

Finn Haugen (finn@techteach.no)

18. november, 2008

Artikkelserien *Reguleringsteknikk*

Dette er artikkel nr. 7 i artikkelserien *Reguleringsteknikk*:

- **Artikkel 1:** Reguleringsteknikkens betydning og grunnprinsipp. (AMNYTT nr. 1, 2007)
- **Artikkel 2:** PID-regulatoren: Oppbygning og virkemåte. (AMNYTT nr. 4, 2007.)
- **Artikkel 3:** Innstilling (tuning) av PID-regulatorer. (AMNYTT nr. 1, 2008)
- **Artikkel 4:** Reguleringsstrukturer som bygger på PID-regulatoren (AMNYTT nr. 2, 2008)
- **Artikkel 5:** Modellbasert regulering – foroverkopling. (AMNYTT nr. 3, 2008)
- **Artikkel 6:** Modellbasert regulering – optimalregulering og prediktiv regulering (AMNYTT nr. 4, 2008)
- **Artikkel 7: Softsensor (Kalman-filter)** (AMNYTT, nr. 5, 2008 – foreliggende AMNYTT)
- **Artikkel 8:** Et kritisk blikk på utdanningen innen reguleringsteknikk og automatisering generelt. (Innspill fra lærere, studenter og industriefolk.)

Artikkelserien skrives av Finn Haugen. Han er sivilingeniør fra Institutt for teknisk kybernetikk, NTNU (NTH). Han har nå deltidsstilling som 1. am. ved Høgskolen i Telemark og har også en deltidsstilling ved Høgskolen i Buskerud. Han har i en årrekke undervist høgskolefag, nettkurs og industrikurs innen modellering, simulering og regulering. Driver konsulentvirksomhet og applikasjonsutvikling gjennom TechTeach. Han er forfatter av diverse lærebøker om dynamiske systemer og reguleringsteknikk og tutorials for LabVIEW og MATLAB. Han har også utviklet simulatorbiblioteket **SimView**. Han holder kurs i PID-regulering og simulatorutvikling – såvel eksterne som interne kurs. Mer info fins på <http://techteach.no>.

Softsensor (Kalman-filter)

Innledning

Prosessmålinger brukes til regulering og overvåking. Prosessvariable måles vanligvis med fysiske sensorer. En softsensor er i stedet basert på software og er i form av et program som *beregner* verdier av prosessvariabelen. Hensikten med en softsensor kan være:

- Å erstatte en fysisk sensor ut fra vurderinger knyttet til kostnad eller tilgjengelighet. (De beregnede målingene i softsensoren kan utnyttes til forbedret regulering.)
- Å dublisere en fysisk sensor for å erstatte fysiske målinger dersom den fysiske sensoren feiler.

Denne artikkelen demonstrerer anvendelser av softsensorer implementert som et Kalman-filter, som generelt er en algoritme for tilstandsestimering som ble publisert av Rudolf Kalman i 1960. Noen alternative algoritmer for implementering av softsensorer er nevrale nettverk og multivariabel kalibrering (kjemometri).

Artikkelen bygger på en praktisk applikasjon med en likestrømsmotor, se figur 1.



Figur 1

Motorens turtall måles med et tachometer som er montert på motorakslingen. I applikasjonen skal softsensoren benyttes slik:

- **Til foroverkopling fra prosessforstyrrelsen.** Lastmomentet utgjør en prosessforstyrrelse på motoren. Lastmomentet oppstår ved at brukeren (forfatteren :-)) bremses motoren med hånden via lastskiven. Foroverkopling fra lastmomentet vil kunne forbedre turtallsreguleringen, dvs. redusere reguleringsavviket, ved at pådraget (styresignalet) baseres på informasjon om lastmomentets nåværende verdi. Momentet måles egentlig ikke, men beregnes, dvs. estimeres, av softsensoren.
- **Til forbedring av reguleringsystemets robusthet overfor manglende sensorsignaler.** Anta at turtallsmåleren av en eller annen grunn faller ut slik at målesignalet blir null. Hvis regulatoren fortsetter å regulere ut fra feilaktig turtallsverdi, kan det faktiske turtallet få en uheldig – generelt skadelig - verdi. Softsensoren beregner en til enhver tid fornuftig turtallsverdi ut fra prosessens (motorens) matematiske modell. I denne situasjonen vil Kalman-filteret fungere som en ren prosessimulator, som serverer simulerte turtalls"målinger" til regulatoren, som dermed kan fortsette sitt arbeid som normalt, uten at en trenger å stoppe motoren eller å kjøre regulatoren i manuell. En liknende situasjon er at målesignalet samples *sjelden* slik at regulatoren får sjeldne måleverdier som grunnlag for reguleringen. En softsensor kan da beregne hyppigere målesignaler for vil foreligge mellom de sjeldne måletidspunktene, til fordel for reguleringen.

En konkret industriell anvendelse av ovennevnte punkter er *dynamisk posisjonering* eller Dynpos (Kongsberg Maritime). Der estimeres vannets strømningshastighet for bruk i foroverkopling i posisjonsreguleringssystemet for fartøyene. Og de øvrige tilstandene posisjon og hastighet estimeres for bruk i situasjoner der det er manglende (feilaktige eller sjeldne) sensorsignaler fra f.eks. satellittbaserte sensorer.

Kalman-filteret er basert på en matematisk modell av prosessen. I vår applikasjon er modellen som følger:

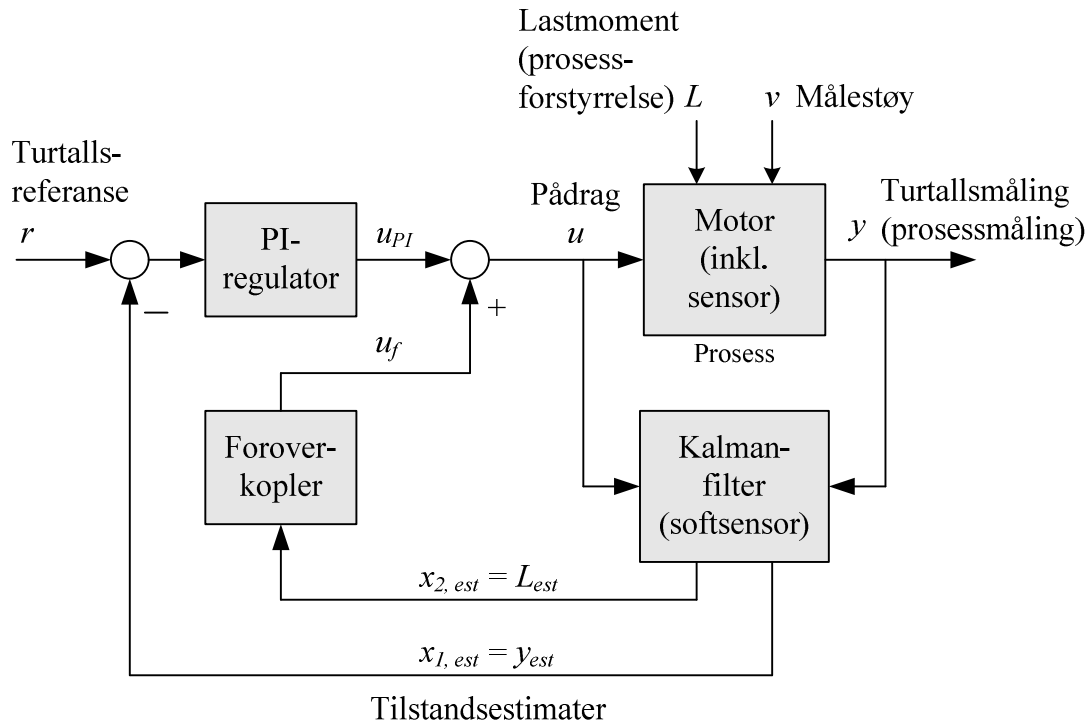
$$T \cdot dy/dt = -y(t) + [K \cdot u(t) + L(t)]$$

der y er turtall representert i måleenheten volt, u er styresignalet i volt, L er lastmomentet i ekvivalent enhet volt, $T = 0,25$ sek er motorens tidskonstant, og $K = 0,8$ er motorens forsterkning. Siden lastmomentet i praksis virker bremsende, får L en negativ tallverdi. Modellen er tilpasset den virkelige motoren med enkel prøving og feiling (justering av K og T inntil virkelig og simulert turtallsmåling ble nokså like).

