 15,24670433$

c) $TD = \frac{0,5 \cdot L \cdot T}{0,3 \cdot L + T} = \frac{0,5 \cdot 2 \cdot 383,56}{0,3 \cdot 2 + 383,56} = 0,998438151$

B.4. Ziegler-Nichols svängningsmetod

Tillämpa nu också svängningsmetoden för att få fram alternativa PID-regulatorparametrar.

B.4.1 Klistra in en graf som visar hur ni fick nivån i undre vattentanken att självsvänga enligt svängningsmetoden:

B.4.2 Vilka kritiska förstärkning K_0 och periodtid T_0 får ni?

$K_0 = 9,97209$

$T_0 = 48$

B.4.3 Vilka blir PID-regulatorparametrarna enligt Ziegler-Nichols tabell?

$K = 0,6 \cdot K_0 = 0,6 \cdot 9,97209 = 5,983254$

$$T_I = 0,5 \cdot T_0 = 0,5 \cdot 48 = 24$$

$$T_D = 0,125 \cdot T_0 = 0,125 \cdot 48 = 6$$

B.4.4 Vilka blir PID-regulatorparametrarna enligt Åström & Hägglunds föreslag?

$$K = 0,35 \cdot K_0 = 0,35 \cdot 9.97209 = 3.4902315$$

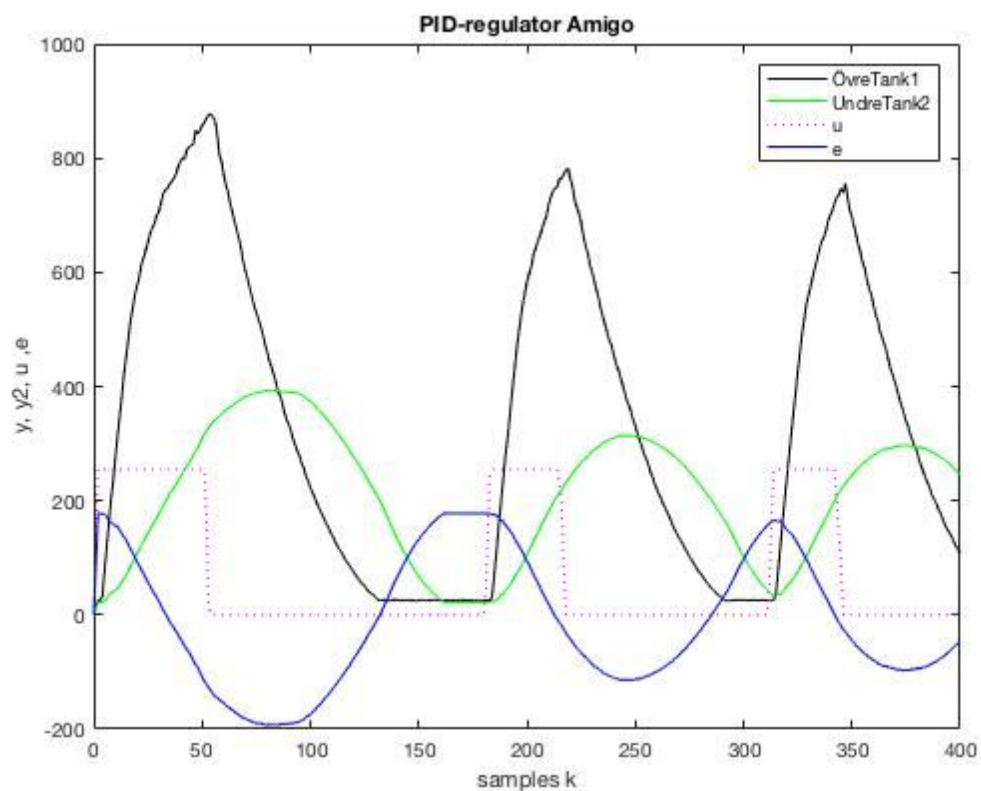
$$T_I = 0,77 \cdot T_0 = 0,77 \cdot 48 = 36.96$$

