

ISSN 0280-5316
ISRN LUTFD2/TFRT--5497--SE

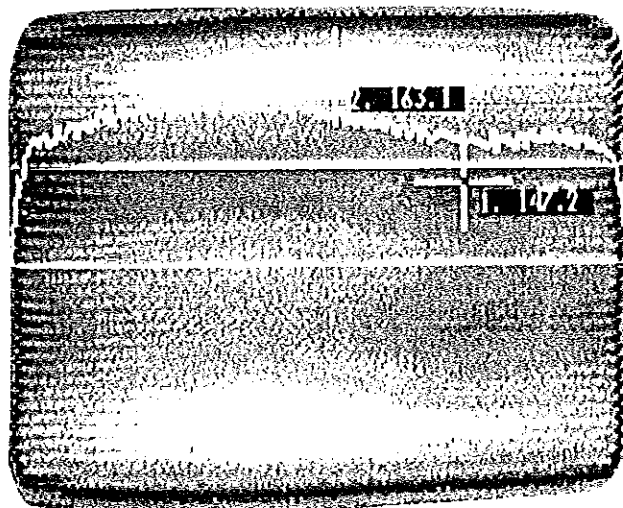
Temperaturreglering av Förpackningsmaskin

Rikard Hesslevik
Pehr Hjalmarsson

Institutionen för Reglerteknik
Lunds Tekniska Högskola
Februari 1994

Department of Automatic Control Lund Institute of Technology P.O. Box 118 S-221 00 Lund Sweden	<i>Document name</i> Master thesis	
	<i>Date of issue</i> February 1994	
	<i>Document Number</i> ISRN LUTFD2/TFRT--5497--SE	
<i>Author(s)</i> Rikard Hesslevik, Pehr Hjalmarsson	<i>Supervisor</i> Jörgen Bengtsson, Tetra Pak, Tore Hägglund	
	<i>Sponsoring organisation</i>	
<i>Title and subtitle</i> Temperaturreglering av förpackningsmaskin (Temperature control of a packaging machine.)		
<i>Abstract</i> <p>The purpose of this project was to improve the control of the heating procedure for the plastic material which is a part of the new package called Tetra Rex Cliq.</p> <p>By process identification, a process model was first determined. Based on the model, a discrete state-feedback controller was designed. To get a faster start-up procedure and decrease the material loss, a prewarmer was added to the controller.</p> <p>The controller and the prewarmer was implemented in the machine with a module called TMCC (Tetra Pak Multipurpose Compact Controller) which is programmable in C++.</p> <p>The start-up time was decreased from 30 s to less than 10 s. The temperature variations in stationarity are the same as with the old PID controller (+/- 5°C). These large variations are caused due to hardware and measurement problems.</p>		
<i>Key words</i>		
<i>Classification system and/or index terms (if any)</i>		
<i>Supplementary bibliographical information</i>		
<i>ISSN and key title</i> 0280-5316		<i>ISBN</i>
<i>Language</i> Swedish	<i>Number of pages</i> 67	<i>Recipient's notes</i>
<i>Security classification</i>		

The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Fax +46 46 110019, Telex: 33248 lubbis lund.



Figur 4: Temperaturfördelning över plasten. De ljusa områdena har den högsta temperaturen.

4.3 Temperaturreglering

För att hålla en konstant temperatur i plasten krävs någon form av reglering. Detta innebär att temperaturen, vilket är det återkopplade ärvärdet, måste mätas kontinuerligt. Efter jämförelse mellan aktuell temperatur (ärvärdet) och önskad temperatur (börvärdet) beräknar regulatorn fram en styrsignal vilken styr den till lamporna matade effekten.

4.3.1 Temperaturmätning med pyrometer

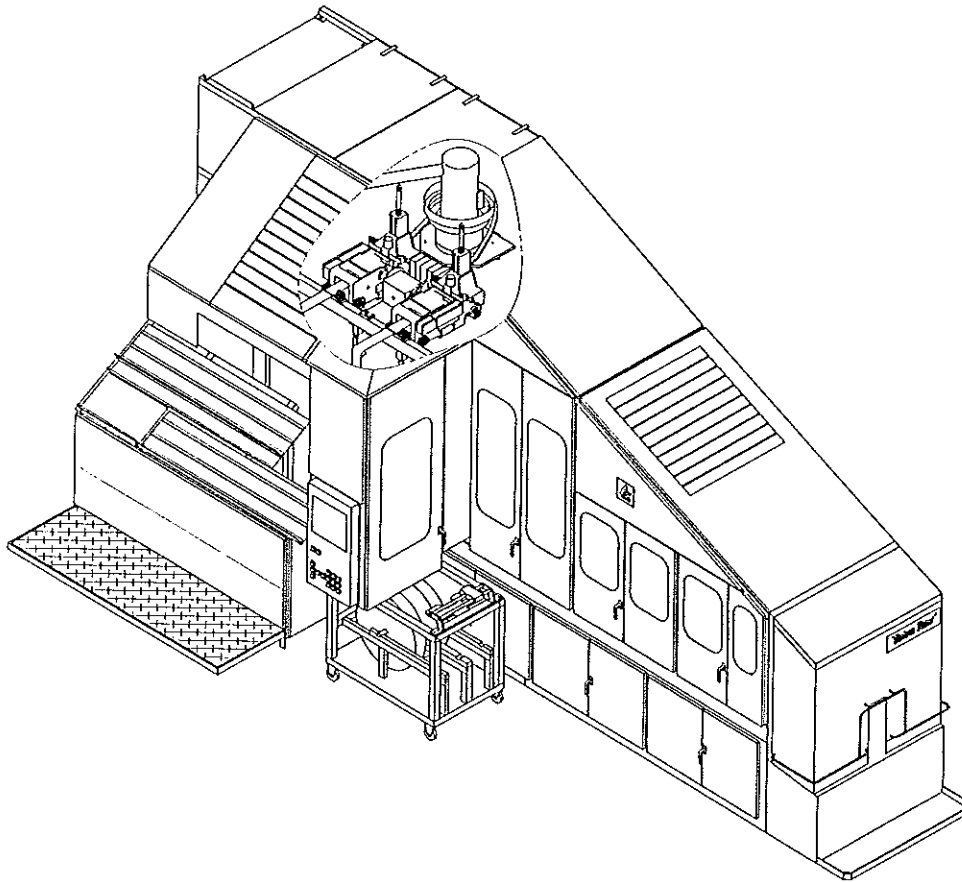
Då plasten är känslig mot deformationer i uppvärmt tillstånd krävs att mätningen är beröringsfri. Den metod som fungerat bäst hittills är temperaturmätning med pyrometer.

Den bygger på teorin att all materia med temperatur över den absoluta nollpunkten emitterar energi i form av elektromagnetisk strålning. Emissionen per ytenhet ökar med ökande temperatur.

För en strålare som emitterar all den strålning den teoretiskt sett kan emittera (svartkroppsstrålare) ges totalstrålningen av Stefan-Bolzmanns lag:

$$W(T) = \sigma T^4$$

4 Förpackningsmaskinens uppbyggnad



Figur 2: Förpackningsmaskin för Tetra Rex Cliq.

4.1 Maskinbeskrivning

Maskinen är fortfarande under utveckling vilket innebär att utrustning och konstruktion kommer att förändras.

Maskinen som tillverkar den nya förpackningen är i grund och botten en Tetra Rex maskin som har modifierats på vissa områden. Den stora förändringen har skett vid formningen av toppen, vilken sker i det inringade området i figur 2.

Formningen sker på följande sätt:

En icke transparent plastfilm (polystyren) matas in i maskinen där den värms upp till en temperatur då den är formbar. För polystyren är denna temperatur 145°C och kan variera $\pm 10^{\circ}\text{C}$. Därefter formas toppen direkt på förpackningen, där plastkorken används som pressverktyg. Med hjälp av tryckluft pressas den

2 Den nya förpackningen Tetra Rex Cliq

För att möta den allt hårdare konkurrensen inom förpackningsindustrin har Tetra Pak sedan en lång tid tillbaka arbetat med att utveckla en ny typ av förpackning.

I grunden är det en vanlig 1 liters mjölkförpackning, en så kallad Tetra Rex, som har försetts med en ny topp. Toppen är plan och försedd med en skruvkork, av plast, i centrum, se figur 1.



Figur 1: Tetra Rex Cliq förpackning.

En av fördelarna med detta är att förpackningen är återförslutningsbar och på så sätt kan den konkurrera med vanliga plastflaskor.

Toppen består av plast medan övriga förpackningen är uppbyggd av kartong med en pålagd plastfilm.

Innehåll

1	Sammanfattning	3
2	Den nya förpackningen Tetra Rex Cliq	4
3	Problemframställning	5
4	Förpackningsmaskinens uppbyggnad	6
4.1	Maskinbeskrivning	6
4.2	Uppvärmningsförlopp	7
4.3	Temperaturreglering	8
4.3.1	Temperaturmätning med pyrometer	8
4.3.2	Effektstyrning med tyristorer	10
4.3.3	Temperaturregulator	11
4.3.4	Blockschema över reglerutrustningen	11
4.4	Styrsystem TPMC (Tetra Pak Machine Controller)	12
5	Design av ny regulator	13
5.1	Identifiering av värmeprocessen	13
5.1.1	Minsta-kvadratmetoden	14
5.1.2	In- och utsignaldata	14
5.1.3	Bestämning av modellstruktur	16
5.2	Tillståndsåterkoppling	19
5.2.1	Polplacering med tillståndsåterkoppling	20
6	Implementering av den nya regulatorn i testriggen	25
6.1	TMCC-modul	26
7	Uppstart med förvärmning	28
7.1	Simulering av uppstartsforlopp med förvärmning	28
7.2	Implementering av förvärmning i TMCC-modulen	29
8	Resultat av den nya regleringen	32
9	Marknadsundersökning av uppvärmningselement	33

A	Tekniska data på värmeutrustning	37
A.1	Styrsystem, TPMC	37
A.2	TMCC-modul	37
A.3	Pyrometer	37
A.4	Tyristorer	38
A.5	IR-lampor	38
B	Beräkning av styrbarhetsmatrisen W_c	39
C	Linearitetskontroll av tyristorer	40
D	Beräkning av L-vektorn vid olika val av polplacering	41
E	Programlistningar	42
E.1	Simulering med tillståndsåterkoppling	42
E.1.1	Processmodell	42
E.1.2	Regulator utan integratorverkan	43
E.1.3	Regulator med integratorverkan	44
E.1.4	Ihopkoppling av process och regulator	45
E.2	Simulering med förvärmning	46
E.2.1	Processmodell inklusive förvärmning	46
E.2.2	Regulator med integralverkan vid förvärmning	47
E.2.3	Ihopkoppling av process och regulator vid förvärmning	48
E.3	Ladderprogram för tillståndsåterkoppling	49
E.4	Ladderprogram för maskinstyrning	55
E.5	C-program med förvärmning och tillståndsåterkoppling alternativt PI-regulator utan förvärmning.	57

1 Sammanfattning

Syftet med detta examensarbete var att förbättra regleringen av uppvärmningsprocessen för den plast som är en del av den nya förpackningen Tetra Rex Cliq.

Genom processidentifiering bestämdes en modell, vilken låg till grund för en diskret tillståndsåterkopplad regulator. För att få ett snabbare uppstartsforlopp och på så sätt minska materialsplet, infördes en förvärmning.

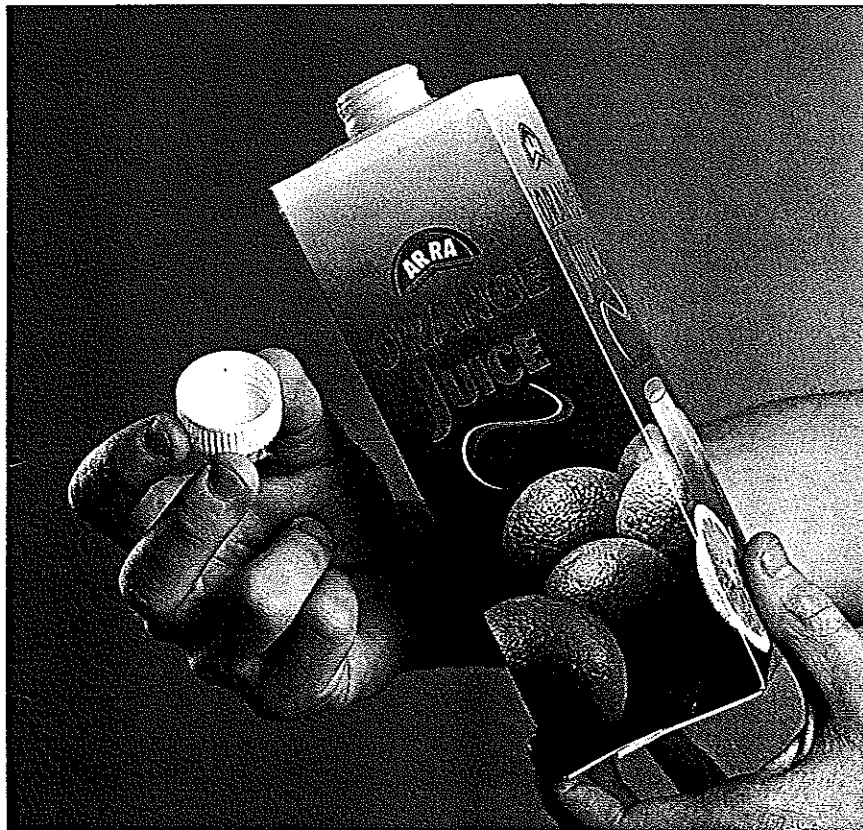
Regulatorn och förvärmningen implementerades i maskinen med en TMCC-modul (Tetra Pak Multipurpose Compact Controller) som programmerades i C++.

Uppstartstiden minskades från ca 30 s till under 10 s. Temperaturvariationerna i stationaritet minskade ej nämnvärt och är nu $\pm 5^{\circ}\text{C}$, denna till synes stora variation grundar sig i hårdvarumässiga och mättekniska problem.

2 Den nya förpackningen Tetra Rex Cliq

För att möta den allt hårdare konkurrensen inom förpackningsindustrin har Tetra Pak sedan en lång tid tillbaka arbetat med att utveckla en ny typ av förpackning.

I grunden är det en vanlig 1 liters mjölkförpackning, en så kallad Tetra Rex, som har försetts med en ny topp. Toppen är plan och försedd med en skruvkork, av plast, i centrum, se figur 1.



Figur 1: Tetra Rex Cliq förpackning.

En av fördelarna med detta är att förpackningen är återförslutningsbar och på så sätt kan den konkurrera med vanliga plastflaskor.

Toppen består av plast medan övriga förpackningen är uppbyggd av kartong med en pålagd plastfilm.

3 Problemframställning

Det problem vi fick till uppgift att lösa berörde uppvärmning av plasten vilken ingår i förpackningens topp.

