

ISSN 0280-5316
ISRN LUTFD2/TFRT--5627--SE

Produkttestbänk baserad på realtidssimulering

Simon Ungerbrandt

Department of Automatic Control
Lund Institute of Technology
September 1999

Department of Automatic Control Lund Institute of Technology Box 118 SE-221 00 Lund Sweden	<i>Document name</i> MASTER THESIS	
	<i>Date of issue</i> September 1999	
	<i>Document Number</i> ISRN LUTFD2/TFRT--5627--SE	
<i>Author(s)</i> Simon Ungerbrandt	<i>Supervisor</i> Henrik Nilsson, Henrik Olsson, TAC AB Tore Häggglund	
	<i>Sponsoring organisation</i>	
<i>Title and subtitle</i> Produkttestbänk baserad på realtidssimulering (Product test bench based on real-time simulation)		
<i>Abstract</i> <p>The aim of this master thesis was to develop a test bench for a series of zone controllers manufactured by TAC AB in Sweden. For this purpose a simulation program called Simnon was used. Shortly, you write your differential equations as models in Simnon and then you let the program simulate the variation of the different variables during time. A main feature of the program is its capability to simulate in real time. This feature was used in the thesis since the test bench simulates the environment the controller works in, for example the zone (room), the radiators, the outdoor climate, the windows, the walls and so on.</p> <p>To simplify for the user, who would not have to learn Simnon, a client-server application was developed, where Simnon is the server and the client is a graphical interface. To simplify even further, so called standard cases were written. In this way the user would only have to start Simnon and the user interface and the start one of the standard cases. The user is also free to choose which variables to plot during simulation and which to save for later use.</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> English	<i>Number of pages</i> 80	<i>Recipient's notes</i>
<i>Security classification</i>		

The report may be ordered from the Department of Automatic Control or borrowed through:
University Library 2, Box 3, SE-221 00 Lund, Sweden
Fax +46 46 222 44 22 E-mail ub2@ub2.lu.se



LUND INSTITUTE OF TECHNOLOGY
Lund University



Testbänk för Xenta100- serien baserad på realtidssimulering

Examensarbete av Simon Ungerbrandt



LUND INSTITUTE OF TECHNOLOGY

Lund University

Testbänk för Xenta100-serien baserad på realtidssimulering

ABSTRACT:

The aim of this master thesis was to develop a testbench for a series of zonecontrollers manufactured by TAC AB in Sweden. For this purpose a simulation program called Simnon was used. Shortly, you write your differential equations as models in Simnon and then you let the program simulate the variation of the different variables during time. A main feature of the program is its capability to simulate in real-time. This feature was used in this thesis since the testbench simulates the environment the controller works in, for example the zone (room), the radiators, the outdoor climate, the windows, the walls and so on.

To simplify for the user, who would not have to learn Simnon, a client-server application was developed, where Simnon is the server and the client is a graphical interface. To simplify even further, so called standard cases were written. This way the user would only have to start Simnon and the user interface and then start one of these standard cases. The user is also free to choose which variables to plot during simulation and which to save for later use.



LUND INSTITUTE OF TECHNOLOGY

Lund University

Testbänk för Xenta100-serien baserad på realtidssimulering

Table of Contents

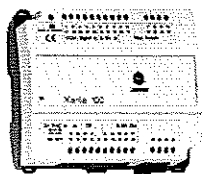
1	Introduktion	8
2	Xenta 102 ES: beskrivning och funktionalitet	9
2.1	TAC XENTA 100 SERIEN	9
2.2	X 102 ES	9
2.2.1	Grundfunktioner	10
2.2.2	Övriga möjligheter	10
2.2.3	Applikationer.....	11
3	Ställdon.....	12
3.1	Thermoställdon	12
3.2	Öka/Minska ställdon.....	12
4	Test	13
5	Testbänken.....	14
5.1	Generellt.....	14
5.2	Hårdvara.....	14
5.2.1	Lite om bryggorna	14
5.3	Mjukvara	16
5.3.1	Microsoft Visual C++ 5.0.....	16
5.4	Produkten X102 ES.....	16
6	SIMNON	17
6.1	Allmänt.....	17
6.2	SADK (Simnon AddOn Developers Kit)	17
6.3	SimRPC.....	17
7	Fysikaliska modeller.....	19
7.1	Fysikalisk modell av ett rum.....	19
7.2	Fysikalisk modell av utetemperaturen	20
7.2.1	Dagligt temperaturmedelvärde	21
7.2.2	Ögonblickligt temperaturvärde.....	21
7.3	Fysikalisk modell av en radiator.....	23
7.3.1	Grashof-talet.....	24
7.3.2	Prantl-talet	24
7.3.3	Nusselt-talet.....	25
7.3.4	Summering av värmeflöden.....	26
7.4	Fysikalisk modell av koldioxidkoncentrationen	27
7.5	Fysikalisk modell av thermoställdon	28
8	Det simulerade systemet.....	31
8.1	Beskrivning	31
8.2	Modellen och de ingående filerna	32
8.2.1	Förenklingar	32
8.3	Beskrivning av SADK-funktionerna	32
9	X102ESTest	34
9.1	SimulationSetup	34
9.2	RunTimeSettings	34
9.3	RoomSettings	35

2 Xenta 102 ES: beskrivning och funktionalitet

t.a.c. utvecklar, tillverkar och marknadsför produkter och system som förbättrar inomhusklimat och energiekonomi. Vid utvecklingsavdelning i Malmö utvecklas regulatorer och datorbaserade system för styrning, reglering och övervakning av värme- och ventilationsanläggningar.

2.1 TAC XENTA 100 SERIEN

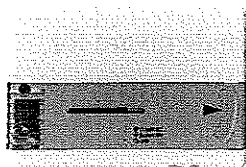
TAC Xenta 100 är en familj av applikationsspecifika, förprogrammerade zonregulatorer som reglerar inomhusklimatet i enstaka rum (zoner). Regulatorn finns i ett antal olika modeller för bl.a. fläktkonvektor, VAV och kyltak. Produkterna kan användas både som självständiga enheter och som del i ett större nätverk.



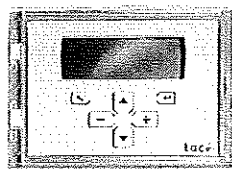
Figur 2.1: Regulator ur X100-serien

TAC Xenta 100 kan integreras i ett nätverk genom LonWorks kommunikation. De kan också kopplas ihop med andra system i byggnader, såsom belysning- och alarmsystem baserade på Lonworks teknologi för att erhålla ett kostnadseffektivt system.

Regulatorernas olika variabelinställningar och avläsningar kan göras både via nätverket, om regulatorn är ansluten till ett sådant, men också direkt med hjälp av en operatörspanel (OP), som kopplas in till rumsenheten. Rumsenheten är direkt ansluten till regulatorn och mäter temperaturen i zonen. På denna rumsenhet finns även en börvärdesratt och en bypassknapp som tillåter användaren att göra inställningar.



Figur 2.2: Rumsenhet till regulatorn ovan...



Figur 2.3: ...och operatörspanelen

2.2 X 102 ES

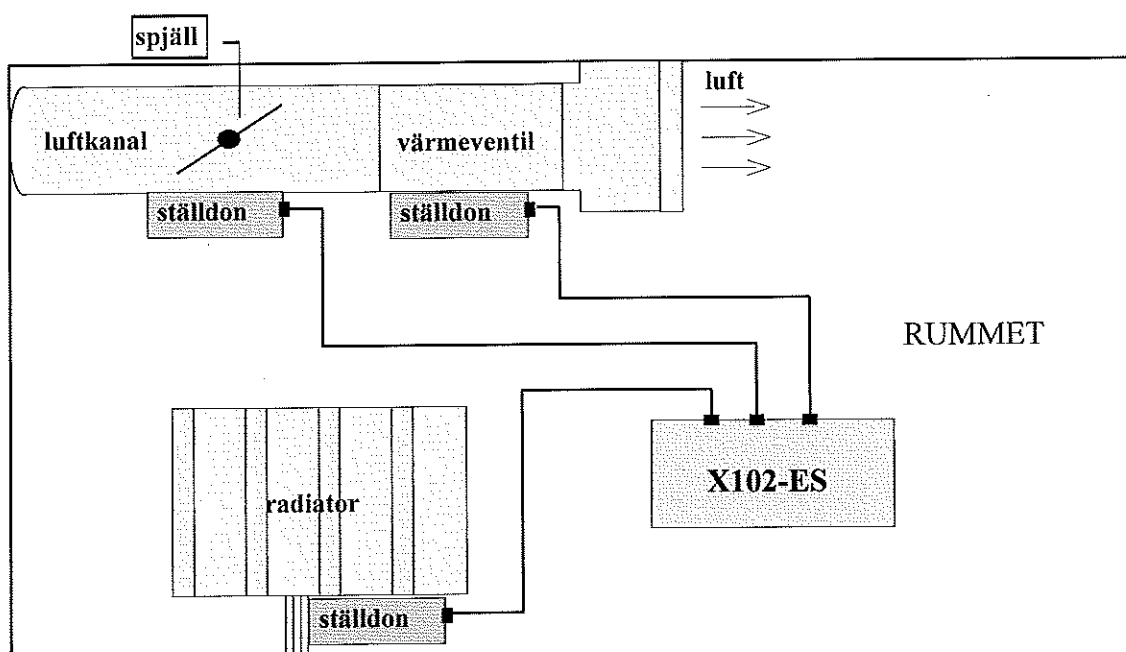
Xenta 102 ES är en ny produkt anpassad för en VAV-process (Variable Air Volume). Det innebär att den i första hand reglerar zontemperaturen genom att variera storleken på ett luftflöde av konstant temperatur som strömmar in till zonen. Med hjälp av en inbyggd luftflödesgivare mäter regulatorn

- koldioxidgivare: när den är ansluten kommer regulatort att hålla koldioxidkoncentrationen i zonen mellan inställda gränser.
- extra temperaturgivare: ämnad till att ersätta temperaturgivare i rumsenheten.