Systemet (turtallsreguleringssystemet inkl. Kalman-filteret) er implementert i LabVIEW (National Instruments).

Regulator og softsensor (Kalman-filter) for motoren

Figur 2 viser strukturen av systemet bestående av motor, softsensor og regulator som igjen består av en PI-regulator og foroverkopler.



Figur 2

Foroverkopleren (fra forstyrrelsen, dvs. lastmomentet) kan finnes fra prosessmodellen ved å uttrykke pådraget som funksjon av forstyrrelsen. Resultatet blir

$$u_f(t) = (-1/K)*L(t)$$

L blir ikke målt direkte, men estimert med et Kalman-filter. Dermed blir foroverkopleren

$$u_f(t) = (-1/K)*L_{est}(t)$$

Kalman-filteret baseres på en tidsdiskret differenslikningsmodell som er funnet ved å diskretisere prosessmodellen gitt ovenfor.

Kalman-filteralgoritmen for vår applikasjon er angitt nedenfor. Det er den såkalte korrektor-prediktor-algoritmen som er implementert i C-kode i LabVIEW. Tidsindeks k angir nåværende tidsskritt, og $kp1$ angir neste tidsskritt (dvs. k pluss 1). y er turtallsmåling fra tachometeret, $x1$ er turtallsverdi, $x2$ er lastmoment, u er styresignal, K er forsterkning, T er tidskonstant, T_s er samplingstid. $K1$ og $K2$ er Kalman-filterforsterkningene - beregnet med Kalman Gain-funksjonen i LabVIEW. (Kalman-filteralgoritmen er nærmere beskrevet i f.eks. referanse [1].)

```
//Beregning av innovasjonssignalet e_k som differansen mellom
//faktisk hastighetsmåling og predikert hastighetsverdi:
```

```
e_k = y_k - x1_pred_k;
```

```
//Beregning av korrigerede estimater som målebasert oppdatering av predikerte estimater:
//(Det er de korrigerede estimatene som benyttes som estimater i applikasjoner.)
```