Uppvärmningsprocessen består av en ugn med 6 st IR-lampor vilka värmer upp en plastfilm. Plastfilmen stegas fram periodiskt, 4 steg i ugnen och strax därefter mäts temperaturen. Processen är således naturligt diskret och innehåller en tidsfördröjning.

I dagens lösning kräver uppvärmning och stabilisering av temperaturnivån i plasten cirka 30 s. Detta medför problem vid uppstart av maskinen eftersom materialsplet blir stort. Ytterligare problem idag är att temperaturen i plasten varierar med cirka $\pm 6^\circ\text{C}$ i stationaritet.

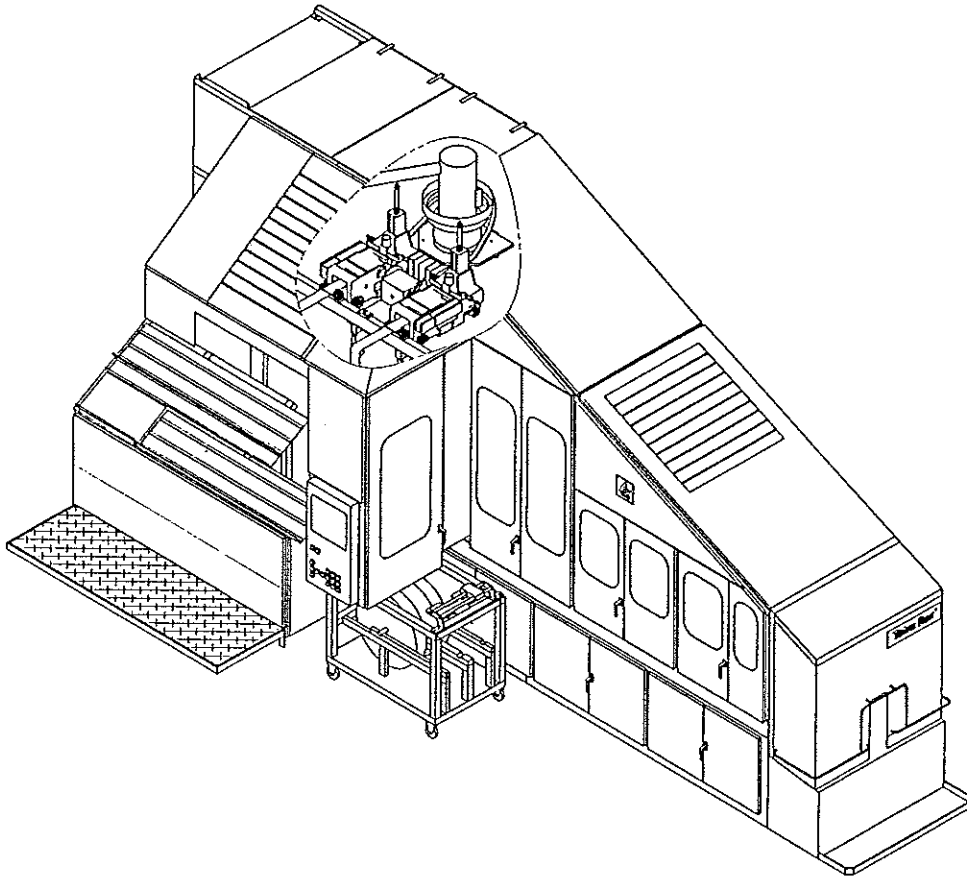
Vår uppgift bestod i att genom mätningar på uppvärmningsprocessen (öppna systemet) identifiera en överföringsfunktion och m h a denna konstruera en ny regleralgoritm och på så vis minska temperaturvariationerna i stationaritet till $\pm 1^\circ\text{C}$.

Vi ville även implementera en förvärmning av materialet vid uppstart vilket möjliggör ett snabbare startförlopp.

För att underlätta arbetet med vårt projekt fanns det på Tetra Pak en testrigg till vårt förfogande. Denna bestod av en uppvärmningsprocess, i stort sett identisk med den i maskinen.

Parallellt med designen av den nya regulatorn fick vi till uppgift att göra en mindre marknadsundersökning av värmeelement för uppvärmning av plasten.

4 Förpackningsmaskinens uppbyggnad



Figur 2: Förpackningsmaskin för Tetra Rex Cliq.

4.1 Maskinbeskrivning

Maskinen är fortfarande under utveckling vilket innebär att utrustning och konstruktion kommer att förändras.

Maskinen som tillverkar den nya förpackningen är i grund och botten en Tetra Rex maskin som har modifierats på vissa områden. Den stora förändringen har skett vid formningen av toppen, vilken sker i det inringade området i figur 2.

Formningen sker på följande sätt:

En icke transparent plastfilm (polystyren) matas in i maskinen där den värms upp till en temperatur då den är formbar. För polystyren är denna temperatur 145°C och kan variera $\pm 10^{\circ}\text{C}$. Därefter formas toppen direkt på förpackningen, där plastkorken används som pressverktyg. Med hjälp av tryckluft pressas den

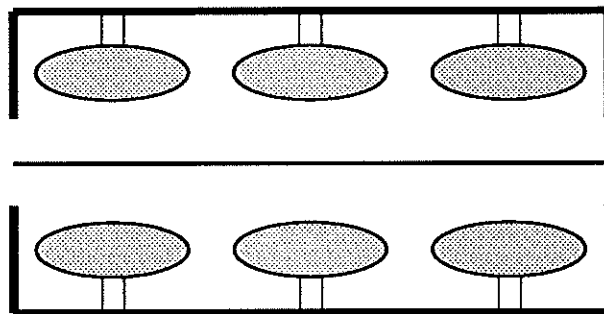
formbara plasten in i korken och på så sätt bildas gängorna. Nästa steg är att fästa plasten i korkens botten, detta sker med ultraljud, för att kunna bryta upp förpackningen första gången den skall öppnas. Nu är förpackningen klar för påfyllning och bottenförslutning, vilket sker med beprövade metoder som vi inte går närmare in på.

4.2 Uppvärmningsförlopp

Tekniken idag bygger på att plasten värms upp med infraröda lampor av quartz-typ. Dessa lampor sänder ut strålning inom det kortvågiga infraröda området ($0.76\mu\text{m}$ - $1.5\mu\text{m}$). Fördelarna med denna typ av värmare är flera:

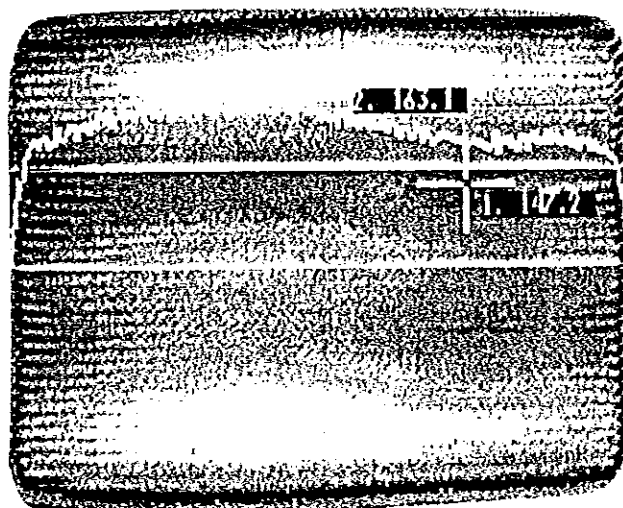
- *hög verkningsgrad*, eftersom värmen överförs genom strålning kan endast fasta kroppar absorbera denna och ingen effekt försvinner genom att omkringliggande luft värms upp.
- *snabb värmeöverföring*, vilket medför att stora värmemängder kan förflyttas under kort tid.

Den befintliga värmeutrustningen består av 6 st IR-lampor på vardera 800 W vid 220 V. Tre är placerade ovan plasten och tre undertill, i längsgående riktning. Se figur 3.



Figur 3: Tvärsnitt av ugnen som visar IR-lampornas placering.

En stor nackdel med detta val av lampor och deras placering är att värmefördelningen över plastens yta blir ojämn, se figur 4.



Figur 4: Temperaturfördelning över plasten. De ljusa områdena har den högsta temperaturen.

4.3 Temperaturreglering

För att hålla en konstant temperatur i plasten krävs någon form av reglering. Detta innebär att temperaturen, vilket är det återkopplade ärvärdet, måste mätas kontinuerligt. Efter jämförelse mellan aktuell temperatur (ärvärdet) och önskad temperatur (börvärdet) beräknar regulatorn fram en styrsignal vilken styr den till lamporna matade effekten.

4.3.1 Temperaturmätning med pyrometer

Då plasten är känslig mot deformationer i uppvärmt tillstånd krävs att mätningen är beröringsfri. Den metod som fungerat bäst hittills är temperaturmätning med pyrometer.

Den bygger på teorin att all materia med temperatur över den absoluta nollpunkten emitterar energi i form av elektromagnetisk strålning. Emissionen per ytenhet ökar med ökande temperatur.

För en strålare som emitterar all den strålning den teoretiskt sett kan emittera (svartkroppsstrålare) ges totalstrålningen av Stefan-Bolzmanns lag:

$$W(T) = \sigma T^4$$

$W(T)$ = den totala strålningsenergin i W/m^2 som ytan utsänder i en halvsfär per tids- och ytenhet.

σ = Stefan-Bolzmanns konstant, $5.67 \cdot 10^{-8} Wm^{-2}K^{-4}$

Den spektrala strålningen för en svart kropp ges av Plancks strålningslag:

$$W(\lambda) = \frac{C_1}{\lambda^5 (e^{\frac{C_2}{\lambda T}} - 1)}$$

$W(\lambda)$ = spektral strålningsenergi som ytan utsänder i en halvsfär per tid- och ytenhet, W/m^2

λ = våglängden, m

T = absoluta temperaturen, K

C_1 = 37415, Wm^2

C_2 = 14388, mK

Det framgår av Plancks och Stefan-Bolzmanns lagar att strålningens storlek och spektralfördelning beror av temperaturen, alltså kan temperaturen mätas hos en kropp genom att mäta strålningen som den emitterar.

Pyrometrin har dock en nackdel som har sin orsak i att knappast något material är en perfekt svartkroppsstrålare.

Då plasten inte är någon svartkroppsstrålare så måste detta kompenseras med en faktor, mellan 0-1, kallad emissionsfaktorn.

Emissionsfaktorn beror på flera faktorer:

- våglängden vid vilken vi betraktar strålningen.
- den strålande ytans temperatur.
- ytans material.
- ytans struktur.
- vinkeln under vilken ytan betraktas.

Totala emissionsfaktorn, ϵ_t , definieras som:

$$\epsilon_t = \frac{W_r}{W}$$

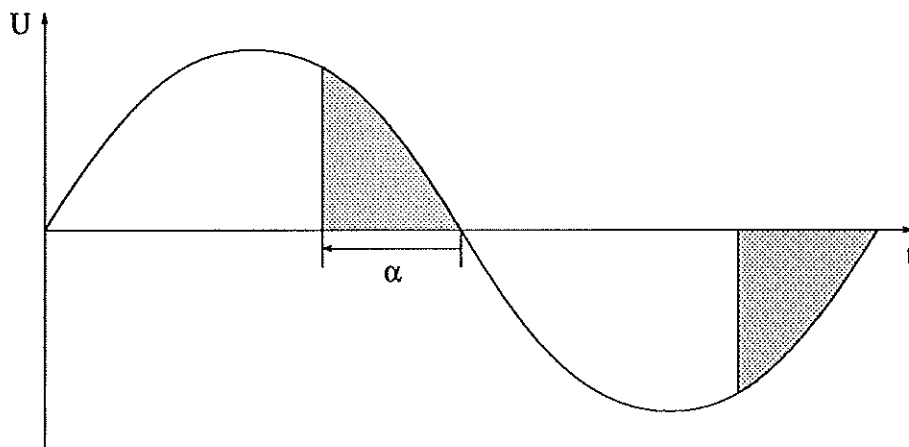
W_r = den totala energin från en verklig kropp

W = den totala utstrålade energin från en svartkropp vid samma temperatur som den verkliga kroppen

För tekniska specifikationer gällande den pyrometer som sitter i maskinen hänvisas till appendix A.

4.3.2 Effektstyrning med tyristorer

För att kunna reglera värmen från lamporna måste effekten till dessa varieras. Detta sker med tyristorer vilka kan ses som spänningsstyrda strömbrytare. Genom att variera styrvinkeln α , med en signal 0 - 10 V till tyristorerna, bestäms hur stor del av periodtiden de är öppna. Effekten till lamporna varierar linjärt med styrvinkeln. Se figur 5.



Figur 5: Fasvinkelstyrning av tyristorer. De skuggade områdena visar den del av perioden då tyristorn leder.

Att effekten har ett linjärt förhållande till styrvinkeln visas i appendix C.

För att styra effekten till de 6 lamporna används 3 st tyristorer, 2 lampor per tyristor och fas.

4.3.3 Temperaturregulator

Temperaturregleringen sker med en diskretiserad PID-regulator med följande algoritm:

$$u(k) = K_1 e(k) + K_2(I_{old} + e(k)) + K_3(e(k) - e(k - 1))$$

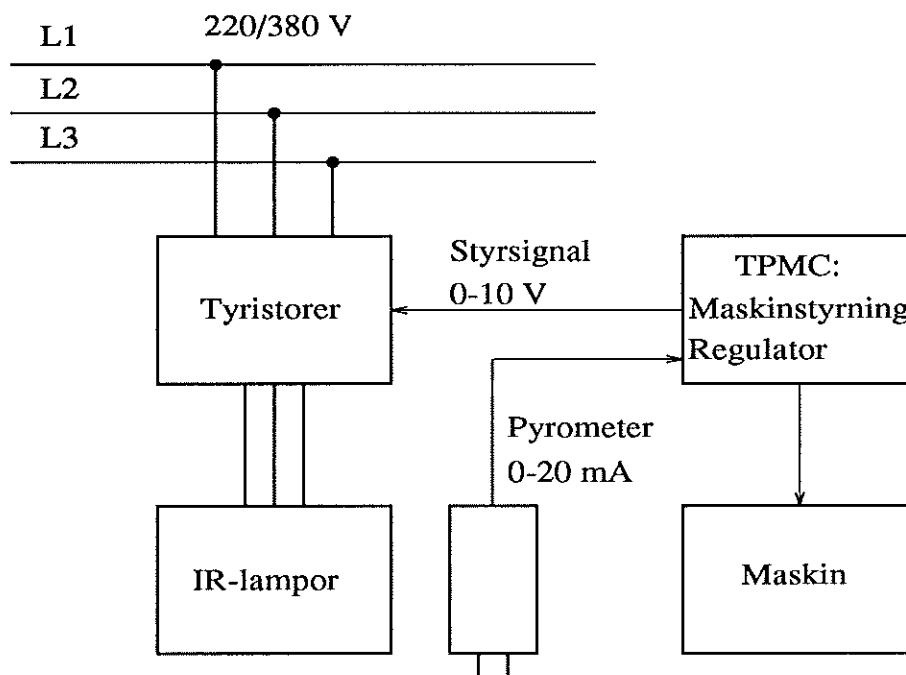
Där $K_1 = 20$, $K_2 = 8$, $K_3 = 2$, e = reglerfel, I_{old} = summan av tidigare fel och u = styrsignal.