2.2.3 Applikationer

Regulatorn är avsedd att användas i följande typ av applikationer:

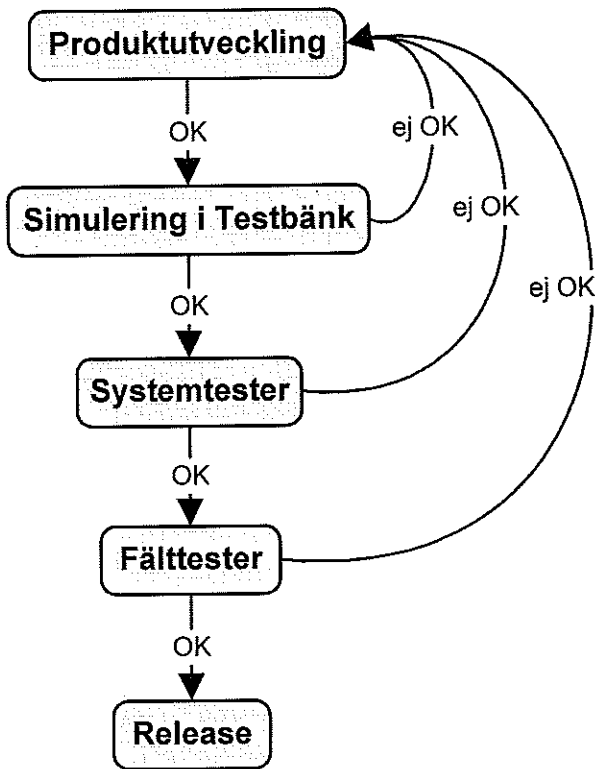
- **Applikationer med endast spjäll (endast kyla):** regulatort försöker hålla zontemperaturen vid ett kylbörvärde genom att endast reglera luftflödet in i rummet. Detta luftflöde har ett minsta och högsta värde, beroende på typ och storlek på luftkanalen in i rummet. Minsta värdet är till för att upprätthålla ventilation i rummet eller ett visst börvärde på koldioxidhalten, det högsta värdet bestäms av gränsen för att trumman ska börja låta och det kallras som kan uppkomma i rummet på grund av de luftrörelser som uppstår.
- **Applikationer med spjäll och ett värmesteg:** i detta fall upprätthålls temperaturen vid börvärdet inte endast genom kylning, utan även genom uppvärmning av luften i zonen. För att öka genomströmningen av luft genom värmesteget kan en fläkt kopplas in. Fläkten kan kopplas antingen i serie eller parallellt med värmesteget.
- **Applikationer med spjäll och två värmesteg:** Som ovan, men i detta fall finns två värmesteg inkopplade, till exempel ett element och en värmeventil.



Figur 2.4: Skiss över en typisk inkoppling i ett rum

4 Test

Den färdiga testbanken kan användas som ett led i utvecklingsfasen, exempelvis som i nedan presenterade flödesdiagram.



Produktutveckling: hårdvara och mjukvara till den nya regulatorn tas fram och prototypen byggs.

Testbank anpassas efter produktkrav, eventuellt skrivs nya modeller.

Simulering i testbank: den nya programvaran testas. Om beteendet inte är som förväntat, gå tillbaka till produktutveckling. Om beteendet är som väntat, gå vidare till systemtester.

Systemtester testas till exempel hur datakommunikation mellan regulator och systemet den är inkopplad på fungerar. Om beteendet inte är tillfredsställande, gå tillbaka till produktutveckling, annars gå vidare till fälttester.

Fälttesterna är de mest omfattande och antagligen tidskrävande. Visar sig något inte fungera, gå tillbaka till produktutveckling, annars kan en release göras.

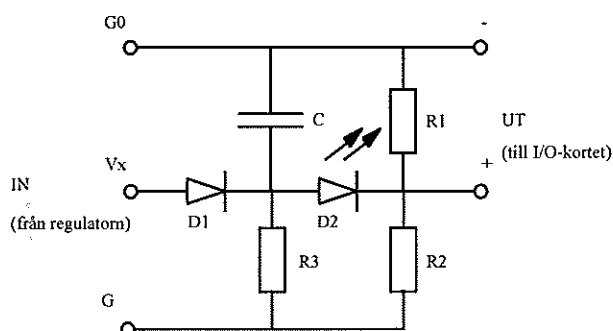
Testbanken kan användas också till att testa någon enstaka regulator som kommit på retur från någon kund för att se vad som inte uppfyller de uppställda kraven. Då de olika inställningarna för rumsstorlek, radiatorstorlek, värmelast, mm har gjorts för att efterlikna kundens miljö, kan man se på loggen vad som betar sig konstigt och åtgärda det.

De två bryggorna som används är en diod- och en reläbrygga. Diodbryggan är till för de analoga signalerna, medan reläbrygga ska ta hand om de digitala signalerna.

5.2.1.1 Diodbryggan

Diodbryggan omvandlar $24 V_{AC}$ från regulatorm till $3 V_{DC}$ till I/O-kortet. Den tar hand om signalerna som i normala fall går till ställdonen. Egentligen består denna brygga av fyra likadana likriktarkretsar, som var och en tar hand om en signal.

Principutseendet för varje sådan krets följer nedan.



$$D1 = 1N4001$$

$$D2 = \text{lysdiod, } \varnothing 3\text{mm}$$

$$R1 = 390\Omega$$

$$R2 = R3 = 2.4\text{k}\Omega$$

$$C = 22\mu F$$

Figur 5.2: Principutseende för likriktarkrets

När dioden lyser innebär det att motsvarande signal är låg. Då går all ström mellan V_x och G_0 . Då dioden är släckt är motsvarande signal hög, vilket innebär att motsvarande signal är hög.

Anledningen till att en parallellkoppling R_2 - R_3 användes istället för en enda resistor är den relativt höga effektutvecklingen. Som mest ligger det över G - G_0 $24V$. Spänningen ger då upphov till en ström

$$\left(I = \frac{U}{R} \right) \text{ på } 20\text{ mA} \text{ om en resistor på } 1.2\text{ k}\Omega \text{ skulle användas, vilket i sin tur innebär en}$$

effektutveckling $(P = U \cdot I^2)$ över resistorn på 440 mW . Om en parallellkoppling används blir denna effekt bara hälften så stor över var och en av resistorerna.

Som en extra åtgärd valdes alla resistorer till metallfilmsresistorer, som tål högre effekter.

5.2.1.2 Reläbryggan

Reläbryggan består av tre reläer, där varje relä tar hand om en digital insignal till regulatorm: fönsterkontaktssignal, närvarosignal och koldioxidssignal. När fönstret är stängt är dess motsvarande signal hög. När någon befinner sig i zonen är dess motsvarande signal hög. När koldioxidgivaren inte är ansluten är dess motsvarande signal hög.

6 SIMNON

6.1 Allmänt

Simnon är ett program som hjälper en att lösa differentiella ekvationer och differensekvationer. Ekvationen(-erna) skrivs in i en fil (fil.t) och sedan ges olika begynnelsevärden. Man kan därefter välja att simulera dessa ekvationer över ett visst tidsintervall för att se deras beteende, vilket visas grafiskt, antingen medan man simulerar eller efter simuleringens slut.

Detta simuleringsprogram används antagligen mest för att simulera komplicerade, dynamiska system, eventuellt indelade i mindre system, vilka skrivs in som differentiella (alternativt differensekvationer ifall dessa är i diskret tid) ekvationer. Dessa mindre system (som skrivs i filer av typen continuous system (1) och discrete system (2)) kopplas samman med hjälp av filer av typen interconnecting system (3) för att få ett sammansatt system. I (1) och (2) definieras INPUT- och OUTPUT-variabler som sedan knyts ihop i (3). Exempelvis kan man definiera en PID-regulator som en modell och den miljö den ska reglera som en annan. Genom att definiera regulatorns utsignal(er) som insignal(er) till miljön och miljöns utsignal(er) som insignal(er) till regulatorn har man fått ett slutet reglersystem. Möjlighet finns som tidigare nämnts att plotta och spara variablerna under exekveringens gång för att följa beteendet hos systemet utgåendes från olika begynnelsevärden.

För att ytterligare öka användbarheten har Simnon två "developers kit": SADK och SimRPC.

6.2 SADK (Simnon AddOn Developers Kit)

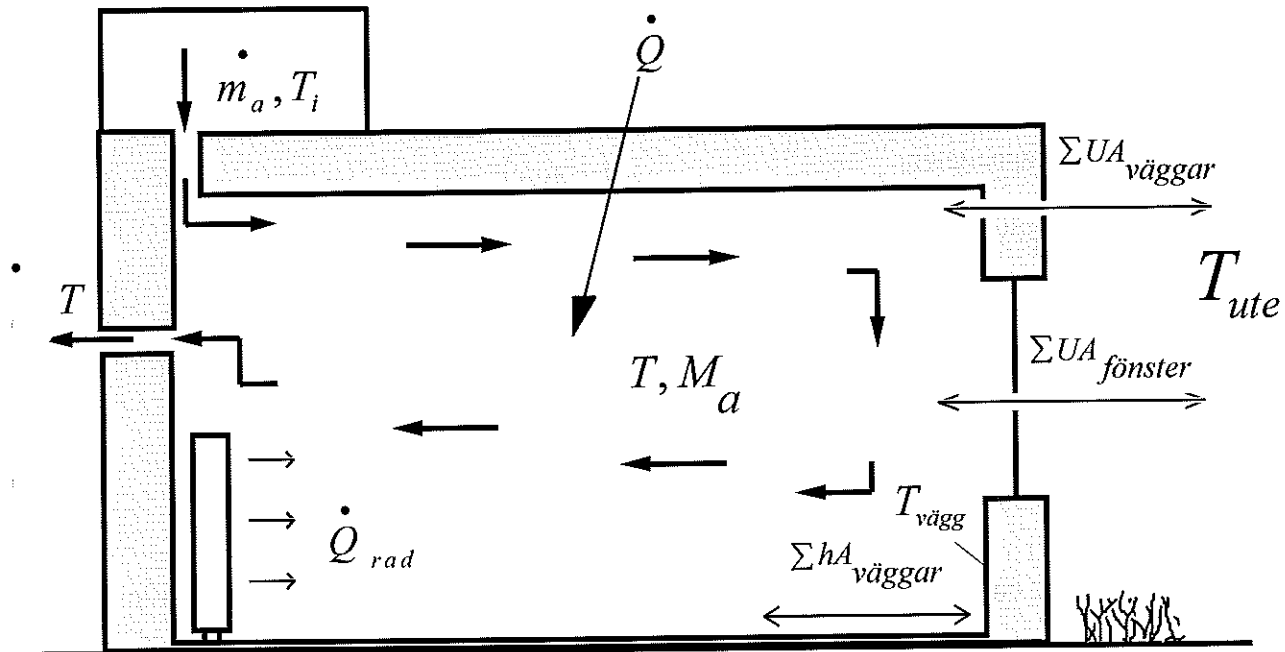
Med SADK får man möjlighet att lägga till s.k AddOns. En AddOn är en DLL (Dynamic Link Library), en speciell fil som kan anropas av flera olika Windows-program. I en Simnon AddOn skriver man egna funktioner som kan utnyttjas av alla modeller man skriver in. Dessa funktioner kan exempelvis vara nya reserverade ord, nya fönstertyper till Simnon, nya menyer, nya matematiska funktioner osv. I fallet med testbänken har med hjälp av SADK skrivits funktioner som läser och lägger ut spänning till de olika in- och utgångarna på I/O-korten.