$$T_D = 0,19 \cdot T_0 = 0,19 \cdot 48 = 9.12$$

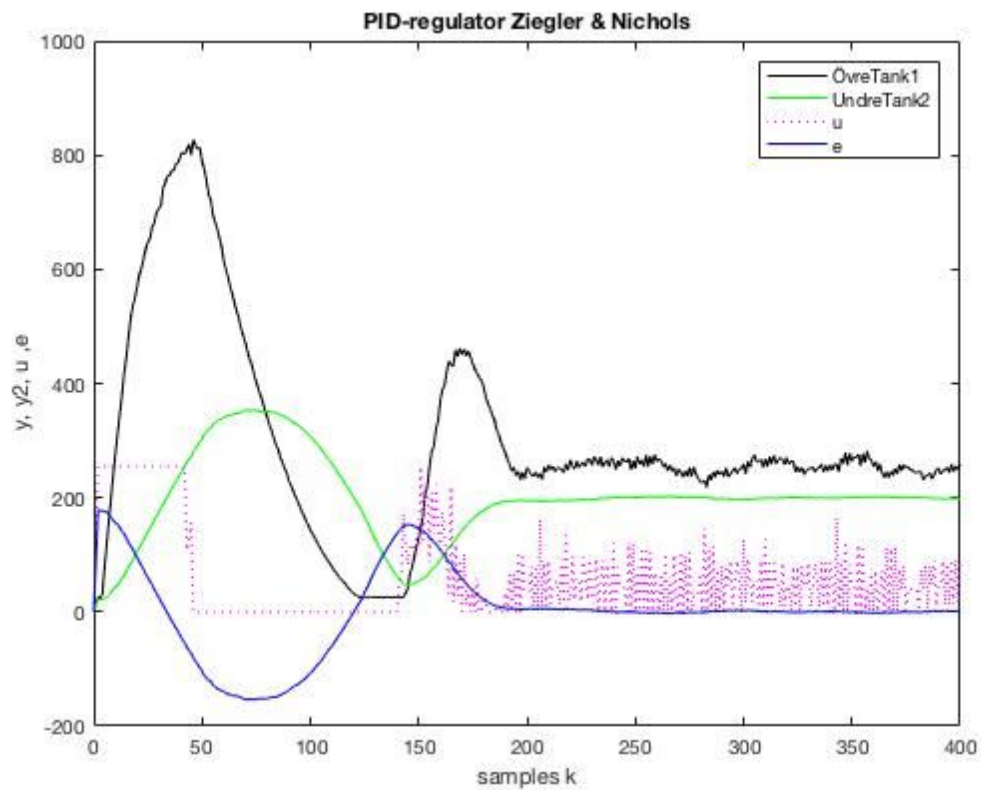
B.5 PID-regulator med olika parametrar enligt tumregler

Testa nu de olika regulatorparametrarna för en PID-regulator av den nedre vattentanken.

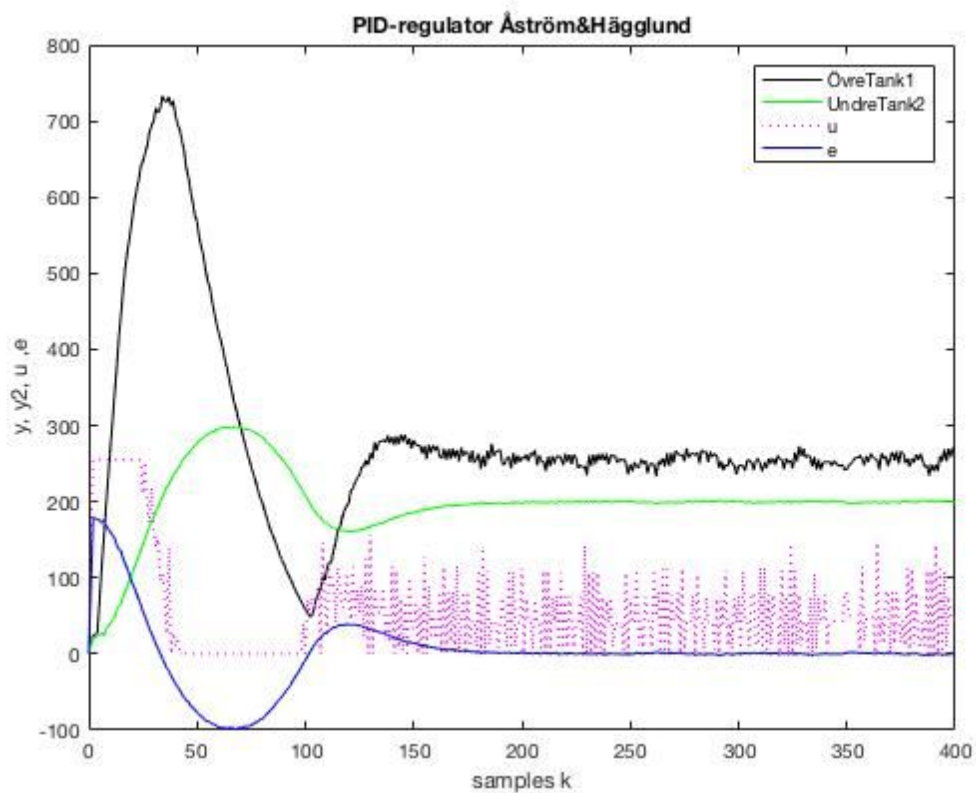
B.5.1 Kopiera in grafen från PID-regulatorn med Amigo-metoden-parametrarna