Detta skall jämföras med standardalgoritmen för en diskret PID-regulator:

$$u(kh) = Ke(kh) + I_{old} + K \frac{h}{T_i} e(kh) + K \frac{T_d}{h} (e(kh) - e(kh - h))$$

Som framgår av de två styrlagarna ovan är P- och D-delen i stort sett desamma, den stora skillnaden ligger i I-delen. En stor nackdel med den befintliga algoritmen är svårigheterna med att trimma in regulatorn eftersom om en av konstanterna ändras, t ex K_1 påverkas indirekt T_i och T_d vilka ingår i K_2 och K_3 .

4.3.4 Blockschema över reglerutrustningen



Figur 6: Blockschema för temperaturreglering med kringutrustning.

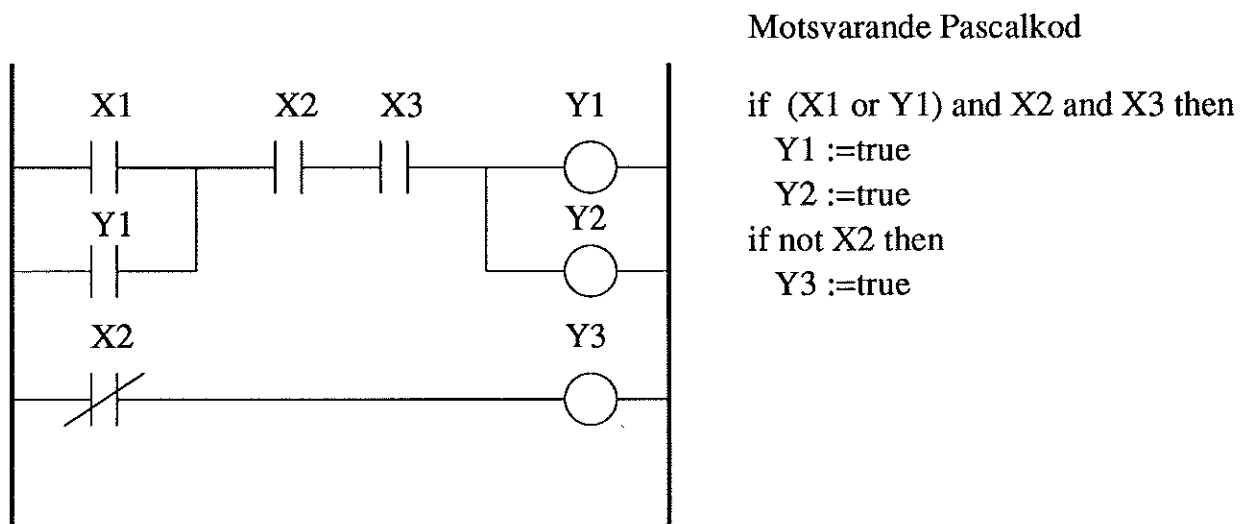
All teknisk data för temperaturregleringen redovisas i appendix A.

4.4 Styrsystem TPMC (Tetra Pak Machine Controller)

Styrsystemet som används i maskinen och där även PID-regulatorn ingår är ett PLC-system framtaget av Satt Control speciellt för Tetra Pak. PLC står för Programmable Logical Controllers vilket är en mikrodator utvecklad för industriellt bruk, och konstruerad för att utföra booleska operationer. TPMC är uppbyggt av olika moduler bl a spänningsförsörjningsenhet, CPU, analoga och digitala in- och utgångar. De analoga ingångarna är på 10 bitar, medan utgångarna är på 8 bitar.

Programspråket är uppbyggt av ladder-diagram, se figur 7, och exekveras cykliskt. Enkelt beskrivet fungerar en cykel i TPMC enligt följande:

- ingångarnas status läses av och sparas i arbetsminnet.
- programkoden exekveras och statusen på utgångarna registreras i arbetsminnet.
- alla utgångar uppdateras efter de nya värdena i arbetsminnet.



Figur 7: Enkelt exempel på ladderprogrammering.

5 Design av ny regulator

Alla maskinens rörelser sker intermittent. Detta får till följd att plastfilmen matas fram i steg om 7.3 cm med en periodtid om 1.2 s. Filmen värms upp under 4 cykler (4.8 s) och strax därefter mäts temperaturen. Processen innehåller alltså en tidsfördröjning och kan ses som en naturligt diskret process med en samplingstid på 1.2 s. En PID-regulator är olämplig vid långa tidsfördröjningar.

En bättre lösning är att använda en design med en diskret regulator, i vilken tidsfördröjningen lättare kan hanteras.

5.1 Identifiering av värmeprocessen

Det finns i princip två olika möjligheter att finna en approximativ beskrivning av värmeprocessen.

Den första metoden är att använda fysikens lagar och kända konstanter för att teoretiskt beräkna fram en matematisk modell, s k *modellbygge*. Detta är väl lämpat för enklare processer med klara fysikaliska samband. Den andra metoden kallas *identifiering*, vilket innebär att mäta in- och utsignaler och utifrån denna information bestämma en lämplig modell. Då de fysikaliska sambanden i värmeprocessen är komplexa är den senare metoden ett lämpligt val.

Antag att processen går att beskriva som ett linjärt system. Vi kan då ansätta följande differensekvation:

$$A(q)y(k) = B(q)u(k)$$

där u är insignal, y utsignal och A - respektive B -polynomen innehåller de okända parametrarna vilka skall skattas.

Det som krävs för att skatta parametrarna i modellen vid identifiering är:

- in- och utsignaldata från värmeprocessen.
- en modellstruktur.
- ett kriterium för att bestämma hur väl modellen överensstämmer med verkligheten.

Vi beslutade att använda *minsta-kvadratmetoden* för att skatta parametrarna.

5.1.1 Minsta-kvadratmetoden

Den skattade utsignalen \hat{y} skall efterlikna den verkliga utsignalen y . Ett mått på skattningen kan fås av kvadratsumman på felen, $y - \hat{y}$, den s k förlustfunktionen, $J(\Theta)$. Det är denna funktion som skall minimeras för bästa parameterskattning. Förlustfunktionen definieras:

$$J(\Theta) = \frac{1}{2} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

Inför beteckningarna:

$$\Theta = [a_1 \dots a_n b_1 \dots b_m]^T$$

$$\varphi^T(k) = [-y(k-1) \dots -y(k-n) u(k-1) \dots u(k-n)]$$

$$\Phi = \begin{bmatrix} \varphi^T(n+1) \\ \vdots \\ \varphi^T(N) \end{bmatrix}$$

I beteckningarna ovan kan nämnas att $a_1 \dots a_n$, respektive $b_1 \dots b_m$ är koefficienter i polynomen som skall skattas. Enligt teorin kan ett samband ställas upp för att bestämma skattningen av parametrarna, $\hat{\Theta}$. Förlustfunktionen blir minimal för $\hat{\Theta}$ då:

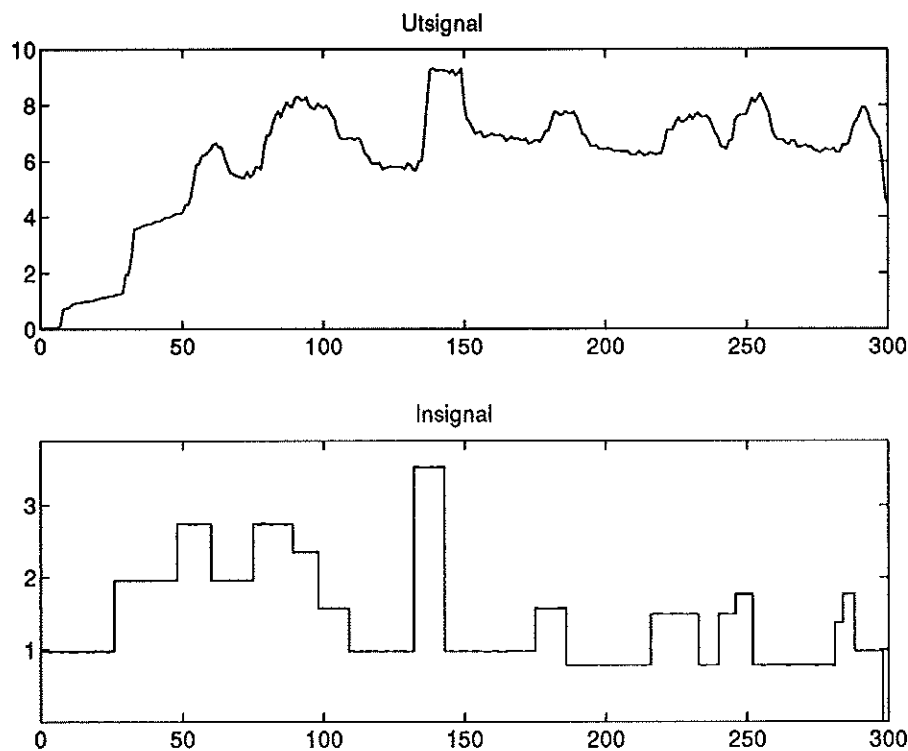
$$\hat{\Theta} = (\Phi^T \Phi)^{-1} \Phi^T y$$

5.1.2 In- och utsigndata

För att få en perfekt skattning av parametrarna skall insignalen till processen vara vitt brus. Då det är svårt att generera vitt brus valde vi att använda oss av en PRBS (pseudo random binary sequence) liknande insignal.

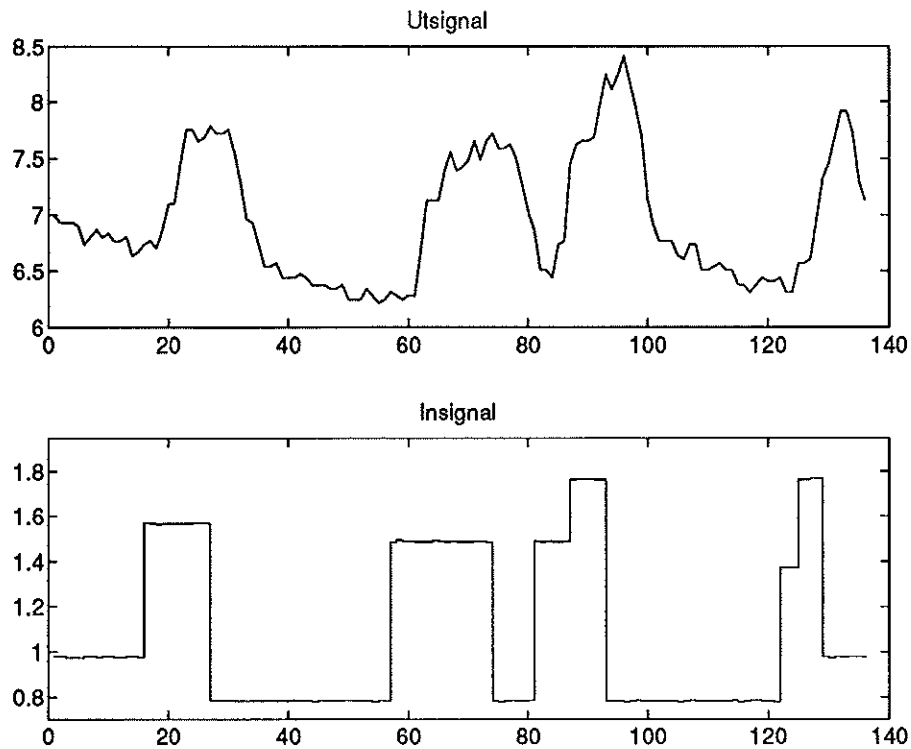
Mätdata insamlades från testtriggen med hjälp av en PC. Vi varierade spänningen till IR-lamporna mellan 1–3 V (insignal) och registrerade temperaturen på plastfilmen (utsignal) med pyrometern. Eftersom mätutrustningen endast kunde hantera spännings signaler omvandlades pyrometers strömsignal (0–20 mA) med ett 100 Ω motstånd till en spänningsnivå 0–2 V. Då nivåerna på in- och utsignalerna skall överensstämma (ligga mellan 0–10 V) med varandra medför det att temperatursignalen skall multipliceras med en faktor 5.

Totalt registrerades 1800 mätvärden med en sampeltid om 200 ms. Vi plockade därefter ut var 6:e sampel då processens naturliga sampeltid är 1.2 s. Mätresultatet åskådliggörs i figur 8.



Figur 8: In- och utsignal för värmeprocessen.

I figur 8 kan man se att pyrometersignalen bottnar kring sampel 140. Detta medför att dessa värden inte går att använda vid modellskattningen. Då vi är intresserade av en bra modell för reglering vid temperaturer kring 140°C (7.0 V) valde vi mätvärden från sampel 150 till 290. Dessa värden bildade Φ -matrisen. Se figur 9.

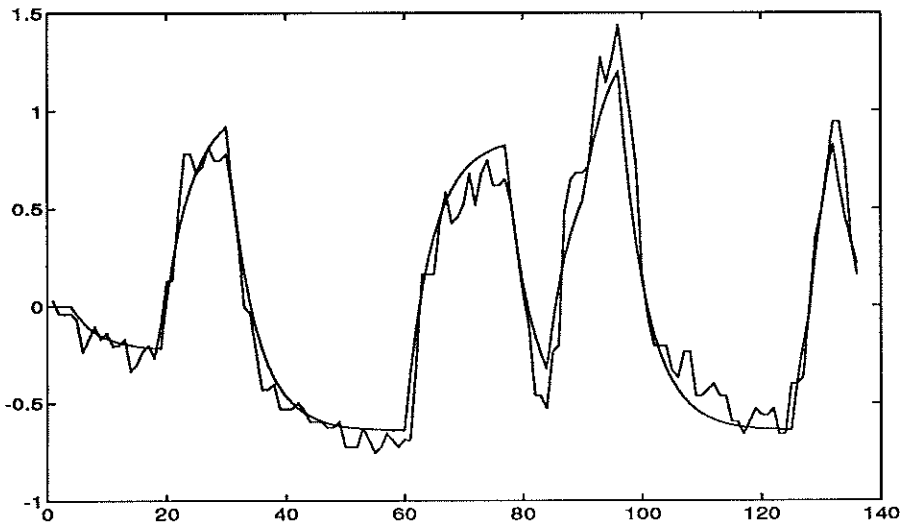


Figur 9: In- och ut signaler vilka ligger till grund för parameterskattningarna.