$$x1_corr_k = x1_pred_k + K1*e_k;$$

$$x2_corr_k = x2_pred_k + K2*e_k;$$

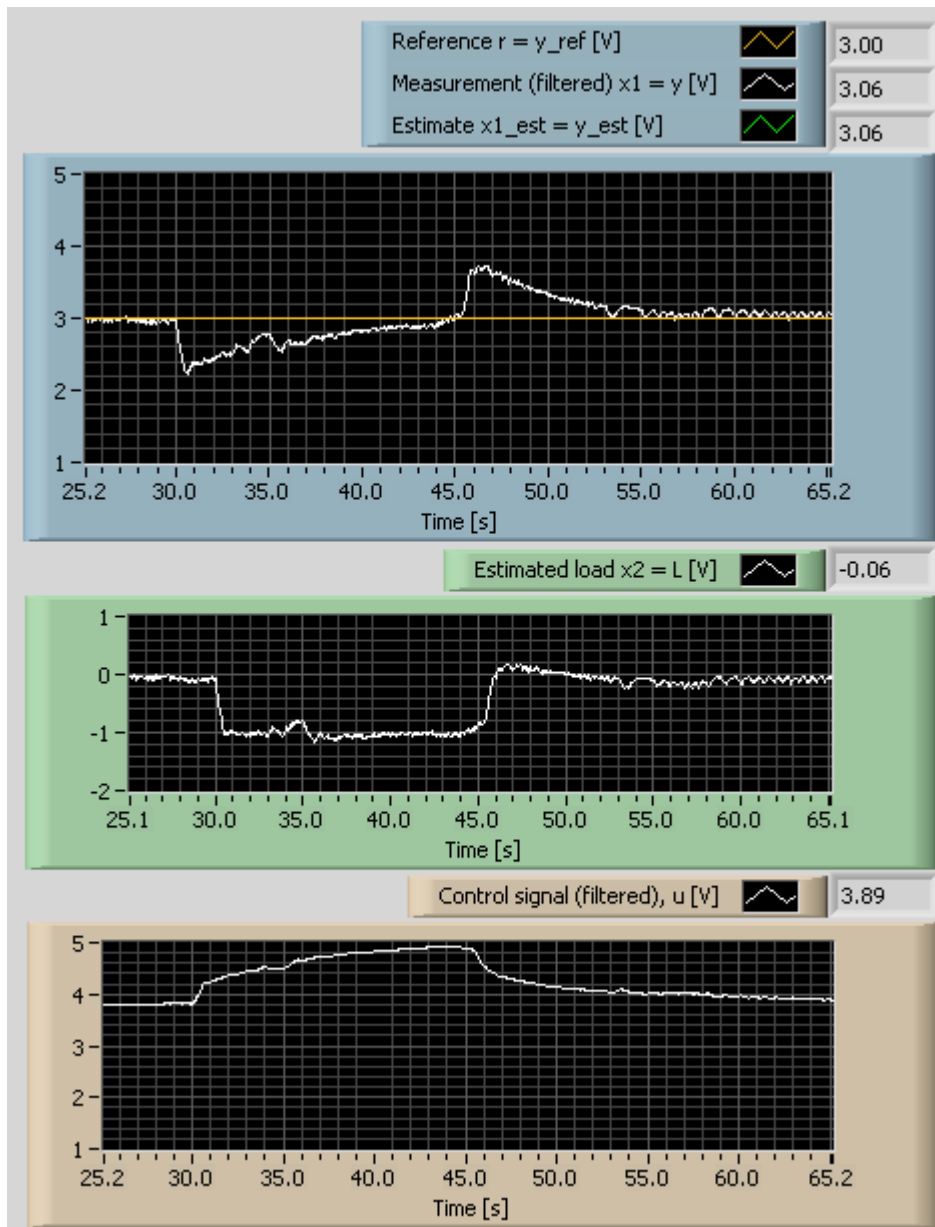
//Beregning av predikerte estimater for neste tidsskritt vha. tidsdiskret prosessmodell:

$$x1_pred_kp1 = (1-Ts/T)*x1_corr_k + (Ts/T)*x2_corr_k + (K*Ts/T)*u_k;$$

$$x2_pred_kp1 = x2_corr_k;$$

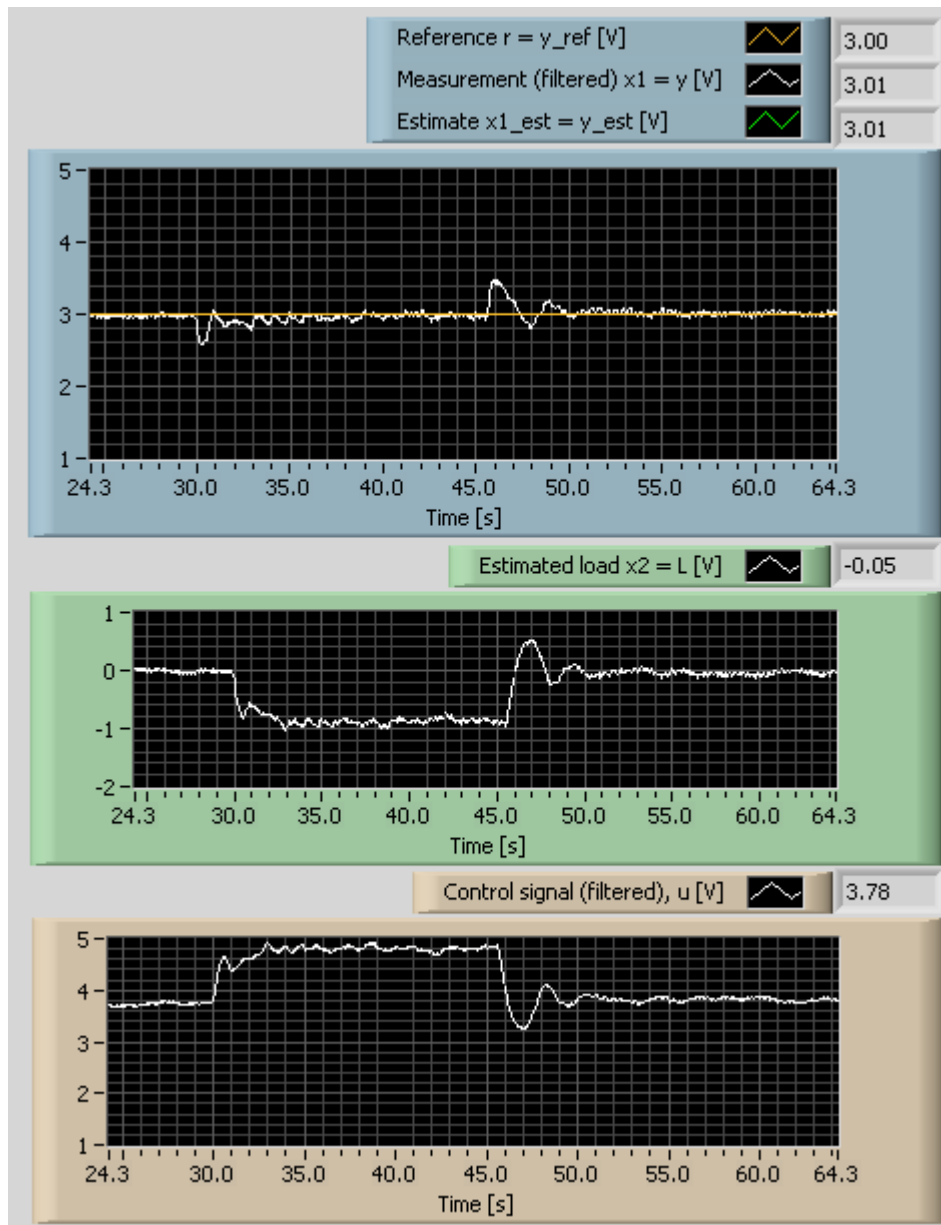
Eksperiment: Foroverkopling fra estimert lastmoment

For å se om det blir noen forbedring av reguleringen ved å bruke foroverkopling fra estimert lastmoment, kan vi først betrakte tilfellet *uten foroverkopling*, men med vanlig PI-regulering (med forsterking 0,5 og integraltid 2 sek – en nokså slapp regulatorinnstilling, gjort for å demonstrere effekten av foroverkopling). Figur 3 som viser turtallets forløp etter at motoren ble bremsset med et tilnærmet konstant lastmoment. Det estimerte lastmomentet (se den midterste grafen i figur 3) gir faktisk godt uttrykk for det påtrykte lastmomentet. Vi ser at regulatoren klarer å få turtallet tilbake til referansen igjen, med etter en viss tid. Jeg har beregnet den såkalt IAE-indeksen av forsøket (IAE = integral of absolute value of control error) over ca 30 sekunder. IAE er altså integralet av reguleringsavviket over intervallet. Jeg fikk **IAE = 7,9**, som vi skal sammenlikne med tilfellet *med foroverkopling*, se nedenfor.



Figur 3

Figur 4 viser resultatet for et tilsvarende eksperiment *med foroverkopling*.