B.5.2 Kopiera in grafen från PID-regulatorn med Ziegler-Nichols-metoden-parametrarna



B.5.3 Kopiera in grafen från PID-regulatorn med Åström & Hägglunds parametrar



C.4 Jämförelse mellan olika reglersystem

Ni har reglerat samma process (=vattentankmodell) med tre olika inställningar av en PID-regulator. Jämför de tre resulterande systembeteende med varandra genom att fylla i tabellen nedan.

C.4.1 Jämförelse av systemens egenskaper. Fyll i tabellen:

	PID-Amigo	PID-ZN	PID-ÅH
Stigtid	337	179	140
Max. översvängning	193	154	99
Insvängningstid	32	33	32
Kvarstående fel	-70%	1.3125%	-0.5625%

A.4 Bestämning av tidsdiskreta överföringsfunktionen

Utifrån stegsvaren av de övre och undre vattentankarna kan man med hjälp av minsta kvadratmetoden räkna ut överföringsfunktionerna. Det kallas också för en "Black-box"-systemidentifikation.

Använd de filtrerade stegsvaren för bästa resultat.

A.4.1 Beskriv och visa steg för steg hur ni bestämmer överföringsfunktionen för den övre vattentanken utifrån det filtrerade stegsvaret:

$$\begin{aligned}y(k) &= a_1 y(k-1) + b_1 u(k-1) \Rightarrow y(k) - a_1 y(k-1) = b_1 u(k-1) \Rightarrow \\Y - a_1 Y z^{-1} &= b_1 U z^{-1} \Rightarrow Y[1 - a_1 z^{-1}] = U[b_1 z^{-1}] \\H(z) = \frac{Y}{U} &= \frac{b_1 z^{-1}}{1 - a_1 z^{-1}}\end{aligned}$$

matris med värden

$$\begin{bmatrix} y(550) \\ \vdots \\ y(2) \\ y(1) \end{bmatrix} = \begin{bmatrix} y(550-1) & u(550-1) \\ \vdots & \vdots \\ y(1) & u(1) \end{bmatrix} \cdot \begin{bmatrix} a \\ b \end{bmatrix}$$

$y1filt$ A u

$$u = \begin{bmatrix} 60 \\ \vdots \\ 60 \end{bmatrix}$$

matlab definierar matris $A = y1filt(1:549)$
 $A(:,2) = u(1:549)$

Svar = $A \backslash y1filt$

$H2 = tf([0 \ b_1], [1 \ a_1], 1, 'variable', 'z^{-1}')$

A.4.2 Beskriv och visa steg för steg hur ni bestämmer överföringsfunktionen för den övre och undre vattentanken utifrån det filtrerade stegsvaret:

$$y(k) = a_1 y(k-1) + a_2 y(k-2) + b_1 u(k-1) + b_2 u(k-2) \Rightarrow$$

$$y(k) - a_1 y(k-1) - a_2 y(k-2) = b_1 u(k-1) + b_2 u(k-2) \Rightarrow$$

$$Y - a_1 Y z^{-1} - a_2 Y z^{-2} = b_1 U z^{-1} + b_2 U z^{-2} \Rightarrow$$

$$Y [1 - a_1 z^{-1} - a_2 z^{-2}] = U [b_1 z^{-1} + b_2 z^{-2}] \Rightarrow$$

$$H(z) = \frac{Y}{U} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 - a_1 z^{-1} - a_2 z^{-2}}$$

pwmm-signal array 550 värden
 $u = [60 \dots 60 \dots 60]$

$AB = y2fit(2:550-2)$
 $A3(2) = y2fit(1:550-3)$
 $A3(:,3) = y(2:550-2)$
 $A3(:,4) = u(1:550-3)$