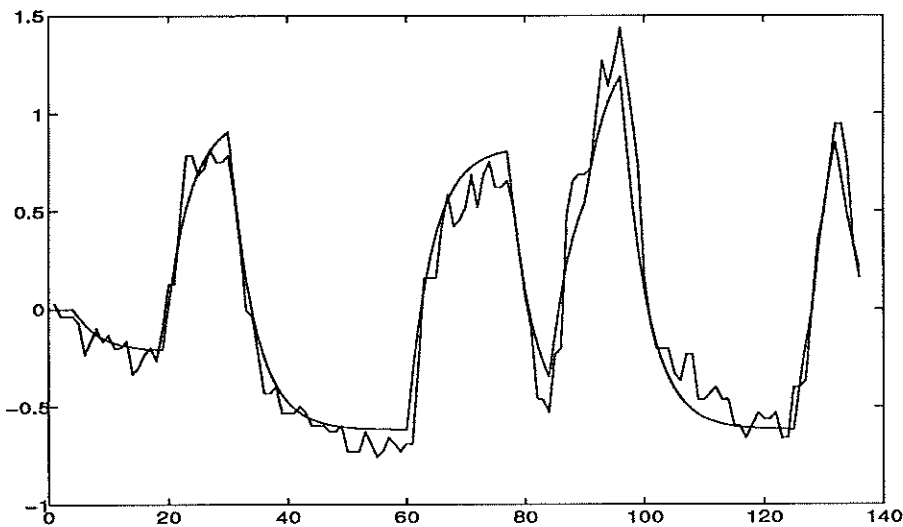
5.1.3 Bestämning av modellstruktur

Sista steget innan en skattningen av parametrarna kan ske blir att bestämma ordningen och tidsfördröjningen i systemet.

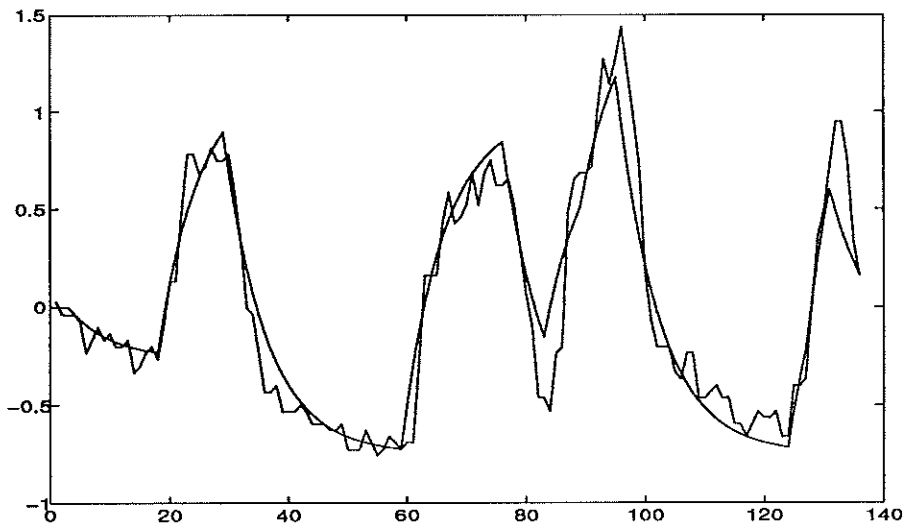
Vi provade olika kombinationer på modellens ordning och tidsfördröjning. Resultaten visas i figurerna 10 – 13.



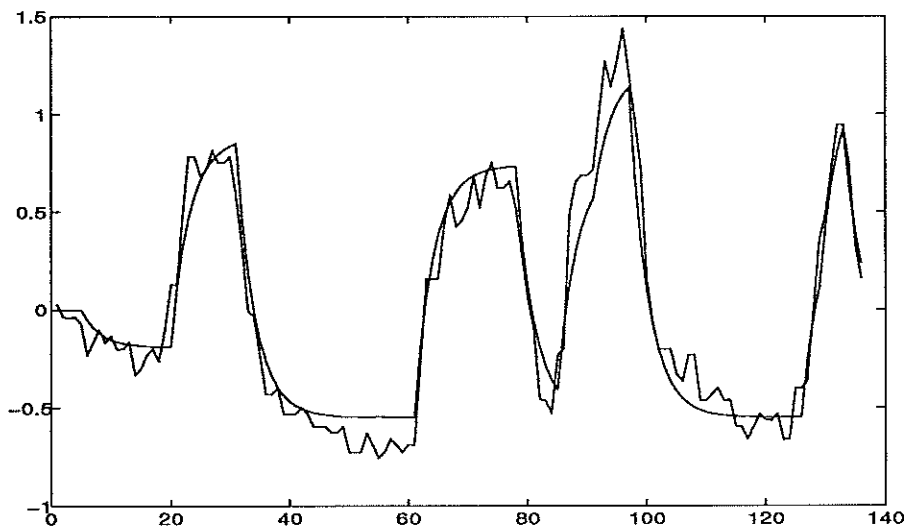
Figur 10: Jämförelse mellan verkliga processen och modell av 1:a ordningen med 4 sampels tidsfördröjning, förlustfunktionen $J(\Theta) = 0.01306$.



Figur 11: Jämförelse mellan verkliga processen och modell av 2:a ordningen med 4 sampels tidsfördröjning, förlustfunktionen $J(\Theta) = 0.01287$.



Figur 12: Jämförelse mellan verkliga processen och modell av 1:a ordningen med 3 sampels tidsfördröjning, förlustfunktionen $J(\Theta) = 0.01682$.



Figur 13: Jämförelse mellan verkliga processen och modell av 1:a ordningen med 5 sampels tidsfördröjning, förlustfunktionen $J(\Theta) = 0.01316$.

Vid en jämförelse mellan 1:a och 2:a ordningens system kunde vi konstatera att förändringen av förlustfunktionen, $J(\Theta)$, vid 2:a ordningen blev så ringa, att fördelarna med ett enklare system, färre tillstånd, övervägde. Vi valde således ett system av 1:a ordningen.

Därefter bestämdes lämplig tidsfördröjning på modellen. Vi jämförde förlustfunktionen, $J(\Theta)$, vid 3, 4 och 5 sampels fördröjning. Som framgår av figurerna 10, 12 och 13 är det mest lämpligt med 4 sampel.

Modellen fick då följande utformning:

$$\hat{y}(k) = -a_1\hat{y}(k-1) + b_4u(k-4)$$

Modellens överföringsfunktion:

$$H(q) = \frac{b_4}{1 + a_1q^{-1}}q^{-4}$$

För att skatta parametrarna a_1 och b_4 använde vi funktionen *ARX* i Matlab, vilken gör beräkningar efter minsta-kvadrat metoden. Efter beräkningarna kom vi fram till följande modell av överföringsfunktionen för värmeprocessen:

$$H(q) = \frac{0.4158}{1 - 0.8048q^{-1}}q^{-4}$$

Modellen ovan fick ligga till grund för det fortsatta arbetet med en ny regulator. Till vår hjälp hade vi programmen Matlab och Simnon.

5.2 Tillståndsåterkoppling

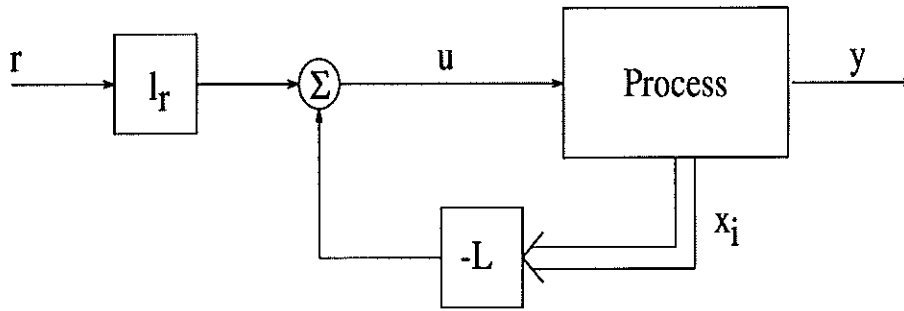
Som tidigare nämnts sker dagens reglering med en PID-regulator. Då processen innehåller en fördröjning om 4 sampel fungerar ej denna typ av regulator tillfredsställande.

Vår avsikt var att designa en diskret regulator för att på så sätt undvika problemet med tidsfördröjningen. Vi valde att använda en regulator som bygger på polplacering med tillståndsåterkoppling.

Fördelen med tillståndsbeskrivning är att det representerar allt som är värt att veta om systemets tidigare uppförande. Känner vi tillståndet $x(k)$ kan vi alltså bedöma effekten av olika insignaler.

En allmän styrlag för linjär tillståndsåterkoppling:

$$u(k) = -Lx(k) + l_r r(k)$$



Figur 14: Blockschema över tillståndsåterkoppling.

5.2.1 Polplacering med tillståndsåterkoppling

För att få full frihet med polplaceringen krävs att systemet är styrbart. Att detta kriterium är uppfyllt visas i appendix B.

Modellens överföringsfunktion på observerbar kanonisk tillståndsform:

$$\mathbf{x}(k+1) = \begin{bmatrix} 0.8048 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.4158 \end{bmatrix} u(k)$$

$$y(k) = [1 \ 0 \ 0 \ 0] \mathbf{x}(k)$$

På enklare form:

$$\mathbf{x}(k+1) = \Phi \mathbf{x}(k) + \Gamma u(k)$$

$$y(k) = C \mathbf{x}(k)$$

Styrlag:

$$u(k) = -L \mathbf{x}(k) + l_r r(k)$$

Med denna styrlag blir det slutna systemets överföringsfunktion:

$$H_c(q) = C(qI - (\Phi - \Gamma L))^{-1} l_r \Gamma$$

Den karakteristiska ekvationen blir således:

$$\det(qI - (\Phi - \Gamma L)) = 0$$

Värdena i L-vektorn bestäms av slutna systemets poler.

Vi provade först att placera samtliga poler i origo, s k deadbeat. Detta är den snabbast tänkbara regulatorn. Teoretiskt sett skall reglerfelet, efter en stegförändring i referenssignalen, bli noll efter lika många sampel som antalet tillstånd om den stationära förstärkningen är ett. Det kan uppnås genom att variera l_r beroende på polernas placering.

För att bestämma L-vektorn så att polerna hamnar i origo använde vi funktionen *acker* i Matlab. Denna bygger på Ackermann's formel. L blir:

$$L = [1.01 \quad 1.25 \quad 1.56 \quad 1.94]$$

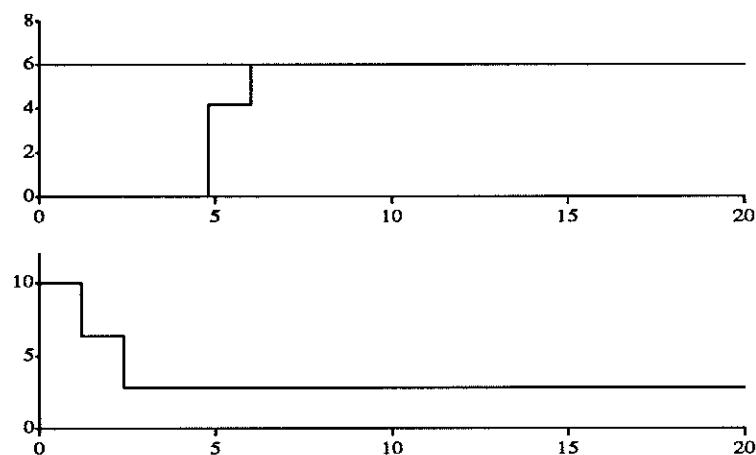
Den statiska förstärkningen blir (utan l_r):

$$H_c(1) = C(I - (\Phi - \Gamma L))^{-1}\Gamma = 0.416$$

Då vi önskar en statisk förstärkning lika med ett, bestäms l_r till:

$$l_r = \frac{1}{0.416} = 2.40$$

Nu har vi tillräcklig information om processen och styrlagen så att vi kan skriva en enkel simnonkod och utföra simuleringar. Programmet för simuleringarna finns i appendix E.



Figur 15: Stegsvår för deadbeat-regulator. Är- och börvärde i den övre grafen, styrsignal i den nedre.

Då reglerfelet i stationaritet skall vara noll, krävs att l_r är exakt beräknad, för att uppnå en statisk förstärkning lika med ett. Detta innebär att den verkliga processen måste vara identisk med modellen. Så är inte fallet, utan för att förhindra stationära fel är den naturliga lösningen att införa en integrator i regulatorn. Ett sätt är att införa ett extra tillstånd som integrerar felet, $r - y$:

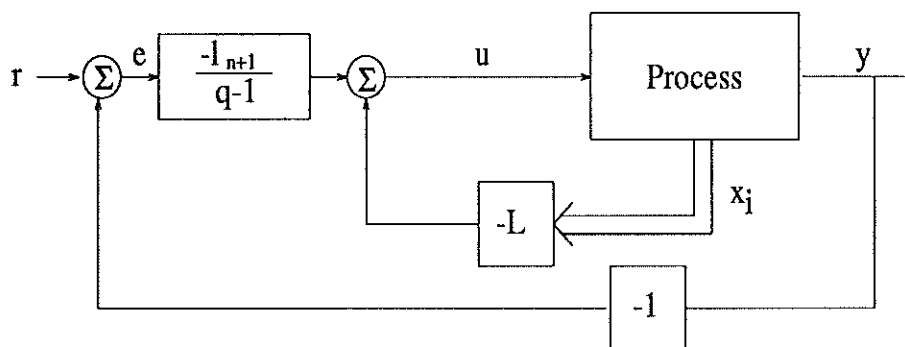
$$x_{n+1}(k+1) = x_{n+1}(k) + r(k) - Cx(k)$$

Systemet på tillståndsform med det extra tillståndet får då följande struktur:

$$\begin{bmatrix} x(k+1) \\ x_{n+1}(k+1) \end{bmatrix} = \begin{bmatrix} \Phi & 0 \\ -C & 1 \end{bmatrix} \begin{bmatrix} x(k) \\ x_{n+1}(k) \end{bmatrix} + \begin{bmatrix} \Gamma \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_c(k)$$

Styrlagen för den nya regulatorn blir:

$$u(k) = -Lx(k) - l_5 x_5(k)$$



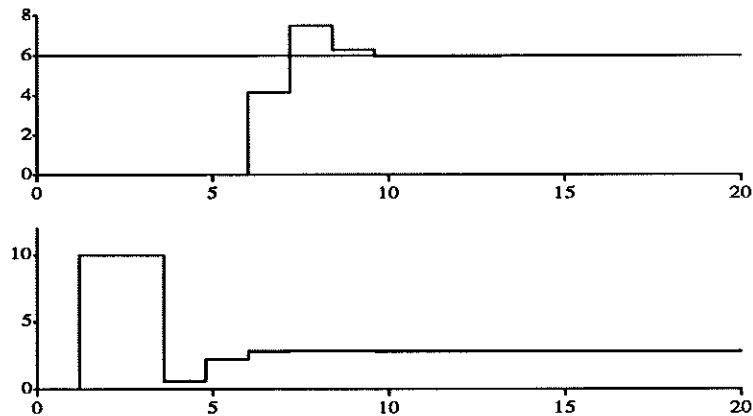
Figur 16: Blockschema över tillståndsåterkoppling med integrator.

Observera att direkttermen l_r saknas. Eftersom börvärdet, r , kommer in i integratorn valde vi att plocka bort l_r .