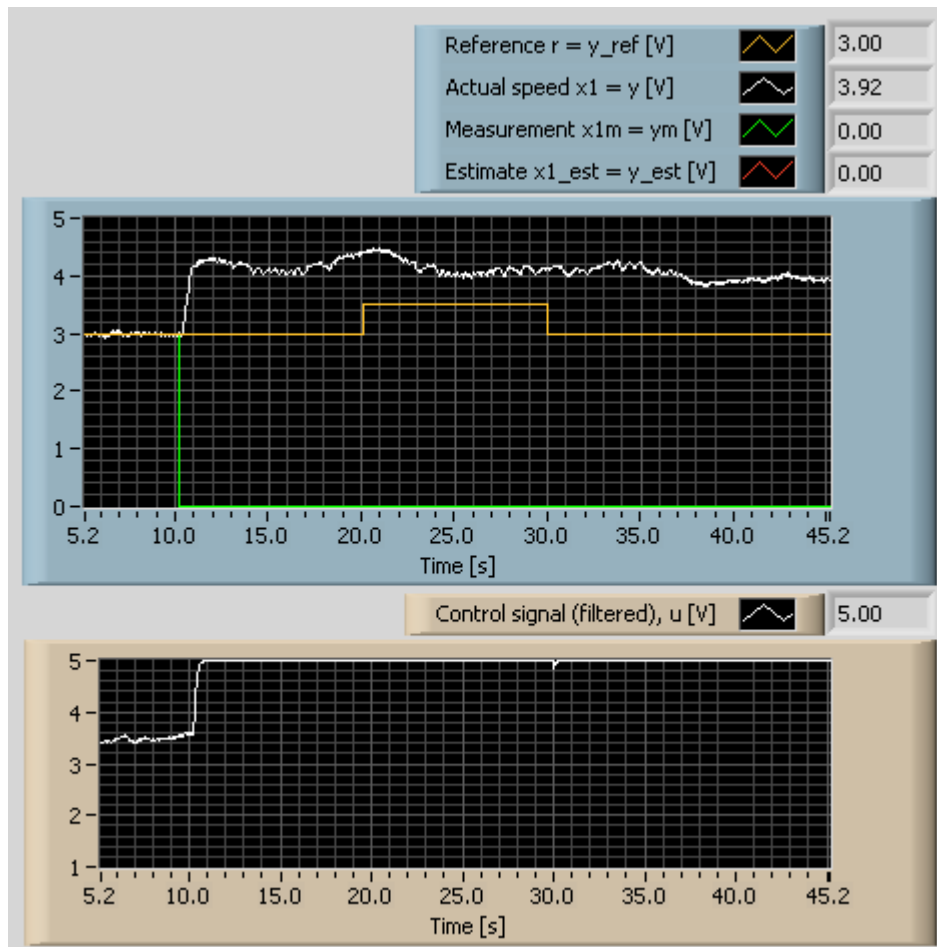
Figur 4

Fra figur 4 ser vi at reguleringen er mye bedre *med foroverkopling* fra softsensoren (Kalman-filteret). Resultatet ble **IAE = 1,9**, mye mindre enn uten foroverkopling (7,9). Vi kan si fire ganger bedre!

Et annet viktig poeng, i tillegg til at reguleringen er forbedret, er at vi har fått et ganske godt estimat av den ikke-målte forstyrrelsen. Generelt kan slike estimater være nyttige for overvåkingsformål.