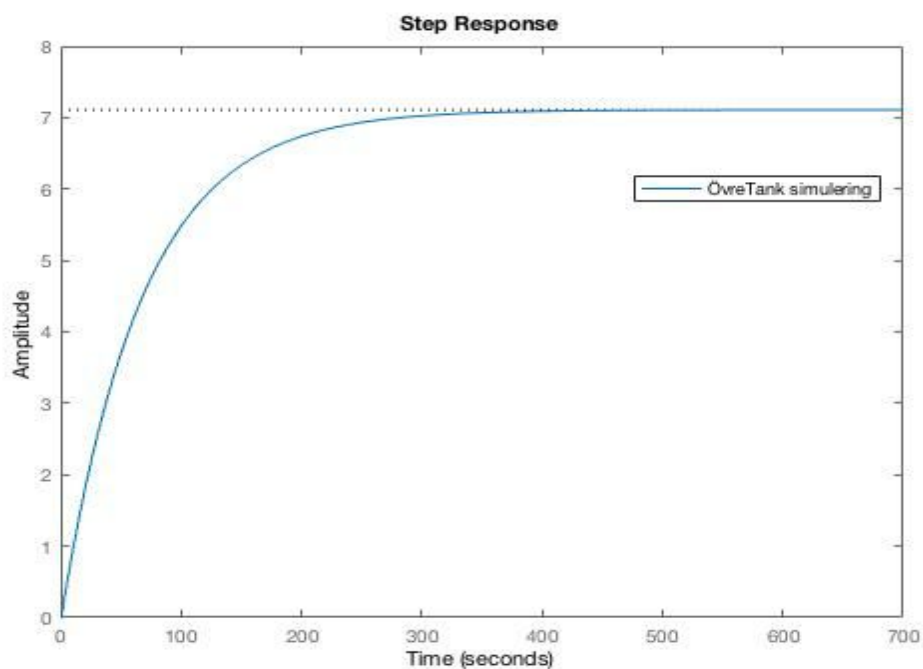
$$\begin{bmatrix} y(3) \\ \vdots \\ y(550) \end{bmatrix} = \begin{bmatrix} y(2) & y(1) & u(2) & u(1) \\ \vdots & \vdots & \vdots & \vdots \\ y(10) & y(9) & u(10) & u(9) \\ \vdots & \vdots & \vdots & \vdots \\ y(N-2) & y(N-3) & u(N-2) & u(N-3) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ b_1 \\ b_2 \end{bmatrix}$$

$R = A \setminus y2fit$
 $H3 = tf([b_1, b_2], [a_1, a_2], 1, 'variable', 'z^{-1}')$

A.5 Simulation av stegsvaren med Matlab Control-Toolbox

Med hjälp av överföringsfunktionerna från Black-box systemidentifikationen kan man i Matlab simulera stegsvaren.

A.5.1 Simulera stegsvaren av övre vattentank med Matlab och spara resultatet i en variabel. Klistra in grafen här:

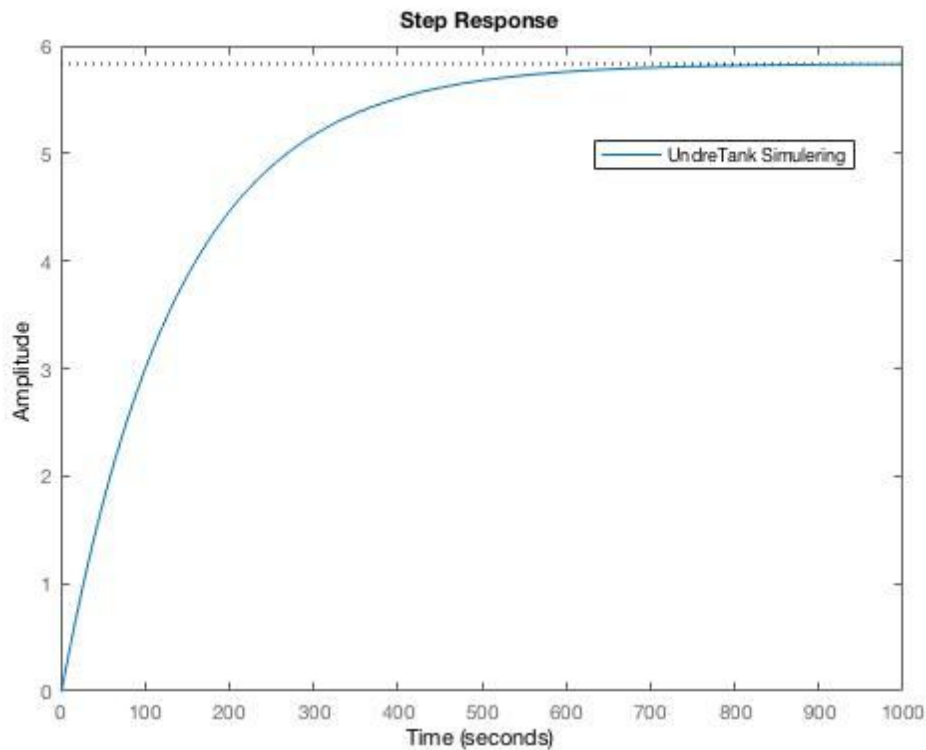


Variabelnamn med sparade stegsvar:

H för överföringsfunktionen, sammansatt stegsvaren A2

A.5.2 Simulera stegsvaren av övre och nedre vattentank tillsammans med Matlab och spara resultatet i en variabel.

Klistra in grafen här:



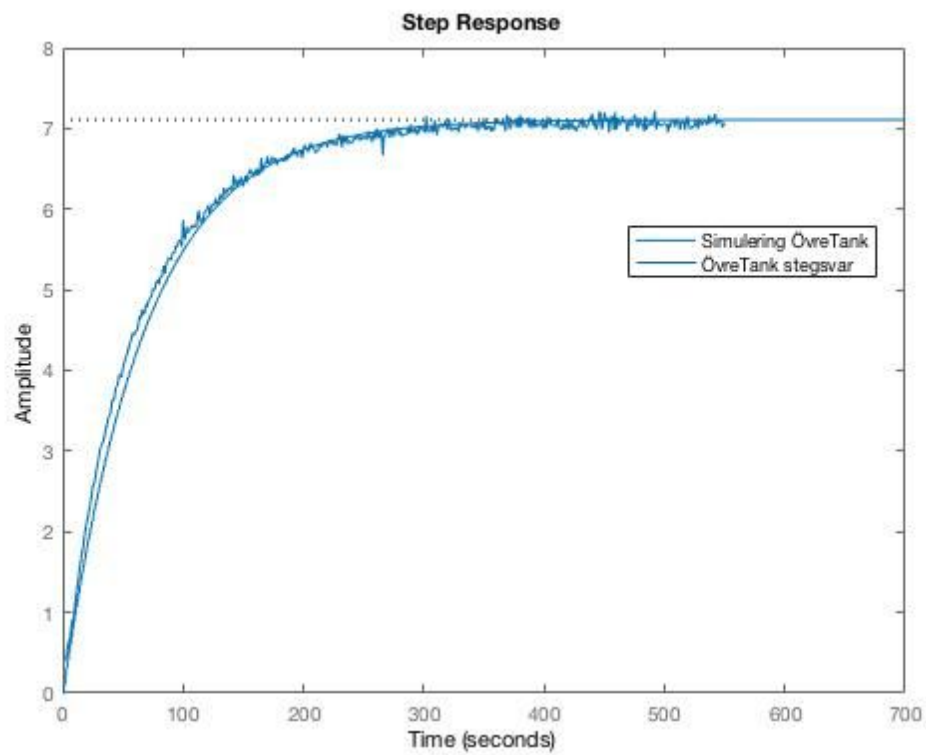
Variabelnamn med sparade stegsvar:

H2 för överföringsfunktionen, sammansatt stegsvaren A2

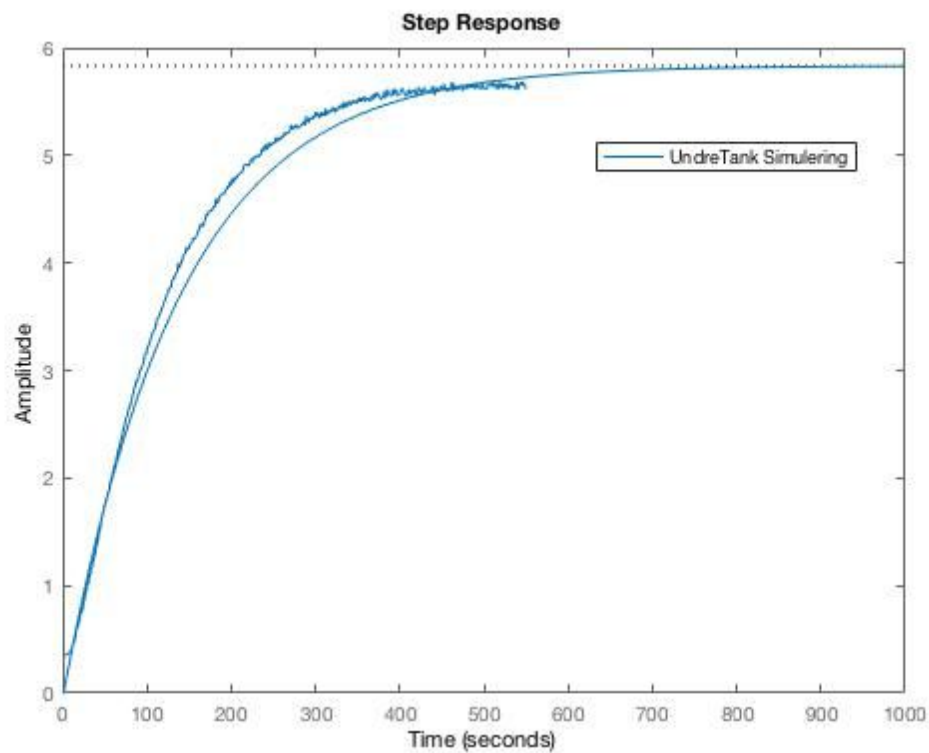
C.5 Jämförelse mellan verkliga och simulerade stegsvaren

Till slut ska ni jämföra de verkliga stegsvaren (innan filtrering) från själva stegsvarexperimenten med de simulerade. Genom att använda sig av Matlabkommandon "hold" kan man plotta flera kurvor över varandra i samma plot, se också Matlabguiden. Ni ska ha sparat de olika stegsvaren i variabler. Dessa kan ni nu plotta i samma graf om ni ser till att de är lika långa.

C.5.1 Jämförelse mellan verkliga och simulerade stegsvaren för övre vattentank. Klistra in sammansatta grafen här:



C.5.2 Jämförelse mellan verkliga och simulerade stegsvaren för övre och nedre vattentank tillsammans, dvs för nivån i nedre vattentanken. Klistra in sammansatta grafen här:



D. Reflektion och utvärdering över det egna lärandet

Tidigare studenter har framfört olika åsikter om denna tredje del. För att förklara vad den går ut på hänvisar jag till kapitel om "Reflektionsdokumentet" i Andersen och Schwenckes bok "Projektarbete – en vägledning för studenter", Studentlitteratur. Författarna beskriver syftet med reflektion över sitt lärande sammanfattningsvis ungefär på följande sätt:

Det handlar om att öka medvetenheten om den egna inlärningsprocessen. Genom denna reflektion lär du känna dig själv, och hur du förhåller dig till dina medstudenter. Du blir också tryggare i dig själv och det du har gjort och står för. Du blir också mer medveten om hur du lär, så att du kan styra inläringen och studierna i din egen riktning och ta ansvar för din egen inläring. Du utvecklar medvetenhet om inläring och stärker din egen förmåga att arbeta effektivt på längre sikt.

Vidare är det ett viktigt syfte att ge feedback, både till läraren, kursansvarig och till skolan. Därmed blir det möjligt att göra förbättringar för senare studentkullar.

I fall att ni känner att ni inte kan komma på något vettigt att skriva om, så finns en lista med nyckelfrågor i bilagan som ni kan titta på och få inspiration ifrån.

D.1 Vad tycker du/ni var lärorik med uppgiften? (Minst 5 meningar och minst 75 ord!)

Det här var vår första laboration i reglerteknik. Det var ett helt nytt område för oss, vilket var spännande, utmanande och väldigt roligt! Kul att testa på olika styrningsmetoder praktiskt med vattentankssystemen utöver det teoretiska som vi har gått igenom på föreläsningarna. Det blir mycket lättare att förstå allting när man kan se vad som verkligen händer i verkligheten. Kul att testa på blackbox också! Det var också något som var helt nytt för oss!

D.2 På vilket sätt har ni fördjupat er i något nytt? Vad kände ni från tidigare och på vilket sätt har ni lärt er något nytt utifrån det ni redan kunde? (Minst 5 meningar och minst 75 ord!)