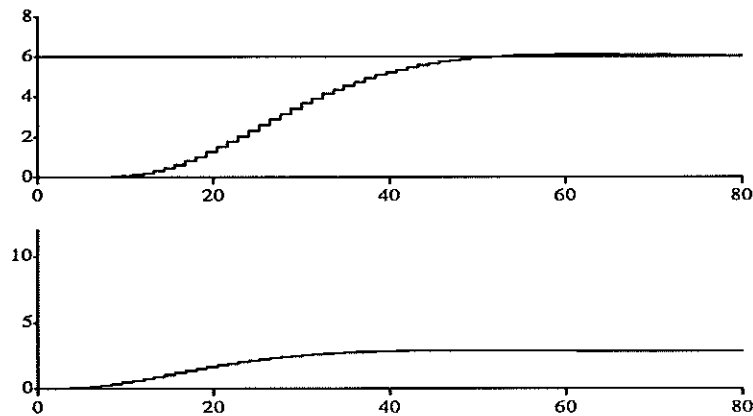
Den med ett tillstånd, l_5 , utökade L-vektorn beräknades på samma sätt som tidigare. Vi valde återigen en deadbeat-struktur (polerna placerade i origo).

$$L = \begin{bmatrix} 8.16 & 7.15 & 5.90 & 4.34 & -2.40 \end{bmatrix}$$

Vi implementerade det 5:e tillståndet i simonkod och utförde nya simuleringar.



Figur 17: Stegsvär för deadbeat-regulator med integrator-del. Är- och börvärde i den övre grafen, styrsignal i den nedre. Observera att styrsignalen bottnar.



Figur 18: Stegsvär för regulator med integrator-del, alla polerna placerade i 0.8. Är- och börvärde i den övre grafen, styrsignal i den nedre.

I figur 17 ser man en översläng på utsignalen, den uppkommer på grund av windup i integrator-delen då styrsignalen bottnar. Detta går dock att råda bot på genom att ändra i uppdateringen av integrator-delen :

$$I := I + l_5(r - y) + k(u - v)$$

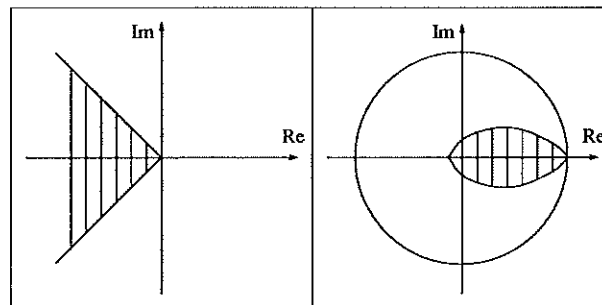
Där u är den begränsade (verkliga) styrsignalen, max 10 V, och v är den teoretiskt beräknade styrsignalen. Konstanten k avgör hur mycket integraldelen skall räknas ned. Värdet på k är något som får bestämmas experimentellt.

Figurerna ovan visar tydligt hur polernas placering påverkar snabbheten på stegs-

varet, ju närmare origo polerna ligger desto snabbare system.

Helst skall regulatoren vara snabb (polerna nära origo). Det finns dock en nackdel med att placera polerna nära origo, eftersom regulatorns känslighet för fel i den skattade modellen ökar väsentligt med poler i närheten av origo. Detta visar sig i att systemet lätt blir svängigt vid deadbeat-reglering om inte modellen perfekt överensstämmer med verkligheten.

Valet av polplacering för kontinuerliga system kan direkt översättas till samplade system. Se figur 19.



Figur 19: Lämplig polplacering för kontinuerliga- (till vänster) och diskreta system.

Vi beräknade L -vektorer för ett antal olika polplaceringar dels helt reella- men även komplexkonjugerade poler. De olika L -vektorerna redovisas i appendix D. För att kunna utvärdera vilken polplacering som ger det bästa resultatet implementerades regulatoren i testriggen.

6 Implementering av den nya regulatorn i testtriggen

Den nuvarande regulatorn i testtriggen är programmerad i s k ladder-kod, se avsnitt 4.4. Det var därför naturligt att implementera den nya regulatorn på samma sätt.

Tyvärr är ladderprogrammering konstruerat för sekvensstyrningar och enkel on/off reglering. Detta medförde att regulatoralgoritmen var omständlig att applicera i styrsystemet och svår att överblicka vid felsökning. Programkoden finns i appendix E. Jämför gärna med den betydligt enklare simnonkoden!

Då provkörningar vid testtriggen påbörjades stötte vi på flera begränsningar i styrsystemet som påverkade våra resultat negativt.

- Styrsystemet kan ej hantera negativa tal.
- Endast heltal mellan 0 – 65535 kan representeras i styrsystemet.
- Kraftiga avrundningsfel p g a heltalsrepresentationen, t ex $2.9 \approx 2$.

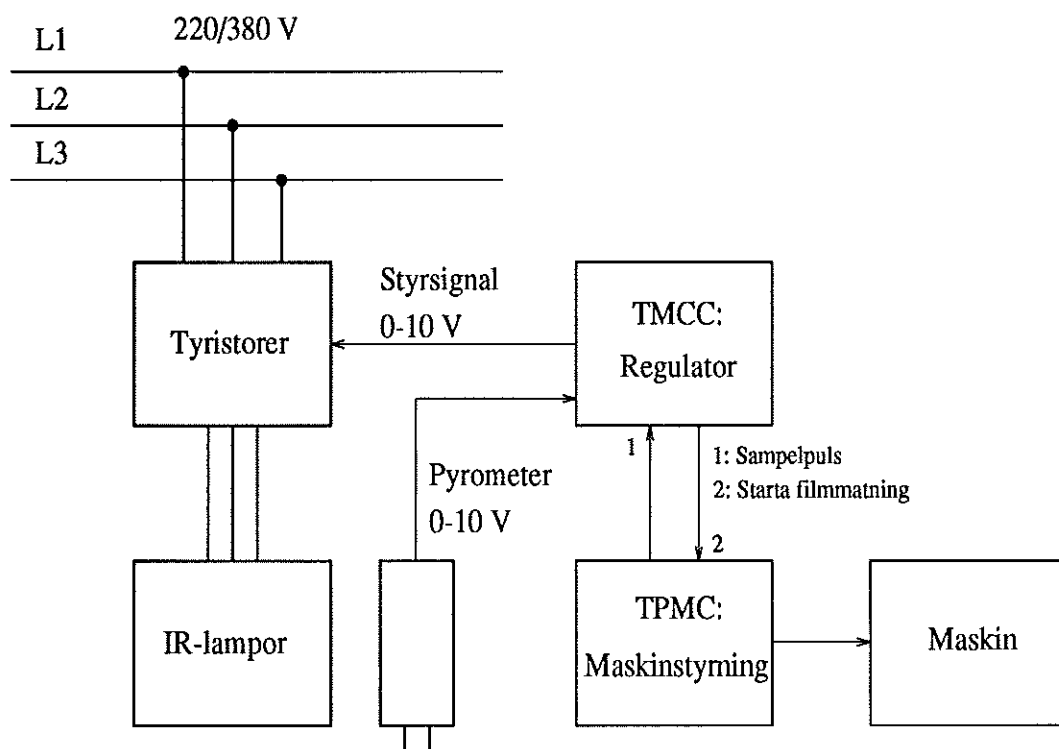
Efter ett par dagars testkörningar fann vi en reglering som inte kunde accepteras, men ej heller förbättras med den befintliga utrustningen. Vårt största problem var de stora avrundningsfelen som uppstod eftersom de förberäknade L-vektorerna innehöll decimaltal. Detta löstes genom att multiplicera L med en faktor 100, vilket fick till följd att vissa beräkningar översteg talområdet (65535). För att undvika detta skalades insignalerna ned med en faktor 10, men regleringen blev inte bra, ty när insignalerna skalades ned gick mycket information förlorad p g a avrundningsfel, t ex $619/10 = 61.9 \approx 61$.

Då vi påtalade ovan nämnda problem för vår handledare på Tetra Pak blev vi upplysta om en ny modul som kan användas tillsammans med TPMC som heter TMCC (Tetra Pak Multi-purpose Compact Controller).

6.1 TMCC-modul

TMCC är en speciell modul som går att använda tillsammans med TPMC. Den är programmerbar i C++ och det är därför lämpligt att implementera den nya regulatorn i TMCC-modulen. Vi kan på så vis undvika de tidigare nämnda räk-nemässiga problemen och får en mer exakt algoritm. Samtidigt fås en bättre förståelse för regulatorns uppbyggnad jämfört med det omständliga ladderpro-grammet.

TMCC är uppbyggt kring en processor från Philips/Sigetics och konstruerad för Tetra Paks standard på digitala signaler (24 V). TMCC innehåller även analoga in- och utgångar (0 – 10 V). Programmeringen sker med en PC och laddas sedan ned i TMCC genom ett seriellt interface.



Figur 20: Blockschema för temperaturreglering med regulatoralgoritmen imple-menterad i TMCC.

Programmet är uppbyggt av ett kort huvudprogram med meny och två funk-

tioner, en tillståndsåterkoppling och en PI-regulator. All avläsning av ingångar sker med hjälp av tidsstyrda avbrottsrutiner, beräkning av nya styrsignaler och uppdatering av tillstånden styrs av ett externt avbrott från TPMC.

Då våra kunskaper i C++ programmering var begränsade fick vi hjälp av Henrik Adolfsson på Tetra Pak med programmeringen. Programmet finns listat i appendix E.

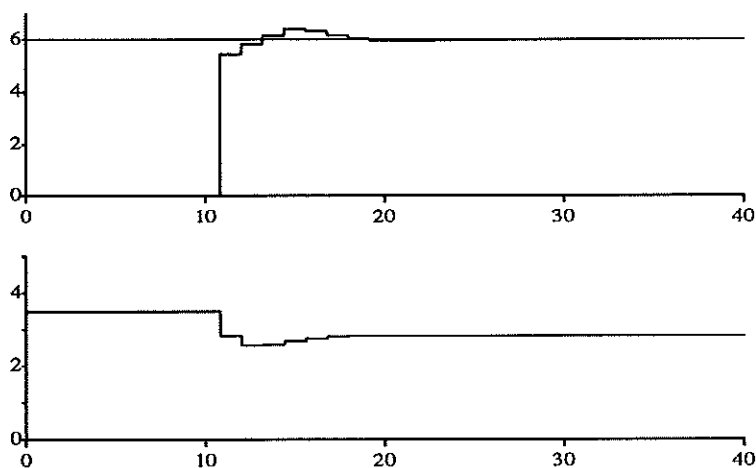
7 Uppstart med förvärmning

För att ytterligare snabba upp startförloppet samt att minska materialspillet infördes en förvärmning av materialet. Detta innebär att uppvärmningen påbörjas en kort tid innan frammatningen startar vilket medför att reglerfelet vid uppstart av maskinen minskar och insvängning sker snabbare.

7.1 Simulering av uppstartsforlopp med förvärmning

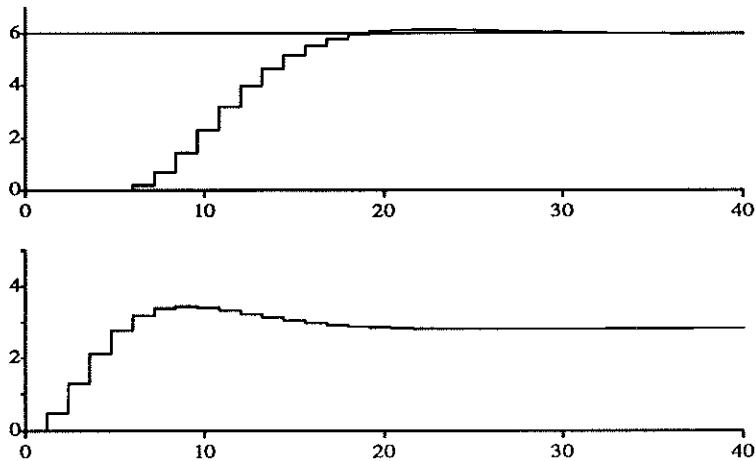
Problemet löstes på så sätt att styrsignalen, u , till lamporna är konstant i 10 s, alla tillstånd uppdateras vid varje sampel och integraldelen initieras till en lämplig konstant. Därefter startar frammatningen och regulatorns beräknade styrsignal läggs ut till lamporna.

Vi simulerade detta i simnon, se appendix E för programkod, och upptäckte att integraldelens initialvärde kraftigt påverkar övergången från konstant till beräknad styrsignal. Ett bra initialvärde på integraldelen är att använda integraldelens stationära värde. I figur 21 visas en simulering med styrsignal 3.5 och integraldelen initierad till 4.5, frammatningen startar efter 10.8 s.



Figur 21: Uppstart med förvärmning under 10 s, poler i 0.5. Är- och börvärde i den övre grafen, styrsignal i den nedre.

Då man jämför med figur 22 syns att rätt temperaturnivå uppnås betydligt snabbare med förvärmning.



Figur 22: Uppstart utan förvärmning, poler i 0.5. Är- och börvärde i den övre grafen, styrsignal i den nedre.

7.2 Implementering av förvärmning i TMCC-modulen

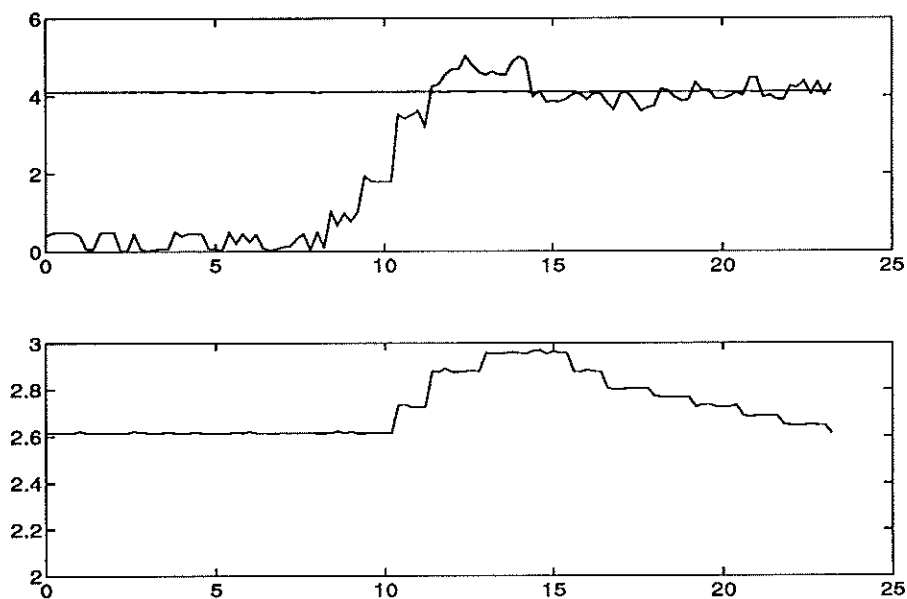
Då förvärmningen skulle införas i C-programmet användes samma tillvägagångssätt som vid den nyss beskrivna simuleringen. Det problem som nu uppstod var att synkronisera maskinstyrningen i TPMC med regleralgoritmen i TMCC, dvs få frammatningen av film att starta vid rätt tidpunkt. Det löstes på så sätt att TMCC skickar en digital signal till TPMC när det är klart att påbörja frammatningen av plastfilm.