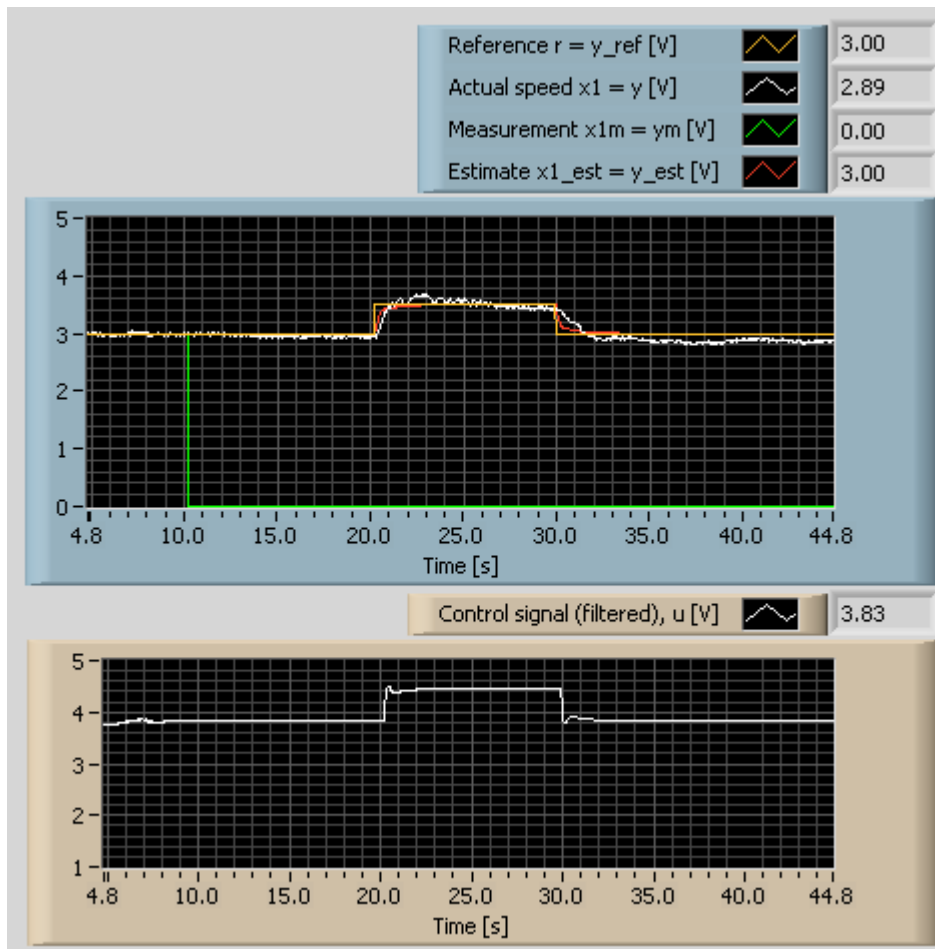
Eksperiment: Robusthet overfor utfall av sensorsignal

I det første eksperimentet her benytter PI-regulatoren turtallmålingen fra tachometeret. Anta at denne turtallmålingen faller ut – her: blir null. Da vil regulatoren tro at motoren har stoppet og (fortvilet) øke pådraget til maksimal verdi i et forsøk på å bringe turtallet opp til referansen igjen. Dette medfører at motoren ruser til maksimalt turtall, godt over referansen. Dette er demonstrert i figur 5. Den grønne kurven er den feilaktige turtallmålingen som regulatoren mottar. Turtallsreferansen ble også øket fra 3 til 3,5 og deretter redusert til 3 igjen. Det er jo tydelig at reguleringen ikke fungerer.



Figur 5

Også i det neste eksperimentet ble turtallsmålingen koplet ut. Men softsensoren ble programmert til å fortsette å beregne eller simulere turtallet selv om turtallsmålingen er falt ut. (Egentlig ble ikke målingen koplet ut i eksperimentet, men et signal som representerer målingen, ble satt til null. Jeg ville jo fortsette å plote turtallsmålingen for å få demonstrert at reguleringen fortsatte uten særlige problemer.). Siden det beregnede turtallet brukes av regulatoren, fortsetter regulatoren å motta vettuge turtallsverdier, og reguleringen fortsetter tilnærmet som normalt. Dette framgår av figur 6. Den grønne kurven er fremdeles den feilaktige turtallsmålingen, men den benyttes nå ikke av regulatoren, som altså i stedet benytter det simulerte turtallet.



Figur 6

Konklusjon

En softsensor i form av et Kalman-filter, som er en *modellbasert* algoritme for estimering av prosessstilstander og prosessforstyrrelser, kan utnyttes til forbedret regulering gjennom f.eks. foroverkopling fra estimert prosessforstyrrelse og til økt robusthet overfor utfall av sensorsignalet.

[1] F. Haugen: *Kompendium for Kybernetikk 2 ved Høgskolen i Oslo*, 2008.
(<http://techteach.no>)