6.3 SimRPC

Förutom SADK finns SimRPC. Med hjälp av SimRPC skapar man en klient-server applikation, där Simnon är servern, som då jobbar i bakgrunden. Klienten skriver man själv, förslagsvis i form av ett grafiskt användargränssnitt. Fördelen med en sådan applikation är att man under simuleringens gång kan ändra de ingående variablerna från klienten utan att behöva avbryta simuleringen. Dessutom, beroende på klientens utformning, kan en oinvidg Simnon-användare köra en simulering utan att direkt behöva förstå de bakomliggande mekanismerna.

7 Fysikaliska modeller

7.1 Fysikalisk modell av ett rum



Figur 7.1: Skiss över typiska värme- och massflöden

Den dynamiska modellen av temperaturen i ett rum påverkas av väldigt många faktorer. Det beror på väggarnas och takets yta, storleken på fönstrena, om det finns möbler i rummet, radiatorernas temperatur, antal personer i rummet, om det finns värmeavgivande apparater (t.ex TV, radio, lampor) i rummet för att bara nämna några av dessa faktorer.

Den fysikaliska modell som jag utgick ifrån är definierad av Dr. Bent A. Borresen i artikeln "Thermal room models for control analysis", publicerad i tidskriften ASHRAE Transactions 87 (1981). Denna modell är känd för att ge trogna short-term och steady-state svar, men är lite för snabb däremellan.

Short-term svaret är definierat i detta fall som svaret till en laststörning som ligger inom tidsperioden 0.5 till 10 minuter från det att lasten har lagts på.

$$\frac{\partial T}{\partial t} \cdot M_a c_{pa} = \dot{Q} + \dot{Q}_{rad} + \dot{m} c_{pa} \cdot (T_i - T) + \sum_{ytterväggar} UA \cdot (T_{ute} - T) + \sum_{fönster} UA \cdot (T_{ute} - T) \quad (7-1)$$

där

T temperaturen i rummet (K)

T_i den inkommande luftens temperatur (K)

7.2.1 Dagligt temperaturmedelvärde

Enligt denna modell kommer det dagliga temperaturmedelvärdet att variera sinusformigt under ett år:

$$\bar{T}_{\text{dygn}} = \frac{\bar{T}_{\text{juli}} + \bar{T}_{\text{januari}}}{2} + \frac{\bar{T}_{\text{juli}} - \bar{T}_{\text{januari}}}{2} \cdot \sin(\omega_2 \cdot (t - t_2)) \quad (7-2)$$

där

\bar{T}_{dygn} den dagliga medeltemperaturen (K)

\bar{T}_{januari} månadsmedeltemperatur i januari (K)

\bar{T}_{juli} månadsmedeltemperatur i juli (K)

$\omega_2 = \frac{2\pi}{360 \cdot 24 \cdot 3600}$, vinkelhastigheten (rad/s)

t tiden, $t = 0$ vid årets början (s)

$t_2 = \frac{360 \cdot 24 \cdot 3600}{4}$ (s)

Lite om variablerna:

Ett år i modellen är 360 dagar, dvs 30 dagar i varje månad. Svängningens period blir då 360 dagar, vilket förklarar vinkelfrekvensen ovan.

Enligt data från SMHI gäller det för Malmö-området att $\bar{T}_{\text{januari}} = -1^\circ \text{C}$ och $\bar{T}_{\text{juli}} = -17.5^\circ \text{C}$.

t_2 är till för att förskjuta sinuskurvan. Det medför att den högsta medeltemperaturen inträffar för 180:e dagen på året, dvs 1:a juli, medan lägsta medeltemperaturen inträffar 1:a januari.

7.2.2 Ögonblickligt temperaturvärde

Under dagen oscillerar temperaturen runt den dagliga medeltemperaturen och har sitt maximum vid klockan 2.00 p.m. Denna oscillation kommer också att vara sinusformig. Man måste även ta hänsyn till att nattetid är denna oscillations amplitud mindre än under den ljusa tiden på dygnet

$(A_{\text{natt}} < A_{\text{dag}}, \text{se nedan})$.

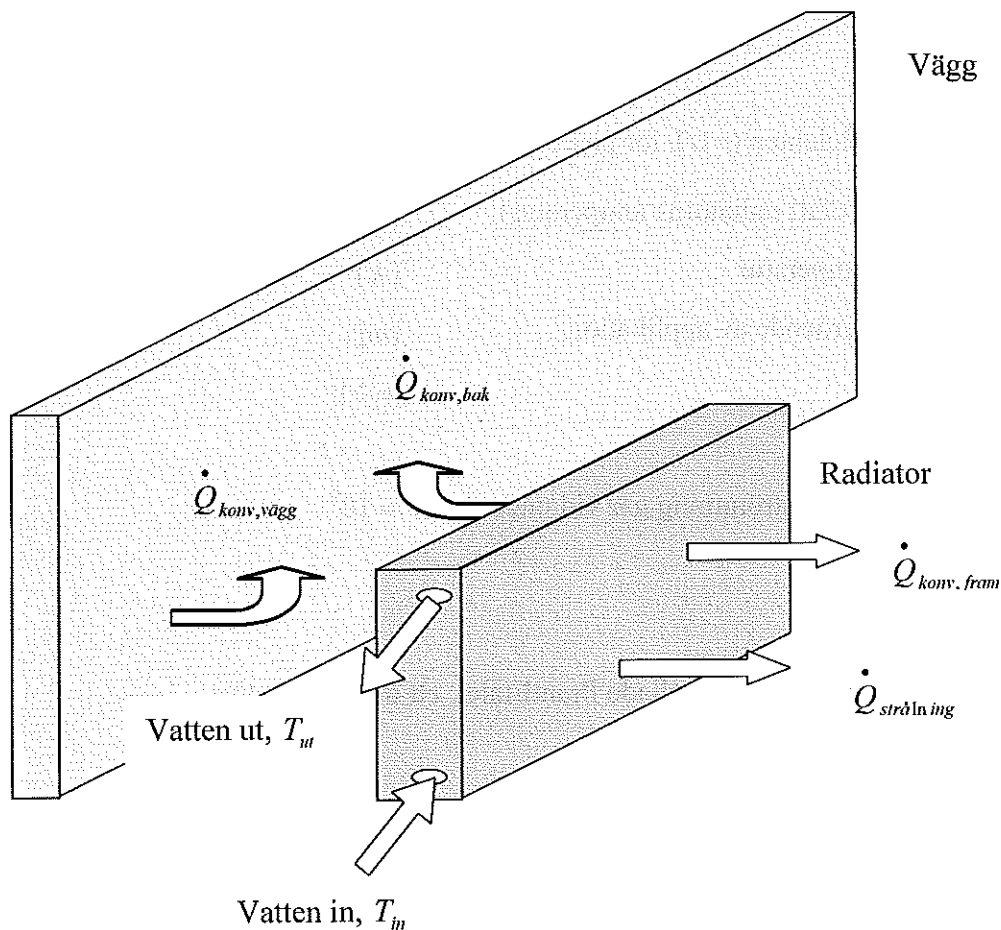
Svängningens period är ett dygn.

A_{natt} och A_{dag} skrivs in i en modell över hela året som stokastiska variabler, för att få variation från ett dygn till nästa. För att åstadkomma detta har en rektangulär funktion använts, så att alla värden får samma sannolighet. A_{natt} kommer att variera i området

$$0 < A_{natt} \leq 3 \text{ (K)} \text{ medan } A_{dag} \text{ kommer att variera i området } 0 < A_{dag} \leq 7 \text{ (K)}.$$

7.3 Fysikalisk modell av en radiator

Värmeflödet från en radiator är icke-linjärt och beror på radiatorns temperatur. Värme avges dels som strålningsvärme, dels som konvektiv värme.



Figur 7.3: Radiator och typiska värmeflöden från den

$$\dot{Q}_{tot} = \dot{Q}_{strålning} + \dot{Q}_{konv,fram} + \dot{Q}_{konv,bak} + \dot{Q}_{konv,vägg}$$

c_p	specifik värmekapacitet	(J/kgK)
μ	dynamisk viskositet	$(Pa \cdot s)$
k	värmekonduktivitet	$(W/m^2 K)$

I vårt fall, i det aktuella temperaturområdet för luft, varierar Prantl-talet så pass lite att det kan antas konstant, $Pr = 0.72$ (enligt tabell konstant för $-40^\circ C < T < 100^\circ C$).