De storheter som går att variera vid intrimningen är styrsignalen, u , initialvärdet på integraldelen och tiden som maskinen står stilla. För att få en bra reglering i stationaritet valde vi en långsam regulator, poler placerade i 0.7. Vi beräknade integraldelens storlek i stationaritet och valde detta till initialvärde. Styrsignalen initierades till det värde den har i stationaritet och tiden valdes till 5 sampel. Dessa värden är ej optimala men kan ses som bra startvärden, för att ytterligare förbättra startförloppet måste en mängd tester med olika värden utföras. Till sist fann vi värden som gav ett bra resultat. Dessa var:

- styrsignal, $u = 2.7$ V.

- intergraldelens initialvärde = -13.
- fördröjning av maskinstart = 5 sampel.

Det slutgiltiga uppstartsforloppet registrerades med hjälp av en PC och presenteras i figuren nedan.



Figur 23: Uppstart med förvärmning, poler i 0.7. Är- och börvärde i den övre grafen, styrsignal i den nedre. Frammatning av plast börjar efter 7 s.

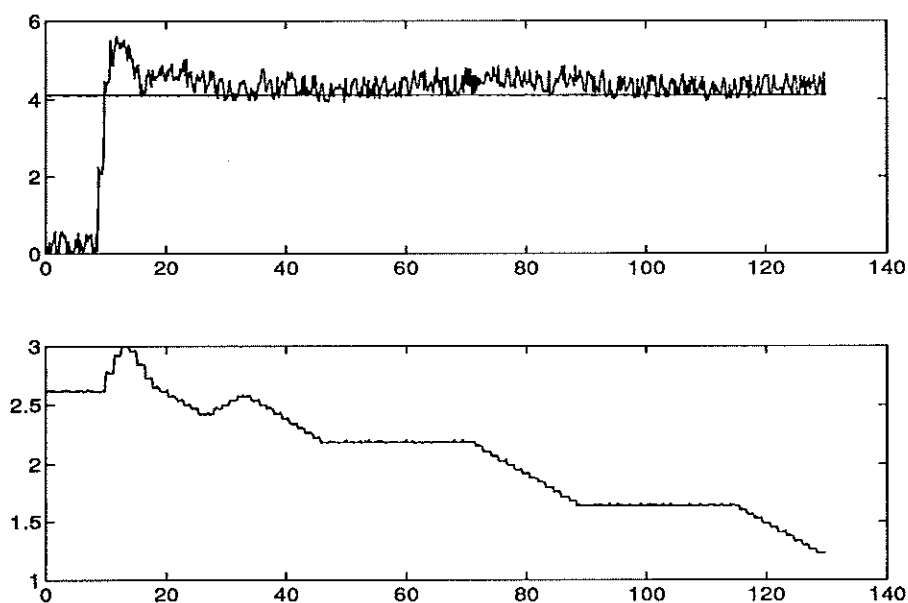
I figuren ovan framgår att styrsignalen är konstant under de 10 första sekunderna. Plasten börjar att matas fram efter 7 s och temperaturen på plasten stiger snabbt och är formbar efter ca 15 s. Det vill säga uppstartstiden blir således 8 s. Detta kan jämföras med den ursprungliga regulatorn vars uppstartstid var ca 30 s.

Som ses i figuren är temperatursignalen något orolig, störningen uppstår p g a att pyrometern ger en strömsignal om 0 – 20 mA vilken måste omvandlas med ett motstånd (500Ω) till en spänningssignal, eftersom TMCC:s analoga ingångar endast accepterar spänningar. Denna extra belastning påverkar pyrometern negativt och spänningen störs av tyristorerna och annan elektrisk utrustning i maskinens omgivning. Det finns även en viss onoggrannhet i pyrometers avläsningar, $\pm 1\%$, vilket motsvarar $\pm 1.5^\circ\text{C}$ i det aktuella temperaturområdet. En lösning på

problemet med spänningsvariationerna kan vara att samla in flera mätpunkter och bilda ett medelvärde av dessa som sedan ligger till grund för beräkning av styrsignalen. Man får på så sätt en mindre variation på spänningen. En annan lösning kan vara att utöka TMCC-modulen med en strömingång.

Då man betraktar temperaturvariationerna i figur 23 får man lätt en felaktig bild av verkligheten. Temperaturen på plasten är inte alls så orolig som det tycks i figur 23. Vid närmare undersökning visar det sig att variationerna är cirka $\pm 5^\circ\text{C}$.

Ett annat problem som uppkom vid längre körningar var att maskindelar och omgivning långsamt värmdes upp. Detta medför att styrsignalen minskar efterhand, se figur 24.



Figur 24: Reglerförlopp med förvärmning, poler i 0.7. Är- och börvärde i den övre grafen, styrsignal i den nedre.

Detta problem uppstår endast i testriggen eftersom i den riktiga maskinen är maskindelarna runt värmeugnen kylda.

Den trappstegsliknande styrsignalen som åskådliggörs i figuren ovan torde ha sin grund i pyrometersignalens upplösning. Man kan konstatera att temperatursignalen måste överskrida ett visst värde för att regulatorn skall reagera.

8 Resultat av den nya regleringen

Den modell av processen vi identifierade stämmer väl överens med den verkliga processen. Vid snabba poler (under 0.5) blir dock regleringen svängig, men vid långsammare poler fungerar regulatorn bra.

Vårt mål med den nya regleringen med förvärmning var att minst halvera uppstartstiden för att på så sätt minska materialspletet. Detta lyckades över förväntan, då vi sänkte uppstartstiden från tidigare dryga 30 s till under 10 s. Att ytterligare sänka denna tid kan vara svårt eftersom den plast som står i ugnen under förvärmningen redan gått till spillo.

En annan önskan var att minska temperaturvariationerna vid stationaritet. Detta lyckades vi ej att förbättra nämnvärt, dels p g a pyrometers onoggrannhet men även för att TMCC-modulen saknade en strömingång. Ytterligare störningsmoment var att testtriggen värmdes upp under drift då maskindelarna saknade kylning.

9 Marknadsundersökning av uppvärmningselement

Eftersom Tetra Pak inte är nöjda med de IR-lampor som används idag, fick vi till uppgift att vid sidan av vårt reglerprojekt undersöka marknaden efter ny värmeutrustning. Problemet med dagens lampor är att de ger en ojämn temperatur över ytan på plasten.

De krav vi ställde på den nya värmaren var:

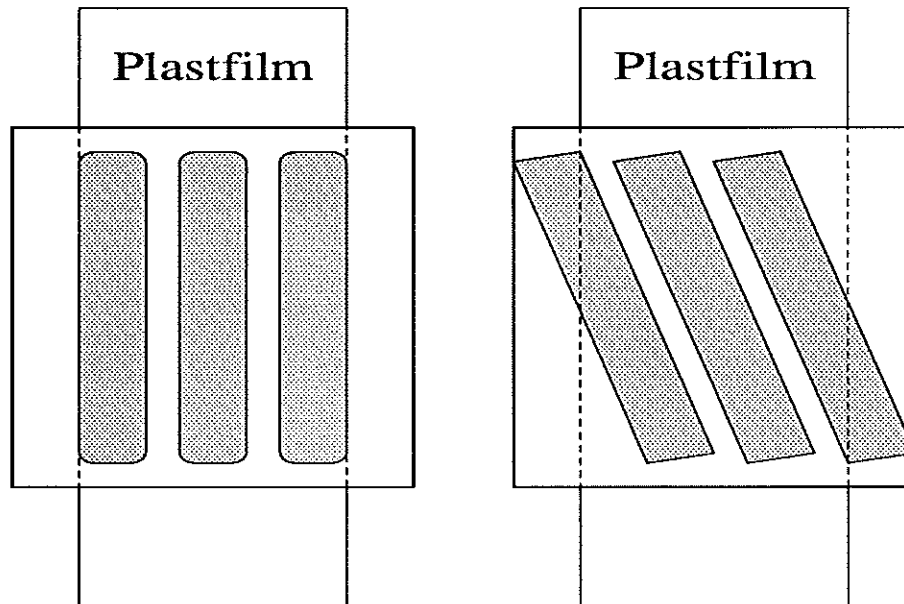
- ge en jämn värmebild på en 110 mm bred plastfilm.
- snabbt kunna uppnå arbetstemperatur och ge snabba svar på ändrad styrsignal.
- max 400 mm lång och 150 mm bred.
- effekt 0.5 – 2 kW.

För att finna lämpliga återförsäljare av värmare sökte vi i databaser på Tetra Pak. Vi skrev ned specifikationen ovan och faxade till ett stort antal företag i Europa och USA. Totalt fick vi iväg ett 100-tal förfrågningar. Svaren droppade in efterhand, men få företag hade produkter som uppfyllde våra krav.

Många företag föreslog keramiska värmare men då dessa har en stor termisk tröghet medför detta att regleringen blir långsam, andra föreslog IR-lampor av quartz-typ. De senare uppvisar liten variation i utförande och prestanda, det som skiljer dem åt är typ av reflektor. Vissa har som i likhet med lamporna som sitter i dagens maskin inbyggd reflektor, medan andra har en separat reflektor med lampan placerad i fokus.

IR-lampornas enda nackdel är den dåliga värmebildningen på plasten, skillnaden mellan en lampa med inre- eller yttre reflektor är ringa. Det skall dock påpekas att med en lampa med yttre reflektor erhålls en lägre temperatur rakt under lampan. Eftersom lampan är placerad i reflektorns fokus (för att skapa ett parallellt strålnippe) får detta till följd att den förhindrar en del av värmestrålningen att nå plasten.

Värmebilden kan dock förbättras genom att placera lamporna enligt figur 25. Detta är tyvärr ej möjligt i dagens maskin då utrymmet där lamporna sitter är konstruerat för en rak placering.



Figur 25: Nuvarande lampplacering till vänster, lampplacering för bättre värmebild till höger.

Genom Mikael Karlsson på Tetra Pak fick vi tag på en artikel i Elektroniktidningen nr 16-93. Denna behandlade s k Rapid Thermal Processing (RTP) vilket används vid tillverkning av kiselskivor. Det går till som så att kiselskivan (diameter ca 10 cm) går in i en ugn där den värms upp m h a ett antal lampor, Tungsten halogen eller Quartz halogen, med en våglängd mellan 0.4 – 1.6 μm . Enligt artikeln kan temperaturen på skivan ökas från rumstemperatur till 1000°C på så kort tid som 10 s. Problemet med regleringen kvarstår dock, eftersom det är svårt att mäta temperaturen så snabbt som krävs i denna applikation. En metod som kanske kan lösa problemet och kan vara intressant att titta mer på är Accufiber. Mer information om detta finns att få av Roland Gullqvist, Pentronic, telefon: 0490 – 23120. Jörgen Bengtsson, Tetra Pak R&D har ett mindre informationshäfte rörande RTP och Accufiber.

Slutligen vill vi påpeka att trots att vi kontaktade många företag så var deras produkter väldigt lika i utförande och prestanda och skiljde sig ytterst lite från de lampor som sitter i dagens maskin. Det som kan vara intressant att titta vidare på för det nya konceptet är RTP-tekniken.

Referenser

- [1] Karl J. Åström, Björn Wittenmark. *Computer controlled systems*. Prentice-Hall inc., 1990.
- [2] Gustaf Olsson, Gianguido Piani. *Computer systems for automation and control*. Prentice-Hall inc., 1992.
- [3] Lennart Ljung. *Moderna analys- och syntesmetoder*. Studentlitteratur, 1981.
- [4] Torsten Söderström. *System identification*. Prentice-Hall inc., 1989.
- [5] John Florian. *Practical thermoforming: Principles and applications*. Dekker, 1987.
- [6] Grahm, Jubrink, Lauber. *Modern Industriell Mätteknik*. Bokförlaget Teknik information, 1990.
- [7] Hilding Elmqvist, Karl Johan Åström, Tomas Schönthal, Björn Wittenmark. *Simnon, user's guide for MS-DOS computers*. SSPA Systems, 1990.
- [8] *PCB83C552, Users manual*. Philips, 1989.
- [9] *TPMC-64/16, Technical description*. Tetra Pak.
- [10] *TMCC User's Manual*. Tetra Pak, 1993.
- [11] Per Foreby. *Att skriva rapport med L^AT_EX*. PH:s kopieringsmaskin, 1992.

A Tekniska data på värmeutrustning

Den värmeutrustning som sitter i testriggen är i stort identisk med den i maskinen. Det som skiljer är fabrikatet på pyrometern. Eftersom alla våra tester är gjorda på testriggen redovisas teknisk data för denna.

A.1 Styrsystem, TPMC

Fabrikat och typ: Satt Control, TPMC-16.

Matningsspänning: 220 V AC.

Antal digitala in- och utgångar: max 256 st.

Analog insignals modul: 4 ingångar, 0 – 10 V eller 0 – 20 mA.

Analog utsignals modul: 4 utgångar, 0 – 10 V eller 0 – 20 mA.

Programminne: 2000 bitinstruktioner.

Scanning tid: 10 ms för 2000 bitinstruktioner.

A.2 TMCC-modul

Fabrikat: Tetra Pak.

Matningsspänning: 24 V DC.

Antal digitala in- och utgångar: max 26 st.

Antal analoga in- och utgångar: 4 st, 0 – 10 V.

Processor: Philips/Signetics, P87C592.

A.3 Pyrometer

Fabrikat och typ: Raytek, IT 5.

Matningsspänning: 9 – 24 V DC eller 12 V AC.

Utsignal: 0 – 20 mA.

Mätområde: 0 – 200°C.

Våglängdsområde: 8 – 14 μm .

Noggrannhet: $\pm 1\%$ av avläst värde.

Svarstid: 400 ms.

A.4 Tyristorer

Fabrikat: Eurotherm.

Märkspänning: 240 V.

Märkström: 15 A.

Styrspänning: 0 – 10 V.

A.5 IR-lampor

Fabrikat och typ: Heraeus, H1741.

Matningsspänning: 220 V.

Effekt: 800 W.

Våglängd: 1 – 2 μm .

B Beräkning av styrbarhetsmatrisen W_c

För att kunna välja polplacering helt fritt krävs att systemet är styrbart. Detta kontrolleras genom att beräkna den s k *styrbarhetsmatrisen*, W_c .

$$W_c = \begin{bmatrix} \Gamma & \Phi\Gamma & \dots & \Phi^{n-1}\Gamma \end{bmatrix}$$

Om W_c har full rang så innebär detta att polplaceringen kan väljas godtyckligt.