Reglerteknik är ett helt nytt område för oss. Vi har använt MATLAB mycket tidigare i andra labbar, men inte till just reglerteknik. Det kändes bra att lära sig något helt nytt genom en utvecklingsmiljö vi har använt tidigare. Då kan man fokusera mer på det nya, och inte så mycket på att lära sig utvecklingsmiljön också. Denna labb var ett väldigt bra sätt att praktiskt kunna se och förstå vad reglerteknik egentligen innebär och hur vi praktiskt kan använda saker vi gått igenom på föreläsningar.

D.3 Vad var det svåraste med uppgiften? (Minst 5 meningar och minst 75 ord!)

Det svåraste var att veta när man skulle nöja sig med sina mätresultat och gå vidare. Det var alltså svårt att veta om man gjort rätt eller inte. Annars var A.4 och A.5 också ganska jobbiga innan vi hittade rätt sidor i boken.

Hur mycket tid totalt har ni lagt ner på att lösa uppgiften och hur mycket av denna tid har ni lagt på det som ni anser var det svåraste?

Vi har lagt ner ungefär 25 timmar på labben, och det som tagit längst tid har varit att köra och mäta flertal gånger för de olika styrsystemen, eftersom vi aldrig riktigt visste när vi skulle nöja oss med våra mätvärden.

D.4 Synpunkter, förslag, kommentarer? (Minst 5 meningar och minst 75 ord!)

Fler vattentanksystem, eller fler ordinarie labbtillfällen önskas. Vattentanksystemen verkar inte vara kalibrerade riktigt lika. Går man från ett system till ett annat kan mätvärden skilja rätt mycket trots att kod och koppling är samma. Detta kan vara problematiskt om man bokat en annan tank vid ena tillfället och en annan tank vid andra tillfället. Detta kan vara bra att informera studenterna om i framtiden, så de ser till att boka samma tank vid sina labbtillfällen.

Bilaga

Översikt över Matlab instruktioner

Matlab-instruktion	argument	Beskrivning
pinMode(a,pin,state)	a: arduino-objektet pin: pin-nummer (2-13) state:'OUTPUT', 'INPUT'	Initialiserar port med pin-nummer till in- eller utgång.
digitalWrite(a,pin,level)	a: arduino-objektet pin: pin-nummer (2-13) level: 0, 1	Sätter en pinne till 0 eller 1. Måste först deklarerats som utgång
digitalRead(a,pin)	a: arduino-objektet pin: pin-nummer (2-13)	Läser av status (digitalt värdet) av en pin. Bör deklarerats som ingång innan den läses.
analogWrite(a,duty)	a: arduino-objektet Duty: 8-bit PWM signal 0..255	PWM-utgång, låst till DAC1. Använd motorshielden för att styra motorer!!
analogRead(a,pin)	a: arduino-objektet pin: 'A0', 'A1', ('A3')	Läser av en analog pinne. Skall ej deklarerats som ingång innan läsning. AD-upplösning 1024bit

```
>> a=arduino_com('COM4')
Attempting connection .....
Connection successful!

a =

Connected to COM4 port

fx >>
```

Exempel av nyckelfrågor i samband med reflektioner och utvärdering av det eget lärande

- Är du nöjd med samarbetet i gruppen? Vad fungerade bra vad mindre bra?
- Hur fungerade beslutsprocessen i gruppen? Kunde det ha varit bättre?
- Vilken rolluppdelning valde ni? Hur kändes det?

- Hur effektivt och systematiskt var ni? På vilket sätt finns utrymme till förbättring?
- Hur uppfattade ni sambandet mellan teori och tillämpning?
- Hur uppfattade ni sambandet mellan föreläsning och laboration?
- Genomförde ni uppgifterna i ordningen som de står i rapporten eller i vilken ordning besvarade ni frågorna? Hur kändes det så som ni gjorde?
- Vad gjorde ni när det blev problem och/eller ni inte kom vidare? Använde ni både samma strategi? Hur kändes det?
- Hur organiserade ni er inför slutinlämningen? Gick ni igenom alla frågor en gång till gemensam eller lämnade ni bara in allt när ni kände att ni var färdiga?
- Planerade ni fasta tider när ni gemensam i gruppen genomförde laborationsuppgifterna eller hur organiserade ni er?
- ??