7.3.3 Nusselt-talet

När det gäller Nusselt-talet finns det ett antal formler som gäller, både teoretiskt och experimentellt framtagna. Svårigheten ligger i att Nusselt-talet blir olika även beroende på om det är laminärt eller turbulent strömning. Dessutom är Nusselt-talet inte konstant över en yta, därför används ett medelvärde i räkningarna. Eftersom elementet dessutom har en framsida som vetter mot rummet och en baksida som vetter mot väggen blir det helt olika gränsvillkor, vilket också leder till olika formler.

För framsidan gäller, enligt Bejan (1993) och Eckert (1987):

$$\overline{Nu}_f = 0.555 \cdot (Gr \cdot Pr)^{0.25} \quad \text{om } Gr < 10^9$$

$$\overline{Nu}_f = 0.021 \cdot (Gr \cdot Pr)^{0.40} \quad \text{om } Gr > 10^9$$

För baksidan gäller, enligt Miyatake(1973):

$$\overline{Nu}_b = 0.58 \cdot \left(1 + 0.165 \cdot \left(\frac{\Delta T_h}{\Delta T_l} \right)^{0.36} \right) \cdot \left(\frac{D}{H} \cdot Gr \cdot Pr \right)^{0.25} \quad (7-7)$$

där

$$\Delta T_h \quad |T_{zon} - T_h| \quad (K)$$

$$\Delta T_l \quad |T_{zon} - T_l| \quad (K)$$

$$T_h \quad \text{radiatorns högsta temperatur} \quad (K)$$

$$T_l \quad \text{radiatorns lägsta temperatur} \quad (K)$$

$$D \quad \text{avståndet mellan radiatoren och väggen} \quad (m)$$

$$H \quad \text{radiatorns höjd} \quad (m)$$

ε	emittansen, radiatorns framsida	(-)
σ	Stefan-Boltzmanns konstant ($= 5.670 \cdot 10^{-8}$)	($W/m^2 K^4$)

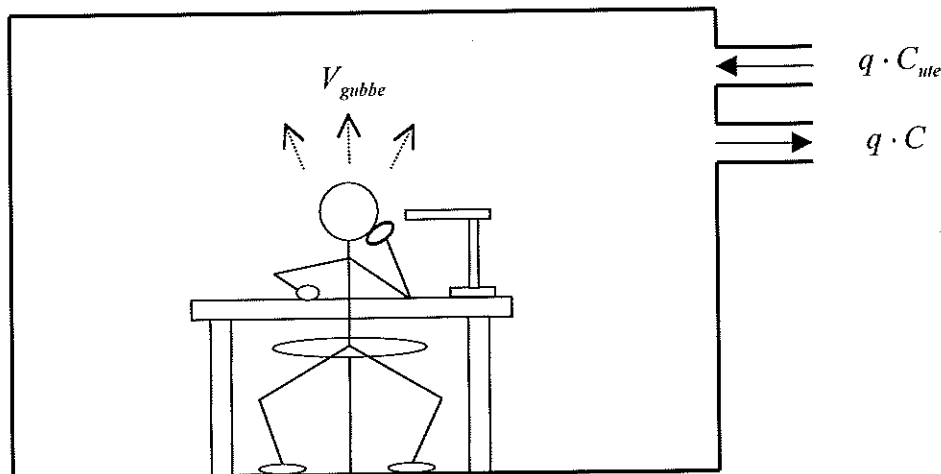
7.3.4.1 Lite om $T_{radiator}$

$T_{radiator}$ är radiatorns medeltemperatur och används för att underlätta beräkningarna. Vattnet i matarröret till radiatorn håller en viss temperatur T_m som är beroende av utetemperaturen och regleras i huvudledningen. När sedan vattnet har passerat en radiator är dess temperatur T_{ut} . Vi får:

$$T_{radiator} = \frac{T_m + T_{ut}}{2} \tag{7-15}$$

T_m :s beroende av utetemperaturen är icke-linjär och bestäms av regulatören till huvudmatarröret.

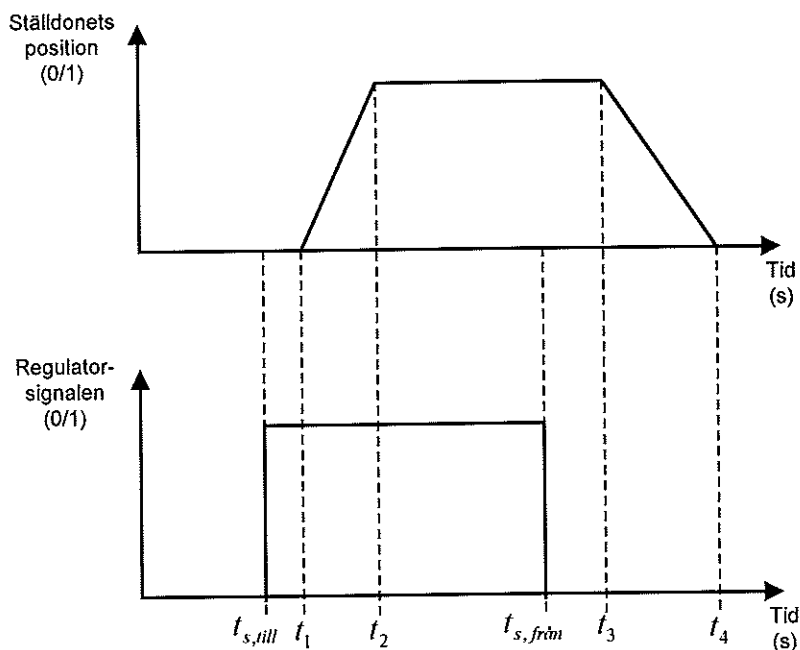
7.4 Fysikalisk modell av koldioxidkoncentrationen



Figur 7.4: Koldioxidmodell

Koldioxidkoncentrationen i ett rum påverkas bara av antalet människor som befinner sig där och hur länge vistelsen varar. Genom ventilation bortförs koldioxiden. Den fysikaliska modellen utgår från antagandet att fullständig omblandning råder i rumsluften, dvs koncentrationen vid varje tidpunkt är lika i rummets alla delar och frånluftens koncentration är lika med rumsluftens koncentration. Det antas också att det inte förekommer något läckage av frånluft till tilluft.

$$\frac{dC}{dt} \cdot V = n \cdot V_{gubbe} + q \cdot C_{ute} - q \cdot C \tag{7-16}$$



Figur 7.5: Dynamisk modell av thermoställdon

Regulatorsignalen slår om från 0 till 1 vid $t_{s,till}$, vilket innebär att spänningen till värmeelementet som värmer upp vaxkroppen slås på. När vaxkroppens temperatur har uppnått en viss kritisk temperatur börjar den expandera vid t_1 . Denna expansion leder till att ställdonet öppnar linjärt och avstannar vid t_2 , då den har uppnått sitt ändläge, dvs är fullt öppet. Vid tidpunkten $t_{s,från}$ slår regulatorsignalen om från 1 till 0, vilket innebär att spänningen till värmeelementet stängs av. Vaxkroppen börja svalna av, vilket åstadkoms av den omgivande luften, och när den kritiska temperaturen uppnås vid tiden t_3 börjar den krympa. Krympningen leder till att ställdonet stänger linjärt och avstannar vid t_4 , då den har uppnått sitt andra ändläge, dvs är fullt stängt.

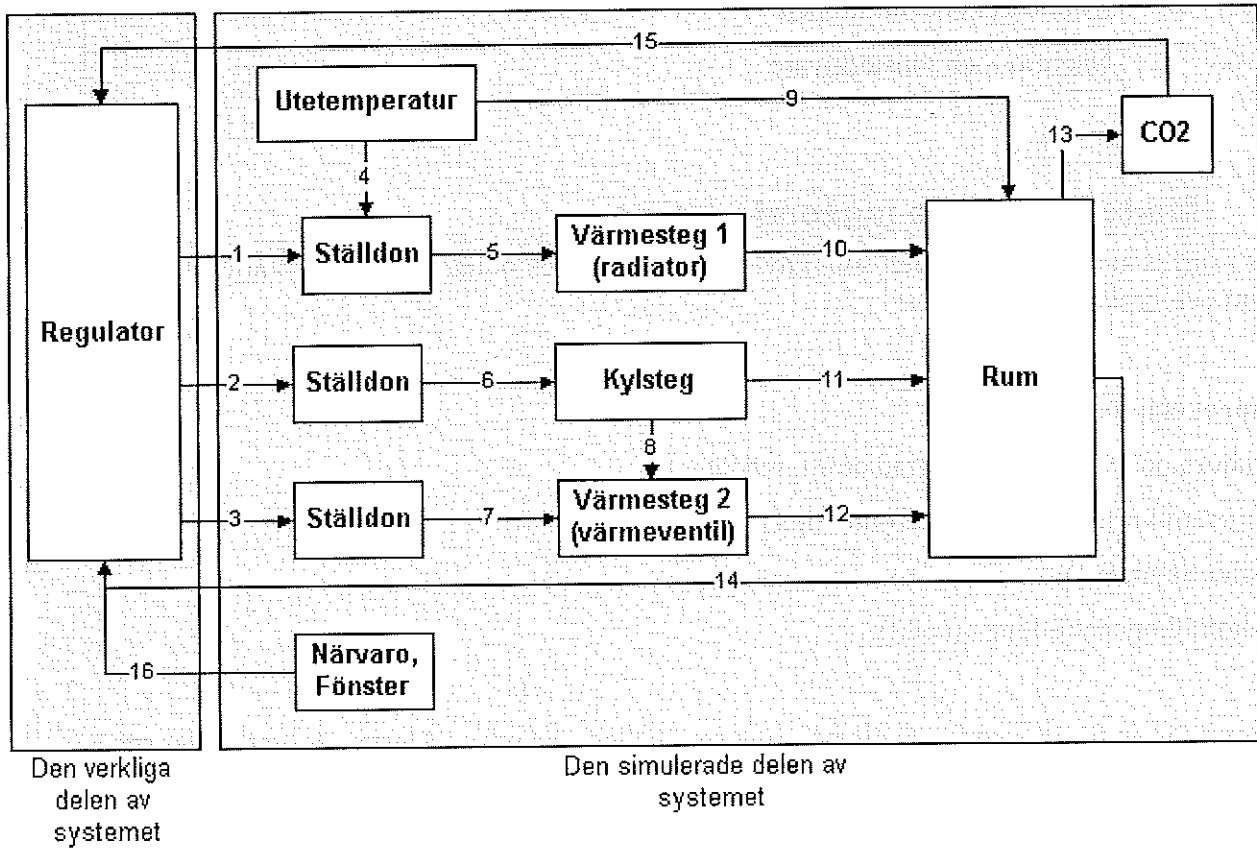
Beteckningar:

$t_1 - t_{s,till}$	Dödtiden innan ställdonet börjar öppna
$t_2 - t_1$	Ställdonets gångtid (i detta fall den tid det tar att röra sig från helt stängt till helt öppet)
$t_3 - t_{s,från}$	Dödtiden innan ställdonet börjar stänga
$t_4 - t_3$	Ställdonets gångtid (i detta fall den tid det tar att röra sig från helt öppet till helt stängt)
$t_{till} = t_{s,från} - t_{s,till}$	Tidsperioden som spänningen har varit tillslagen
$t_{från} = t_{s,till} - t_{s,från,gammal}$	Tidsperioden som spänningen har varit frånslagen

Utgåendes från föregående t_{till} räknas $t_3 - t_{s,från}$ fram och utgåendes från föregående $t_{från}$ räknas $t_1 - t_{s,till}$ fram. Det är denna modell som inte har kunnat ställas upp en fysikalisk modell för, utan experimentella data används för att få fram tider och samband mellan dem.

8 Det simulerade systemet

8.1 Beskrivning



Figur 8.1: Det simulerade systemet. Se figur 2.4 för en skiss av en verklig applikation

Signalnummer	Betydelse
1	Styrsignal till ställdon
2	Styrsignal till ställdon
3	Styrsignal till ställdon
4	Utetemperaturen (radiatorns maxtemperatur är beroende av denna)
5	Radiatorns temperatur
6	Öppningsvinkeln hos luftintaget
7	Värmeventilens temperatur
8	Luftflödet
9	Utetemperaturen
10	Värmetillskottet från värmesteg 1
11	Luftflödet till rummet

Två DLL-filer, så kallade AddOns, har implementerats och hör till testbänken. De är:

- RWIO.dll
- X100.dll

RWIO.dll (står för **Read and Write from/to I/O-cards**) innehåller alla funktioner som hanterar kommunikationen mellan I/O-korten och Simnon. Eftersom dessa funktioner är generella kan de användas även av andra Simnon-applikationer förutom testbänken. Funktionerna är som följer:

- **Aout(value, channel)**: lägger ut spänningen <<value>> på den analoga utgången <<channel>> på Analog Devices RTI-815-kortet;
- **Ain(channel)**: läser av spänningen från den analoga ingången <<channel>> på Analog Devices RTI-815-kortet; för att få ut denna spänning till en variabel och kunna använda det i sin modell måste denna funktion användas som i följande exempel:

...

```
in=Ain(0)
```

```
// läser av spänningen på ingång 0 och sparar datastrukturen i variabeln in
```

```
volts=r_Ain(in,1)
```

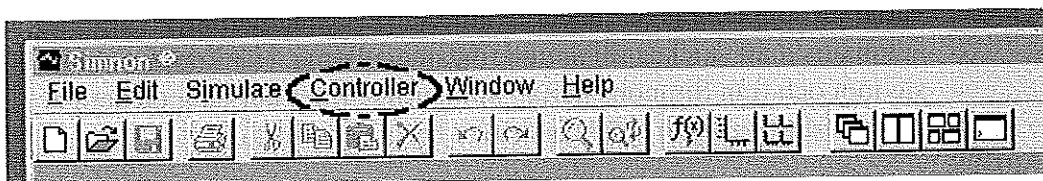
```
// från datastrukturen in lagras spänningsvärdet i variabeln volts, som kan användas
```

```
// fritt
```

...

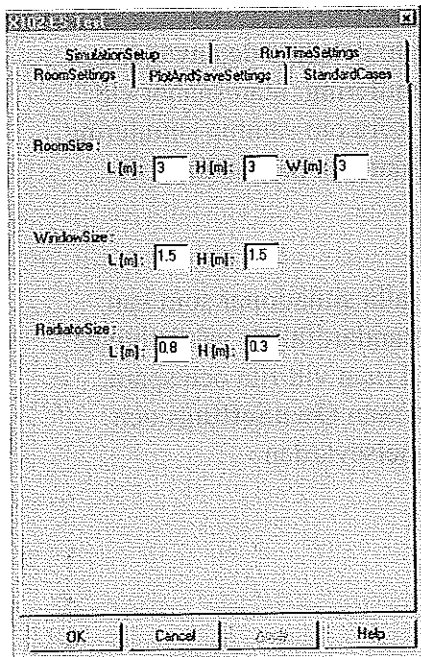
- cbAOut0, cbAOut1, cbAIn, cbDOut, cbDIn

X100.dll lägger lite extra funktionalitet till Simnon genom att lägga till en extra menyrad, som heter Controller. När man väl har startat Simnon behöver man bara klicka och välja i denna meny för att starta testbänken.



Figur 8.2: Den förändrade menyraden i Simnon

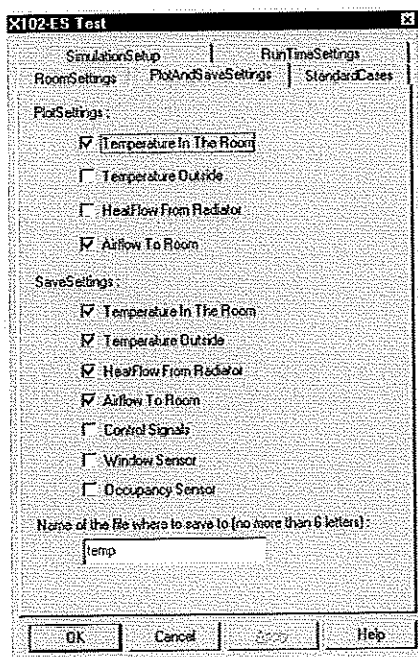
9.3 RoomSettings



Här är de rent fysiska variablerna som ska sättas, dvs rummets, fönstrets och radiatorns storlek. Kan vara nyttigt ifall man kan tänkas vilja se om till exempel värmesteget är underdimensionerat för aktuellt rum.

Figur 9.3: RoomSettings

9.4 PlotAndSaveSettings



Under simuleringens gång vill man säkerligen se hur olika variabler och regulatorsignaler varierar. Detta kan åstadkommas genom att i PlotSettings-underrubriken välja vad som kan vara intressant för tillfället.

Realtidsritandet tar mycket processorkraft! Detta kan i värsta fall leda till att simuleringen kraschar. För att undvika det kan man istället välja att spara sina variabler och signaler i en fil man själv bestämmer namnet på längst ner på sidan. Under underrubriken SaveSettings väljs vad som ska sparas.

Figur 9.4: PlotAndSaveSettings

OBS! Egentligen är det två filer som skapas: i den ena (som har samma namn som man valt) kommer rumstemperaturen, utetemperaturen, värmeflödet och luftflödet till rummet att lagras om de blivit valda, medan i den andra filen (som heter samma som ens valda namn, dock med bokstäverna "sd" framför, sd = signal databas) kommer de olika regulatorsignalerna, fönstergivarens och närvarogivarens status att lagras om man kryssat för detta.

10 Beskrivning av standardfall

10.1 Allmänt

Standardfallen är det primära man ska köra för att få testa de olika egenskaperna hos regulatorn. Testtiden varierar mellan 45 minuter (s.k. short-term) och 4 timmar (s.k. long-term), beroende på vad man testar.

Standardfallen är som följer:

Heat 1, Heat 2, Heat 3, Heat 4: Dessa fall är avsedda att testa hur fort regulatorn lyckas höja lufttemperaturen till börvärdet och hur väl den upprätthåller denna temperatur. Skillnaden mellan fallen är att de använder olika antal värmesteg och olika typer av ställdon till dessa värmesteg.

Cool: Fallet är avsett att testa hur fort regulatorn lyckas sänka lufttemperaturen till börvärdet och hur väl den upprätthåller denna temperatur.

CO2: Fallet är avsett att testa hur väl regulatorn lyckas hålla nere koldioxidkoncentrationen.