I vårt fall har vi:

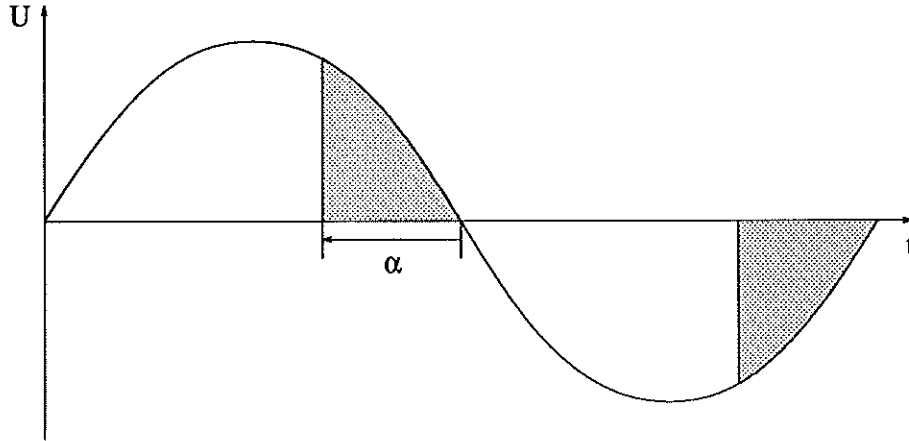
$$\Phi = \begin{bmatrix} 0.8048 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \Gamma = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.4158 \end{bmatrix}$$

det vill säga

$$W_c = \begin{bmatrix} \Gamma & \Phi\Gamma & \Phi^2\Gamma & \Phi^3\Gamma \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0.4158 \\ 0 & 0 & 0.4158 & 0 \\ 0 & 0.4158 & 0 & 0 \\ 0.4158 & 0 & 0 & 0 \end{bmatrix}$$

Som framgår av styrbarhetsmatrisen, W_c , har den full rang. Vi har således inga inskränkningar vad det gäller polplacering.

C Linearitetskontroll av tyristorer



Figur 26: Fasvinkelstyrning av tyristorer

Då IR-lamporna betraktas som rent resistiva innebär detta att uteffekten, P , från lamporna är proportionell mot kvadraten på spänningens effektivvärde, U , d v s:

$$P \sim U^2$$

$$U = \sqrt{\frac{1}{T} \int_0^T u^2(t) dt}$$

där

$$u(t) = \hat{u} \sin \omega t$$

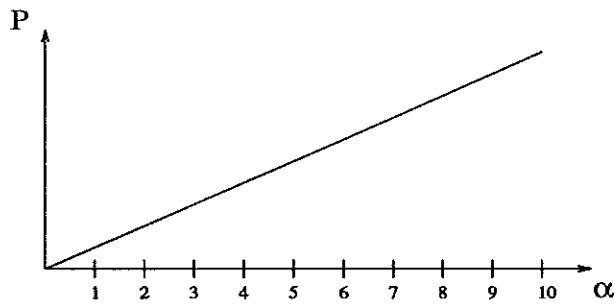
Styrsignalen α kan varieras mellan 0 – 10 V. 10 V innebär normal växelspanning med ett effektivvärde om $\hat{u}/\sqrt{2}$ V och 0 V innebär att tyristorn ej leder.

$$U^2 = \frac{2}{T} \int_{\frac{T}{2}(1-\frac{\alpha}{10})}^{T/2} \hat{u}^2 \sin^2 \frac{2\pi}{T} t dt = \frac{\hat{u}^2}{T} [t - \frac{1}{2} \sin 2t]_{1-\alpha/10}^{T/2} = \frac{\hat{u}^2}{T} (\frac{T\alpha}{20} + \frac{1}{2} \sin T (1 - \frac{\alpha}{10}))$$

Den aktiva effekten från lamporna blir därmed:

$$P(\alpha) \sim \frac{\hat{u}^2}{T} (\frac{T\alpha}{20} + \frac{1}{2} \sin T (1 - \frac{\alpha}{10}))$$

Som framgår ur figur 27 är tyristorena således lineära.



Figur 27: Linearitetskontroll av tyristor.

D Beräkning av L-vektorn vid olika val av polplacering

För att bestämma L-vektorn, $[l_1 \dots l_5]$, användes funktionen *acker* i Matlab.

Vi beräknade L för ett antal olika polplaceringar och redovisar nedan $[l_1 \dots l_5]$.

De fem polerna är placerade tillsammans på reella-axeln, från origo till 0.8.

<i>Polplacering</i>	l_1	l_2	l_3	l_4	l_5
0	8.16	7.15	5.90	4.34	-2.40
0.1	5.13	4.61	3.97	3.14	-1.42
0.2	3.04	2.80	2.52	1.94	-0.788
0.3	1.67	1.56	1.55	0.733	-0.404
0.4	0.824	0.761	1.06	-0.470	-0.187
0.5	0.353	0.251	1.06	-1.67	-0.0752
0.6	0.122	-0.112	1.53	-2.87	-0.0246
0.7	0.0298	-0.472	2.49	-4.08	-0.0058
0.8	0.0039	-0.975	3.93	-5.28	-0.0008

E Programlistningar

E.1 Simulering med tillståndsåterkoppling

E.1.1 Processmodell

DISCRETE SYSTEM syst114

"Tidsdiskret simonbeskrivning över IR-ugnen i testriggen

"Sampeltid 1.2 s.

INPUT u

OUTPUT y

STATE x1 x2 x3 x4

NEW nx1 nx2 nx3 nx4

TIME t

TSAMP ts

$nx1 = 0.8048 * x1 + x2$

$nx2 = x3$

$nx3 = x4$

$nx4 = 0.4158 * u$

$y = x1$

$ts = t + 1.2$

END

E.1.2 Regulator utan integratorverkan

DISCRETE SYSTEM lreg

"Tidsdiskret tillståndsåterkoppling m h a polplacering

"Sampeltid 1.2 s.

INPUT uc y

OUTPUT u

STATE x1 x2 x3 x4

NEW nx1 nx2 nx3 nx4

TIME t

TSAMP ts

SORT

v= $-(l1*x1+l2*x2+l3*x3+l4*x4)+lr*uc$ "Beräknad styrsignal

u= $\text{if } v < 0 \text{ then } 0 \text{ else if } v < 10 \text{ then } v \text{ else } 10$ "Verklig utsignal

ts=t+1.2

nx1= 0.8048*y+x2

nx2=x3

nx3=x4

nx4=0.4158*u

l1:1.0089 "Beräknad L-vektor. Olika värden beroende på

l2:1.2537 "vald polplacering.

l3:1.5577

l4:1.9355

lr:2.4067 "Beräknat lr som ger statiska förstärkningen 1

END

E.1.3 Regulator med integratorverkan

DISCRETE SYSTEM lireg

"Tidsdiskret tillståndsåterkoppling m h a polplacering. Integratorverkan

"införd som ett extra tillstånd. Sampeltid 1.2 s.

INPUT uc y

OUTPUT u

STATE I x1 x2 x3 x4

NEW nI nx1 nx2 nx3 nx4

TIME t

TSAMP ts

SORT

v=((-(l1*x1+l2*x2+l3*x3+l4*x4)-I))

"Beräknad styrsignal

u=if v<0 then 0 else if v<10 then v else 10

"Verklig utsignal

nI= (I+15*(uc-y))

"Integraldel

ts=t+1.2

nx1= 0.8048*y+x2

"Uppdatering av tillstånden

nx2=x3

"x1-x4.

nx3=x4

nx4=0.4158*u

l1:0.35

"Beräknad L-vektor. Olika värden

l2:0.25

"beroende på vald polplacering.

l3:1.06

l4:-1.67

l5: -0.08

END

E.1.4 Ihopkoppling av process och regulator

CONNECTING SYSTEM lconn

"Kopplar ihop processen med regulatorn.

TIME t

uc[lireg]=6

"Börvärdet valt till 6.

y[lireg]=y[syst114]

u[syst114]=u[lireg]

END

E.2 Simulering med förvärmning

E.2.1 Processmodell inklusive förvärmning

DISCRETE SYSTEM syst114

"Tidsdiskret simonbeskrivning över IR-ugnen i testriggen,
"inklusive förvärmning vilket innebär att frammatning av
"plastfilm startar efter 10 s.

INPUT u

OUTPUT y

STATE x1 x2 x3 x4

NEW nx1 nx2 nx3 nx4

TIME t

TSAMP ts

$nx1=0.8048*x1+x2$

$nx2=x3$

$nx3=x4$

$nx4=0.4158*u$

$y=\text{if } t < 10 \text{ then } 0 \text{ else } x1$ "Frammatningen startar efter 10 s.

$ts=t+1.2$

END

E.2.2 Regulator med integralverkan vid förvärmning

DISCRETE SYSTEM lireg

"Tidsdiskret tillståndsåterkoppling m h a polplacering. Integratorverkan
"införd som ett extra tillstånd. Förvärmning av plasten under 10 s.

INPUT uc y

OUTPUT u

STATE I x1 x2 x3 x4

NEW nI nx1 nx2 nx3 nx4

TIME t

TSAMP ts

SORT

v=if t<10 then 3.5 else ((-(l1*x1+l2*x2+l3*x3+l4*x4)-I)) "Beräknad styrsignal

u=if v<0 then 0 else if v<10 then v else 10 "Verklig utsignal

nI=if t<10 then -4.5 else (I+l5*(uc-y)) "Integraldel

ts=t+1.2

nx1= if t<10 then (0.8048*6+x2) else 0.8048*y+x2 "Uppdatering av

nx2=x3 "tillstånd x1-x4.

nx3=x4

nx4=0.4158*u

l1:0.35 "Beräknad L-vektor. Olika värden

l2:0.25 "beroende på vald polplacering.

l3:1.06

l4:-1.67

l5:-0.08

END

E.2.3 Ihopkoppling av process och regulator vid förvärmning

CONNECTING SYSTEM lconn

"Kopplar ihop processen med regulatorn.

TIME tuc[lireg]=6

"Börvärdet valt till 6.

y[lireg]=y[syst114]

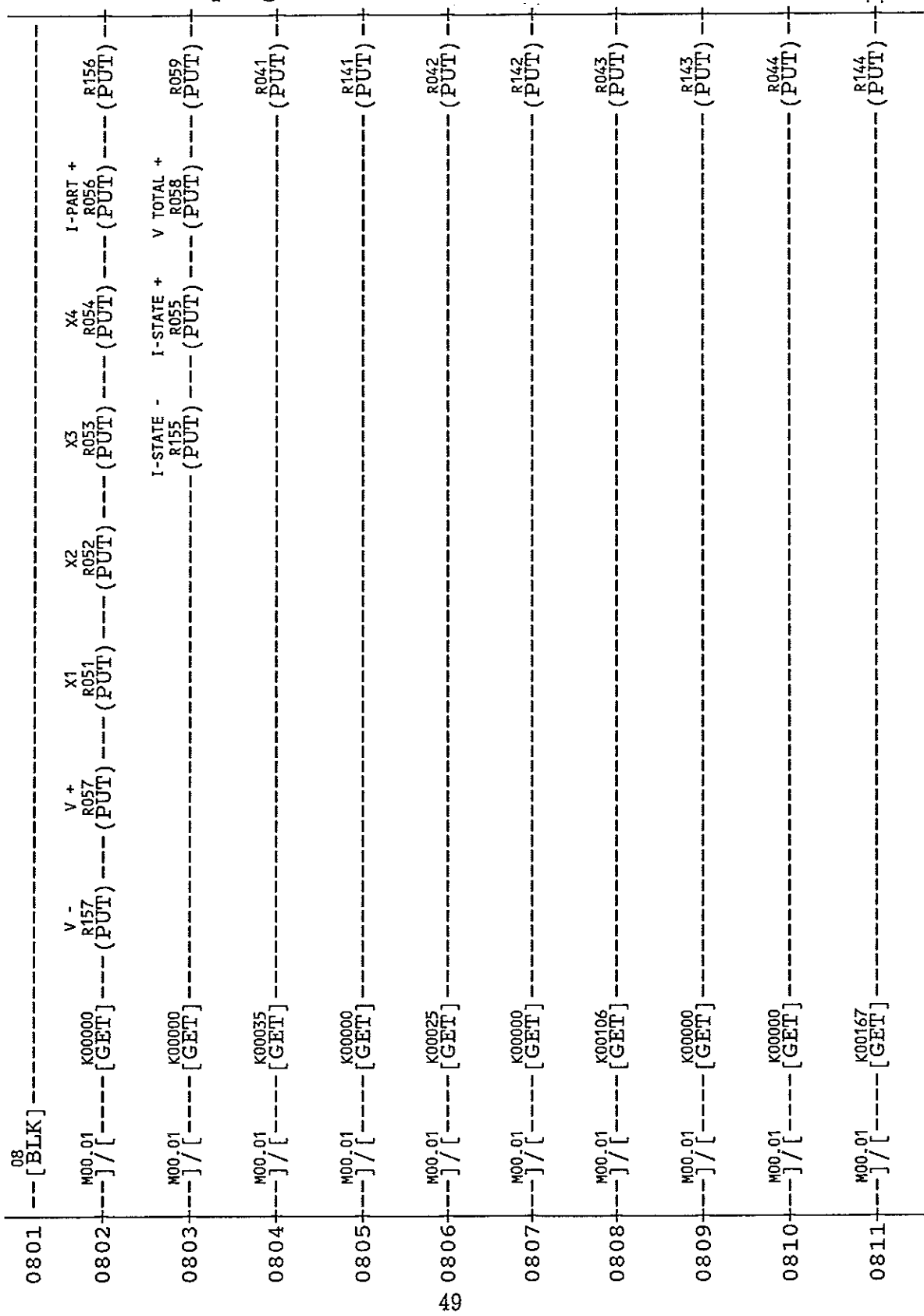
u[syst114]=u[lireg]

END

TPMC-PROGRAMME LIST

TEMPERATUR REGULATOR TILLSTANDS TERKOPPLING

E.3 Ladderprogram för tillståndsåterkoppling



TPMC-PROGRAMME LIST

0812	M00.01] / [K00000 [GET]				R045 (PUT)	15 +
0813	M00.01] / [K00008 [GET]				R145 (PUT)	15 -
0814	CONT. SAMPL M02.00] [PYROMETER R010 [GET]	K00010 [:]			R016 (PUT)	MEASURED VALUE
0815	CONT. SAMPL M02.00] [SET VALUE R012 [GET]	K00010 [:]			R040 (PUT)	SET VALUE
0816	CONT. SAMPL M02.00] [L1 + R041 [GET]	L1 - R141 [>=]	L1 + R041 [GET]	X1 R051 [*]	R061 (PUT)	11*X1 +
				K00000 * [GET]		R161 (PUT)	11*X1 -
0817	CONT. SAMPL M02.00] [L1 + R041 [GET]	L1 - R141 [<]	L1 - R141 [GET]	X1 R051 [*]	R161 (PUT)	11*X1 -
				K00000 * [GET]		R061 (PUT)	11*X1 +
0818	CONT. SAMPL M02.00] [L2 + R042 [GET]	L2 - R142 [>=]	L2 + R042 [GET]	X2 R052 [*]	R062 (PUT)	12*X2 +
				K00000 * [GET]		R162 (PUT)	12*X2 -
0819	CONT. SAMPL M02.00] [L2 + R042 [GET]	L2 - R142 [<]	L2 - R142 [GET]	X2 R052 [*]	R162 (PUT)	12*X2 -
				K00000 * [GET]		R062 (PUT)	12*X2 +