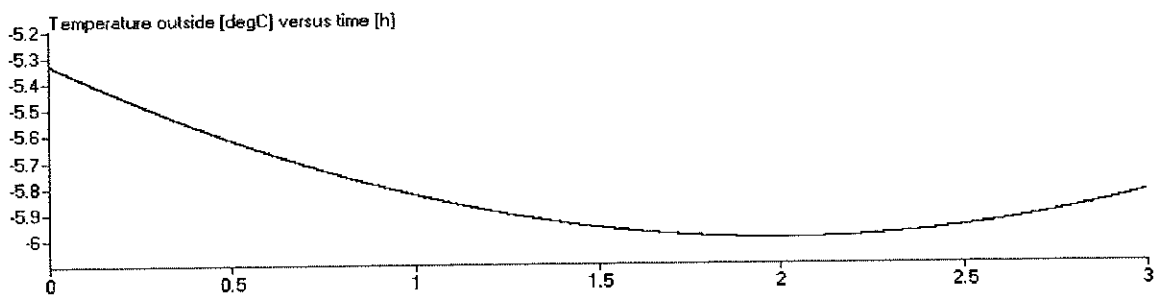
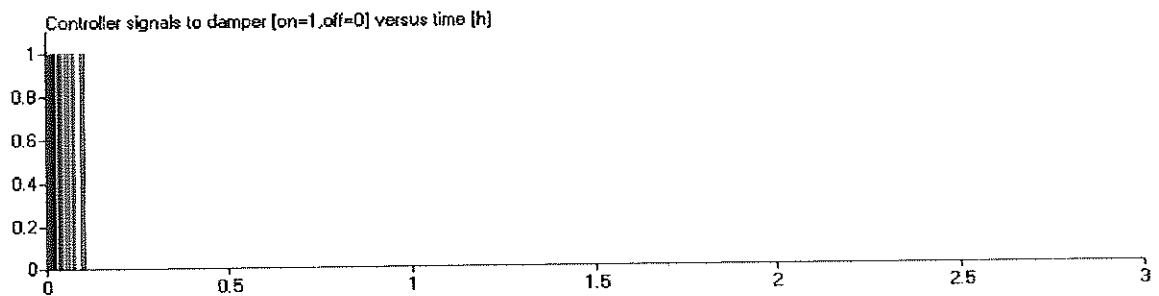
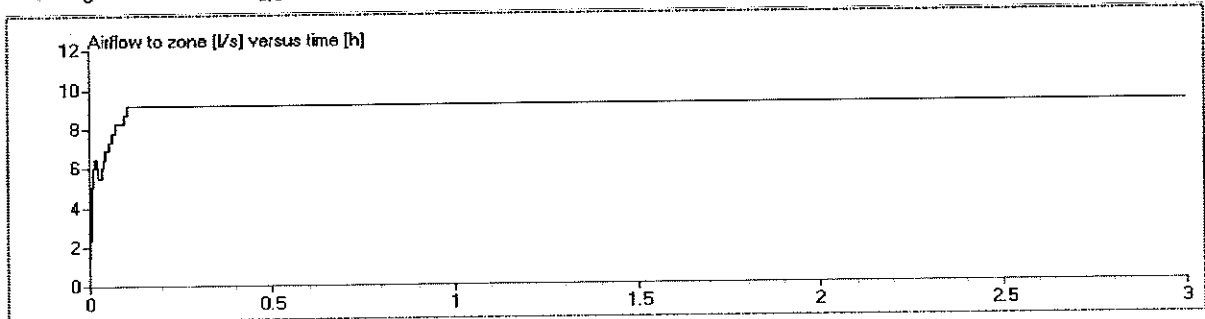
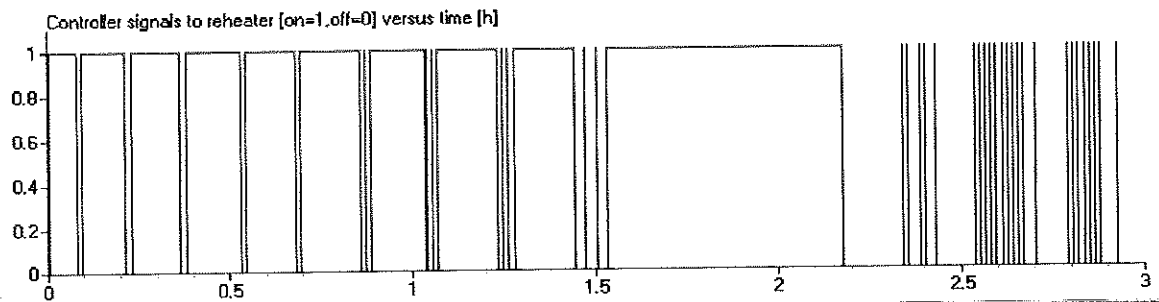
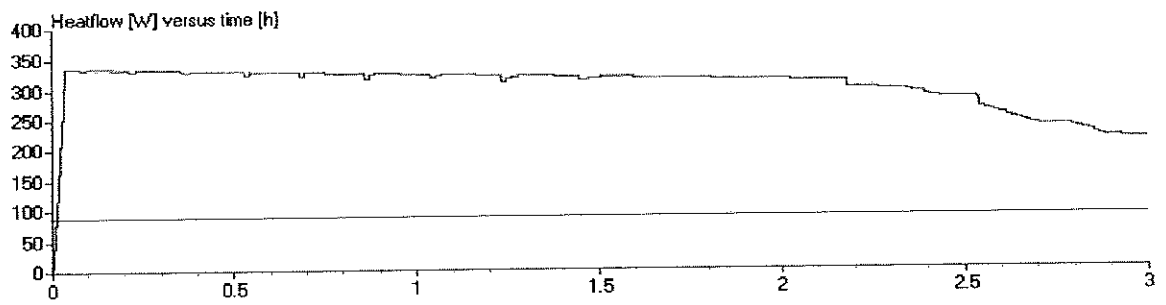
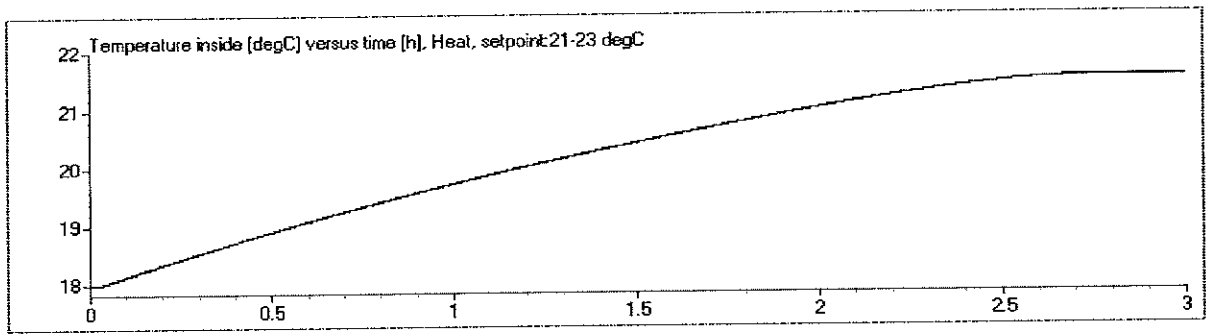
Window: Fallet är avsett att testa funktionaliteten hos fönsterkontakten hos regulatorn.

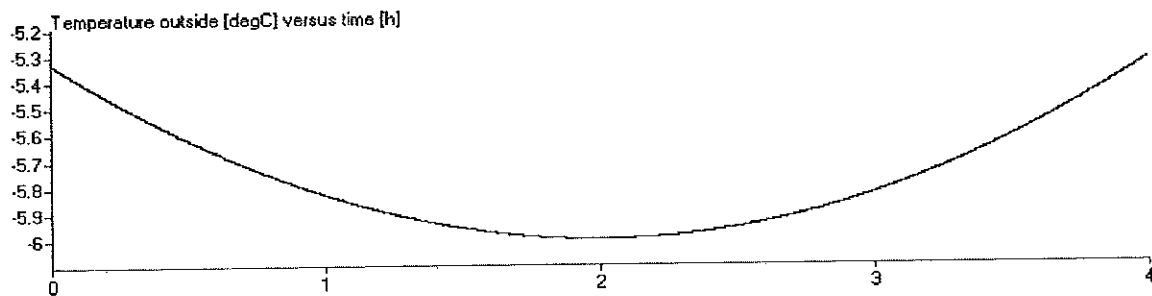
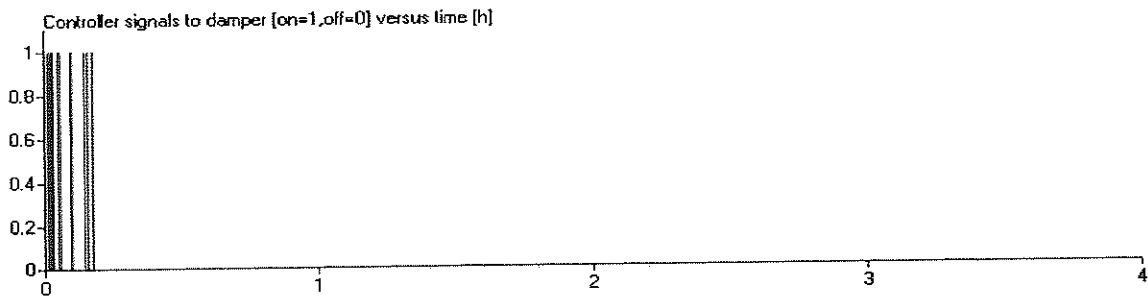
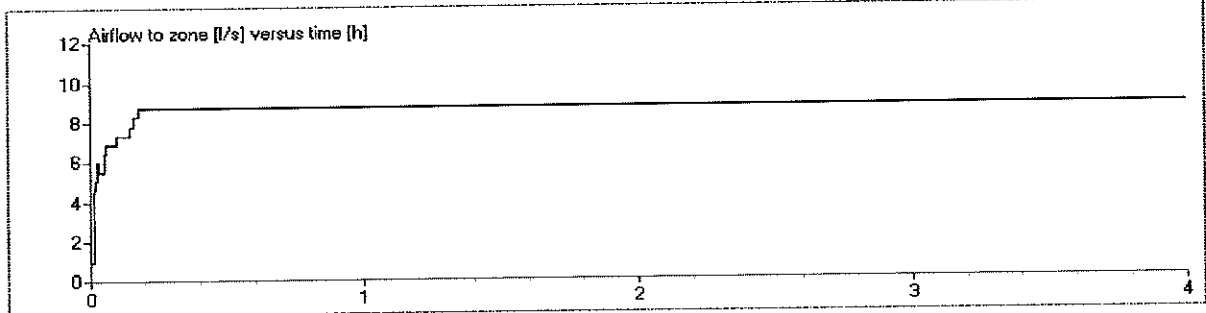
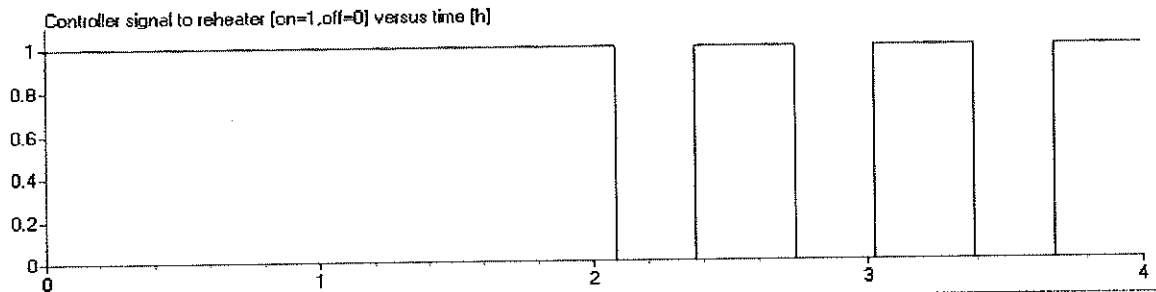
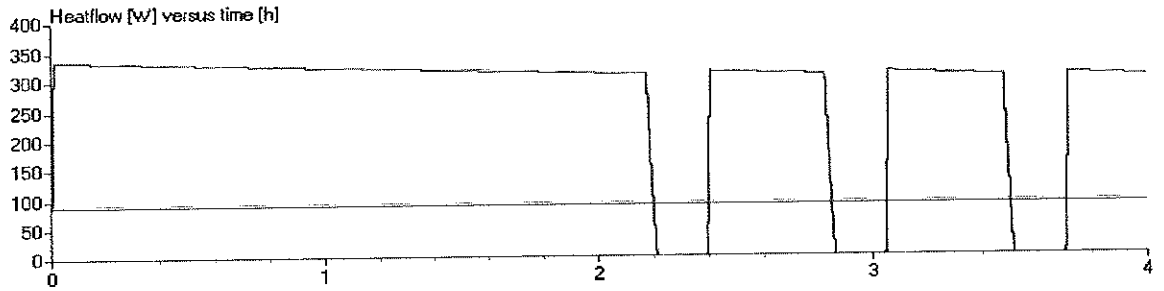
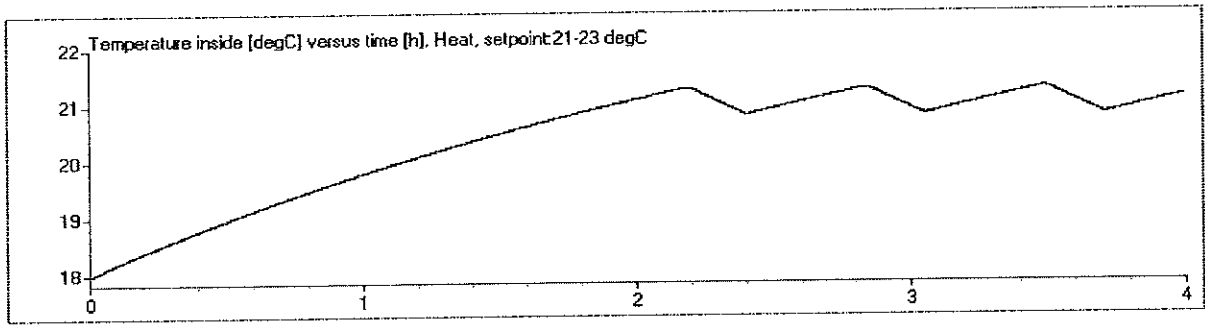
Occupancy: Fallet är avsett att testa funktionaliteten hos närvarogivaren hos regulatorn.

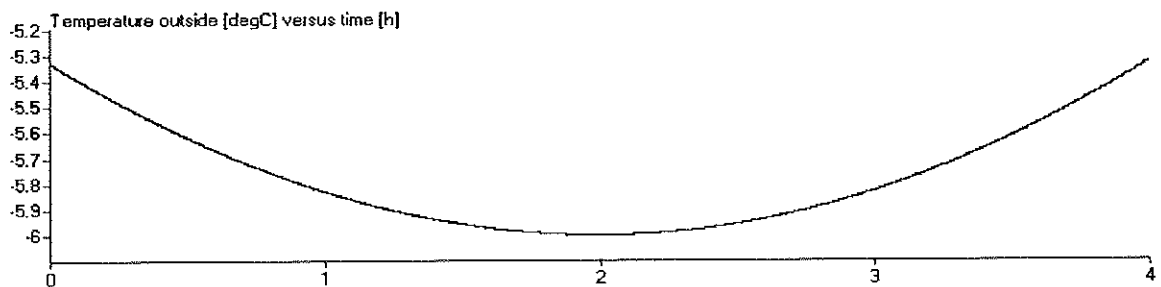
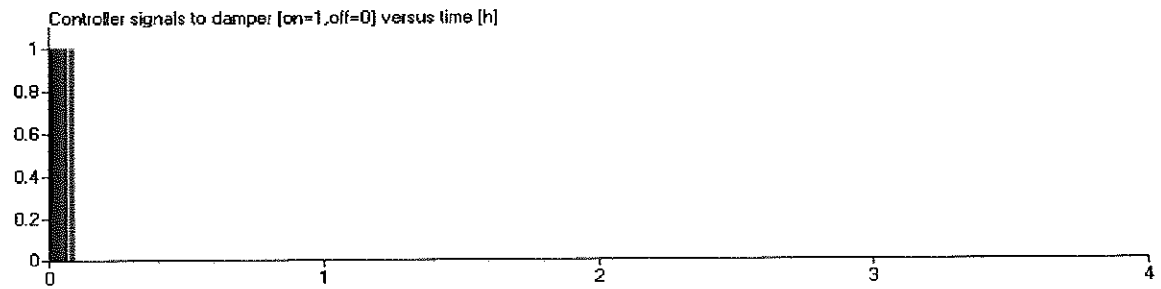
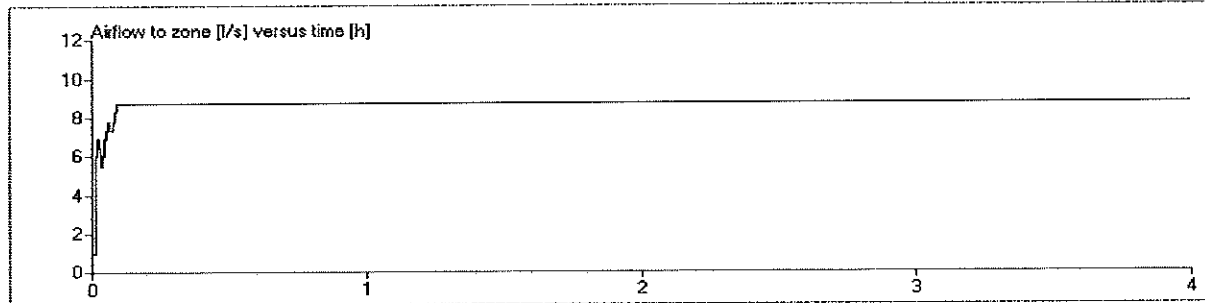
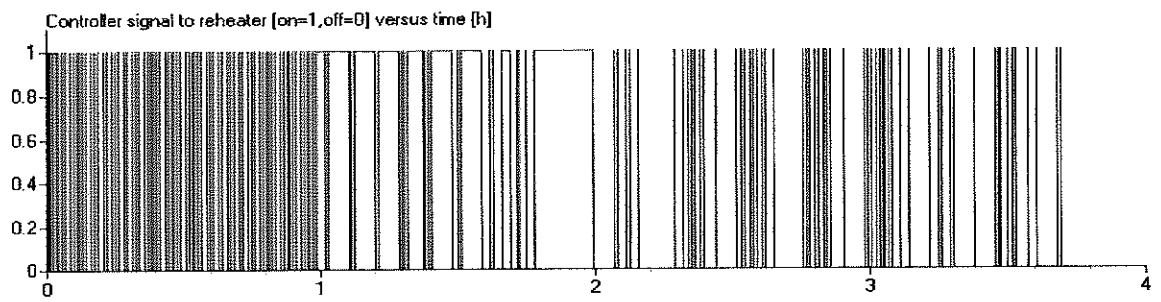
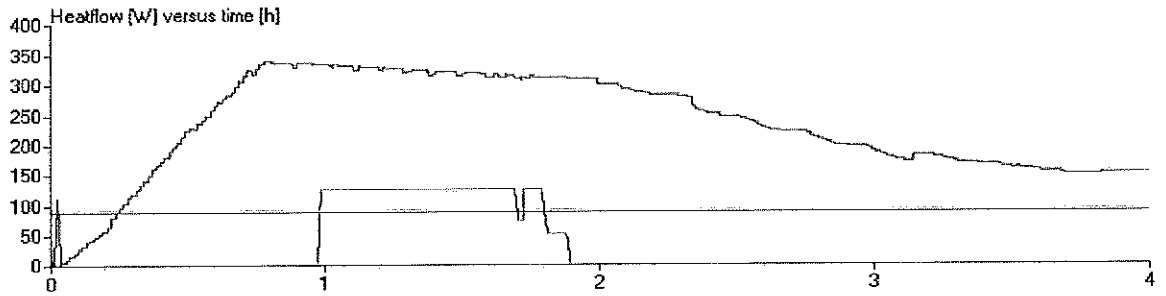
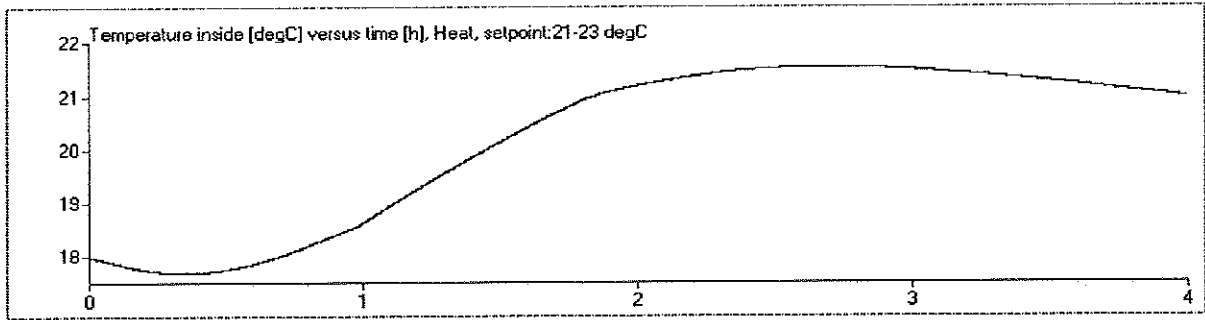
Auto 1, Auto 2, Auto 3, Auto 4: Fallen är avsedda att testa regulatorns beteende då både kyl- och värmebehov föreligger omväxlande samt, i något fall, reglera koldioxidkoncentrationen.