TPMC-PROGRAMME LIST

0820	CONT. SAMPL M02.00	L3 + R043	L3 - R143	L3 + R043	X3 R053	R063 (PUT)	13*X3 +
		[GET]	[>=]	*-[GET]	*-[*]		
				K00000		R163 (PUT)	13*X3 -
				*-[GET]			
0821	CONT. SAMPL M02.00	L3 + R043	L3 - R143	L3 - R143	X3 R053	R163 (PUT)	13*X3 -
		[GET]	[<]	*-[GET]	*-[*]		
				K00000		R063 (PUT)	13*X3 +
				*-[GET]			
0822	CONT. SAMPL M02.00	L4 + R044	L4 - R144	L4 + R044	X4 R054	R064 (PUT)	14*X4 +
		[GET]	[>=]	*-[GET]	*-[*]		
				K00000		R164 (PUT)	14*X4 -
				*-[GET]			
0823	CONT. SAMPL M02.00	L4 + R044	L4 - R144	L4 - R144	X4 R054	R164 (PUT)	14*X4 -
		[GET]	[<]	*-[GET]	*-[*]		
				K00000		R064 (PUT)	14*X4 +
				*-[GET]			
0824	CONT. SAMPL M02.00	L1*X1 + R061	L2*X2 + R062	L3*X3 + R063	L4*X4 + R064	R070 (PUT)	L*X +
		[GET]	[+]	[+]	[+]		
0825	CONT. SAMPL M02.00	L1*X1 - R161	L2*X2 - R162	L3*X3 - R163	L4*X4 - R164	R170 (PUT)	L*X -
		[GET]	[+]	[+]	[+]		
0826	CONT. SAMPL M02.00	L*X + R070	L*X - R170	L*X - R170	L*X + R070	R180 (PUT)	TOT. L*X -
		[GET]	[<]	*-[GET]	*-[*]		
				K00000		R080 (PUT)	TOT. L*X +
				*-[GET]			

TPMC-PROGRAMME LIST

0834	CONT. SAMPL M02.00	V + R057 [GET]	V - R157 [>=]	V + R057 [GET]	V - R157 [-]	R058 (PUT)	TOT. COMPUTED OUTPUT +
				* K00000 * [GET]		R158 (PUT)	TOT. COMPUTED OUTPUT -
0835	CONT. SAMPL M02.00	V + R057 [GET]	V - R157 [<]	V - R157 [GET]	V + R057 [-]	R158 (PUT)	TOT. COMPUTED OUTPUT -
				* K00000 * [GET]		R058 (PUT)	TOT. COMPUTED OUTPUT +
0836	CONT. SAMPL M02.00	V TOTAL + R058 [GET]	K00010 [:]			R058 (PUT)	TOT. COMPUTED OUTPUT +
0837	CONT. SAMPL M02.00	V TOTAL - R158 [GET]	K00010 [:]			R158 (PUT)	TOT. COMPUTED OUTPUT -
0838	CONT. SAMPL M02.00	V TOTAL + R058 [GET]				R059 (PUT)	LIMITED OUTPUT (0-255)
0839	M00.01	U + R059 [GET]	K00255 [>]	K00255 [GET]		R059 (PUT)	LIMITED OUTPUT (0-255)
0840	M00.01	REG CHOICE R090 [GET]	K00001 [=]	K32768 [GET]	U + R059 [+]	007 (PUT)	ANALOG UT
0841	CONT. SAMPL M02.00	U + R059 [GET]	V TOTAL + R058 [=]	K00000 [GET]	K*(U-V) R060 (PUT)	R160 (PUT)	ANTI WINDUP -
0842	CONT. SAMPL M02.00	U + R059 [GET]	V TOTAL + R058 [<]	V TOTAL + R058 [GET]	U + R059 [-]	R160 (PUT)	ANTI WINDUP -
				* K00002 * [*]	K00001 [:]	R060 (PUT)	ANTI WINDUP +

E.5 C-program med förvärmning och tillståndsåterkoppling alternativt PI-regulator utan förvärmning.

```
/******  
;*8051 C code program (C) / \  
;* IAR Compiler ver 4.02 TETRA  
;* Program: (Label on IC) / PAK \  
;* : IRLAMP. testrig /-----\  
;* Project: See above, program for TMCC  
;* Project description:  
;* Associated documents:  
;* Author : Orig 940127 Henrik Adolfson, Rikard Hesslevik, Pehr Hjalmarson  
;* Revisions  
;* Date Ver Sign Comments  
;* 940100 01 HAN First Tetra version  
;*****/  
#include <stdio.h>  
#include <io592.h>  
/*** defines ***/  
#define ON 1  
#define OFF 0  
#define TRUE 1  
#define FALSE 0  
#define FORWARD 1  
#define BACKWARD 0  
#define N 0 /* parity; None, Odd, Even */  
#define O 1  
#define E 2  
/* The definitions below act as variables in the external RAM area on the  
specified 16-bit low order addresses. */  
#define TMCCHighOut (*(char*) 0x01E400) /* pin 25-32 */  
#define TMCCLowOut (*(char*) 0x01E800) /* pin 17-24 */  
#define TMCCLowIn (*(char*) 0x01EC00) /* pin 9-16 */
```

```

#define TMCCHighIn    (*(char*) 0x01F000)    /* pin 17-24 */
#define TMCCSw       (*(char*) 0x01F400)
#define TMCCLEDs     (*(char*) 0x01F800)
#define TMCCA0Out    (*(char*) 0x01FC00)    /* pin 5 */
#define TMCCA1Out    (*(char*) 0x01FC01)    /* pin 6 */
/* program area and parameters storage */
#define InternalRAM   ((char*) 0x012000)
/* profile parameter storage */
#define EEPROM        ((char*) 0x010800)
#define Startpulse   CTOI
/**/ declarations ***/
unsigned char  ch, Timebyte1, NoBytes, AddrH, AddrL, Type, RAMA0Out,
               ChkSum, ib, ib2, Temp, Pulsecount, RAMLEDsOut, cnt,
               Progbyte, NoChars , First, z, z1;
unsigned int   Address, TempInt, i, j, k,m, c, iw, iw2, Timedelay;

long int      u, y, uc;
xdata float   x1, x2, x3, x4, I, v, e;
xdata float   ipart,c1, c2, p;
bit           Pulse;
static xdata unsigned char ComLine[80], Ramline[20];
static xdata float MulPli = 0.25;
/**/ External Interrupt 0 ***/
interrupt[ 0x03 ] void EX0_int( void )
{
    if(TMCCSw & 0x01) {
        Statefeedback();
    }
    else
        if(TMCCSw & 0x02) {
           pireg();
        }
}

```

```

/** Timer0 interrupt */
interrupt [ 0x0B ] void T0_int( void )
{
    TH0 = -6;      /* start Timer0 for a new count */
    if( (TMCCSw & 0x0F) ) { PCON |= 0x010; T3 = 0; } /* WatchDog */
    if( Timebyte1 ) Timebyte1--;
/** prints u to analouge output */
    RAMA0Out = (char)u;
    TMCCA0Out = RAMA0Out;
/** convert and read "y" */
    ADCON=0x00;
    ADCON=0x08; /* TMCC pin 3 */
    while(!(ADCON & 0x10));
    z=ADCH;
    y=ADCON/64 + (int)z<<2;
/** convert and read "uc" */
    ADCON=0x00;
    ADCON=0x09; /* TMCC pin 4 */
    while(!(ADCON & 0x10));
    z1=ADCH;
    uc=ADCON/64 + (int)z1<<2;
}
/** Timer1 interrupt */
interrupt [ 0x1B ] void T1_int( void ) /* T1 busy for serial
                                         communication tasks */
{
}
/** Init Hardware */
void InitHardware( void )
{
    PCON |= 0x010; T3 = 0; /* WatchDog:sets conditions flag and max.interval */
    TMCCLowOut = TMCCHighOut = ~0; /* sets all bits to 0 */
    TMCCLEDs = 0x044;
}

```



```

/* Configure Timer0 as a 16-bit timer,
   Timer1 as an autoloader timer for the serial port */
TMOD = 0x21;
/* Use Timer0 as system timer with interrupts every 1.024th ms */
TH0 = -6;          /* = 14.7456MHz */
TR0 = ON;          /* start Timer0 */
ET0 = ON; PTO = ON; /* enable high prior. interrupt for Timer0 */
EA = ON;           /* enable all interrupts */
RT2 = OFF;  RT2 = ON;
/**/ Initiate External Interrupt 0 /**/
ITO = ON;
EX0 = ON;/**/ Timer0 interrupt /**/
interrupt [ 0x0B ] void TO_int( void )
{
  TH0 = -6;        /* start Timer0 for a new count */
  if( (TMCCSw & 0x0F) ) { PCON |= 0x010; T3 = 0; } /* WatchDog */
  if( Timebyte1 ) Timebyte1--;
/**/ prints u to analogue output /**/
  RAMA0Out = (char)u;
  TMCCA0Out = RAMA0Out;
/**/ convert and read "y" /**/
  ADCON=0x00;
  ADCON=0x08; /* TMCC pin 3 */
  while(!(ADCON & 0x10));
  z=ADCH;
  y=ADCON/64 + (int)z<<2;
/**/ convert and read "uc" /**/
  ADCON=0x00;
  ADCON=0x09; /* TMCC pin 4 */
  while(!(ADCON & 0x10));
  z1=ADCH;
  uc=ADCON/64 + (int)z1<<2;
}

```

```

/** Timer1 interrupt **/
interrupt [ 0x1B ] void T1_int( void )      /* T1 busy for serial
                                             communication tasks */

{
}

void InitSerialPort( int Baudrate, int Parity, int NoBits,
                    int NoStopbits, int Handshake )

{
/* Only valid setting at this moment is 9600,N,8,1,0 */
  SOCON = 0x052;
  TL1 = TH1 = -8; PCON |= 0x080;
  TR1 = ON; ET1 = OFF;
}

char GetCh( void )
{
  while( !RI );    /* '!RI' wait for Receive Interrupt flag */
  RI = OFF;
  return( SOBUF );
}

void PutCh( char c )
{
  while( !TI );    /* '!TI' wait for Transmit Interrupt flag */
  TI = OFF;
  SOBUF = c;
}

void Send( char *s )
{
  char c;
  while( c = *s++ ) {
    if( c == '\n' ) PutCh( '\r' );
    PutCh( c );
  }
}

```

```

unsigned char GetHex( char *s )
/* This routine presumes that the input is correct i.e. hex numbers only */
{
    unsigned char  c1, c2;
    c1 = *s++ - '0';
    if( c1 <= 9 ) c2 = c1 << 4;
        else c2 = ( c1 - ('A'-'0') + 10 ) << 4;
    c1 = *s - '0';
    if( c1 <= 9 ) return( c2 + c1 );
        else return( c2 + ( c1 - ('A'-'0') + 10 ) );
}

void PutHex( unsigned char c )
{
    unsigned char  c1, c2;
    c1 = (c >> 4);
    if( c1 <= 9 ) c1 += '0';
    else c1 += ('A'-10);
    PutCh( c1 );
    c1 = (c & 0x00F);
    if( c1 <= 9 ) c1 += '0';
    else c1 += ('A'-10);
    PutCh( c1 );
}

void PutDecimal( unsigned int i1, unsigned char c1 )
{
    unsigned char b1, line[8];

    for( b1 = 0; b1 < 7; b1++ ) line[b1] = ' ';
    line[b1] = 0;
    do {
        line[--b1] = i1 % 10 + '0';
        i1 /= 10;
    } while( i1 );
}

```

```

    Send( line+7-c1 );
}
unsigned char GetCommand( void )
{
    unsigned char  i, Index = 0, Inchar;
    for( i=0; i<sizeof(ComLine); i++) ComLine[i] = ' ';
    while( TRUE )
    {
        Inchar = GetCh();
        if( Inchar == '\r' ) return( Index );
        if( Inchar == '\b' ) {
            if( Index == 0 ) Inchar = 0x07; else Index--;
            PutCh( Inchar );
        }
        else
            if( Inchar >= ' ' ) {
                if( 'a'<= Inchar && Inchar <= 'z' ) Inchar -= ('a'-'A');
                if( Index == (sizeof(ComLine)-1) ) Inchar = 0x07;
                else ComLine[ ++Index ] = Inchar;
                PutCh( Inchar );
            }
    }
}
}
/***** Main *****/
main()
{
    InitHardware();
    InitSerialPort( 9600,N,8,1,0 );
    while( TRUE ) {
        if(!RI) {
            Send( "\nIRRIG->" );
            if( NoChars = GetCommand() ) {
                /*****

```

```

/* Display HELP menu */
        if( ComLine[1] == 'H' ) {
Send( "\n\n***** IR-RIG  C o n t r o l  P r o g r a m  1.0  *****" );
Send( "\n\n*****                H e l p  M e n u                *****" );
/* #include <\tmcc\time.c>*/
Send( "\n1 = Prints 'u' to screen" );
Send( "\n2 = Prints 'y' to screen" );
Send( "\n3 = Prints 'uc' to screen" );
Send( "\n4 = Prints 'v' to screen" );
Send( "\n" );

                continue;
        }

/*=====*/
        if( ComLine[1] == '1' ) {
                Send("\nu:");
                PutDecimal( u,6 );
                continue;
        }

/*=====*/
        if( ComLine[1] == '2' ) {
                Send("\ny:");
                PutDecimal( y,6 );
                continue;
        }

/*=====*/
        if( ComLine[1] == '3' ) {
                Send("\nuc:");
                PutDecimal( uc,6 );
                continue;
        }

/*=====*/
        if( ComLine[1] == '4' ) {
                Send("\nv:");

```



```

        u = 0;
    else
        if (v > 255)
            u = 255;
        else
            u = v;
    if (m < 1 )
        y = uc;
    m = m + 1;
    x1 = (0.8048 * y ) * MulPli + x2;  /* update state x1 */
    x2 = x3;                          /* update state x2 */
    x3 = x4;                          /* update state x3 */
    x4 = 0.4158 * u;                  /* update state x4 */
    I = I + 15 * (uc - y ) * MulPli;  /* Integralpart */
}
}
/*=====*/
/* PI-reg */
pireg()
{
    float K=0.33;    /* Gainfactor */
    float Ti=4.5;   /* Integraltime */
    float h=1.2;    /* Sampletime */
    TMCCHighOut = FALSE;
/* Calculation of parameters */
    c1 = K * h / Ti;
    c2 = h / Ti;
/* control algorithm */
    e = (uc - y) * MulPli;  /* calculate control error */
    p = K * e;              /* calculate proportional part */
    v = p + ipart;         /* calculate output */
    if (v < 0)             /* limit control output */
        u = 0;
}

```

```
else
    if (v > 255)
        u = 255;
    else
        u = v;
ipart = ipart + c1 * e; /*Integral part */
}
```