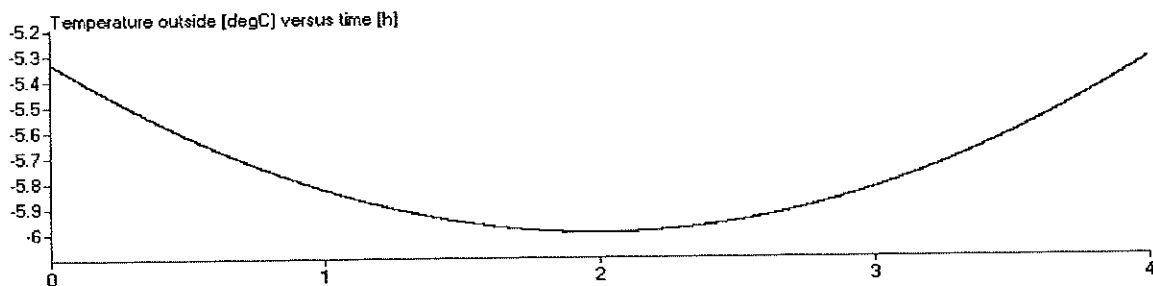
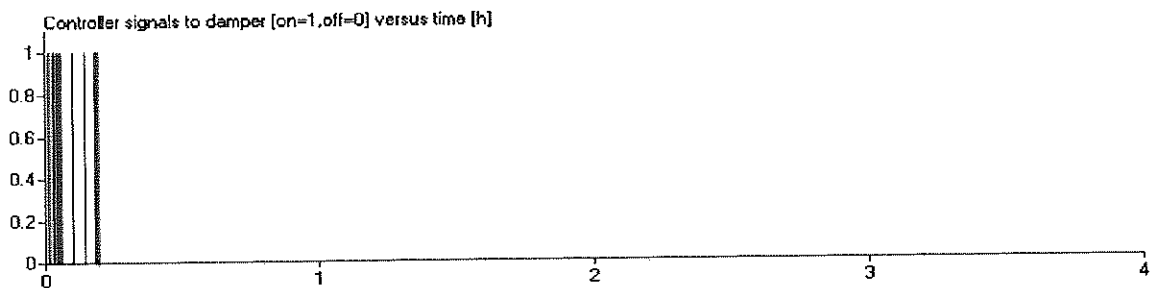
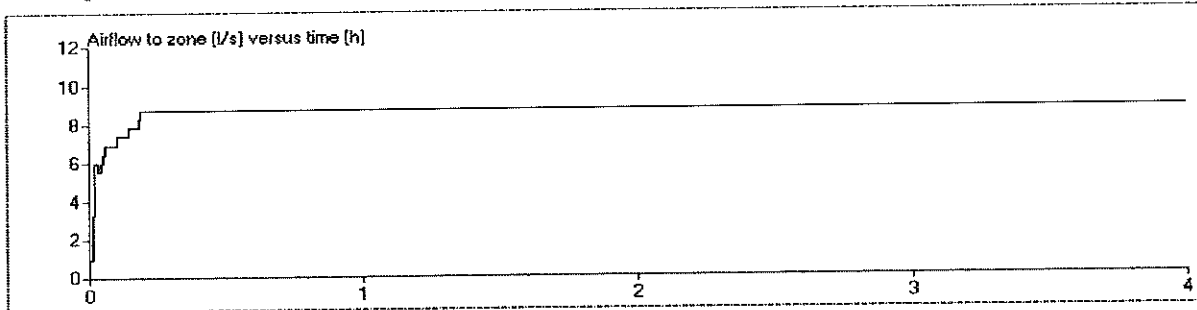
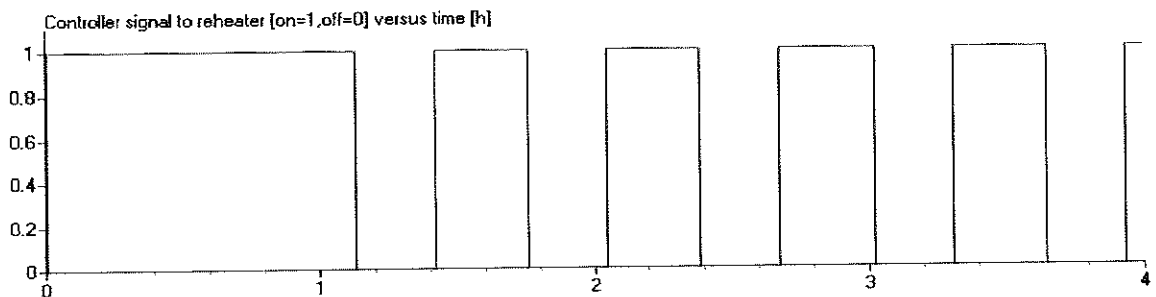
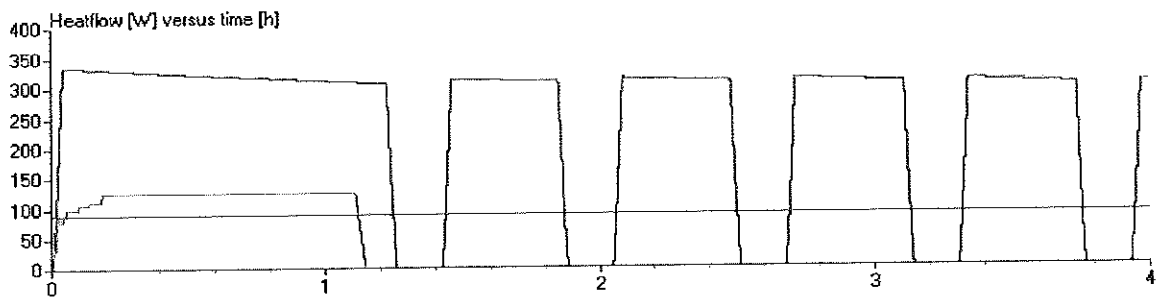
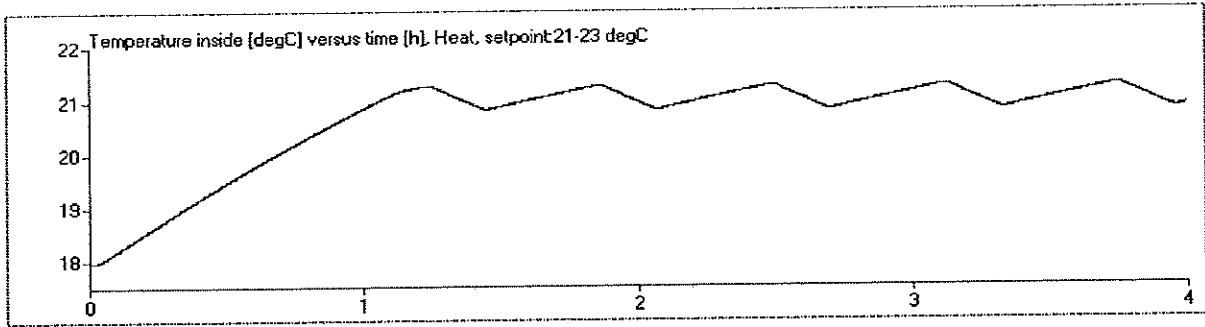
På de följande sidorna diskuteras dessa fall utgående från en testkörning för vart och ett.

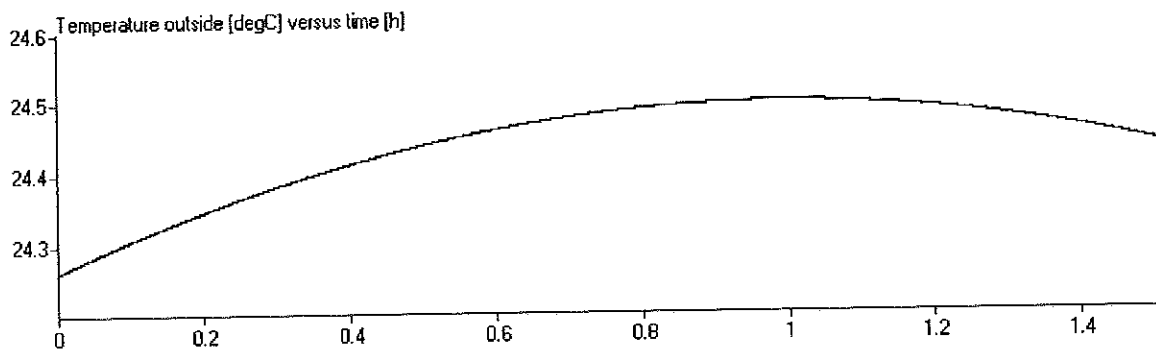
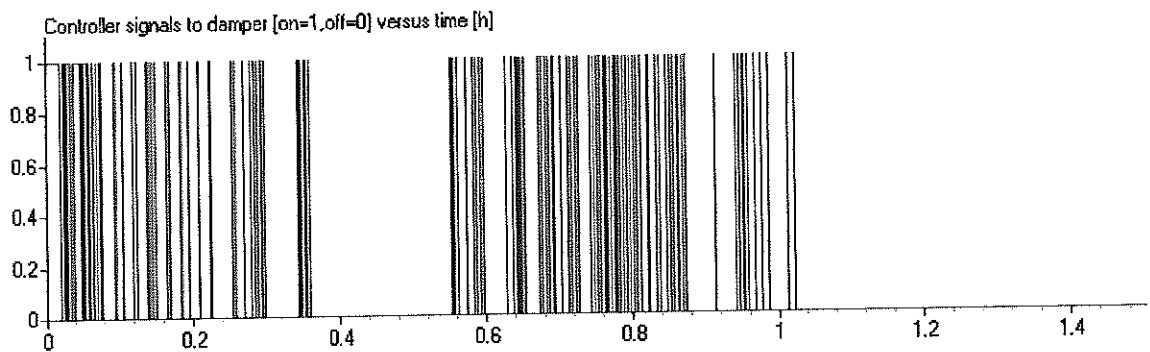
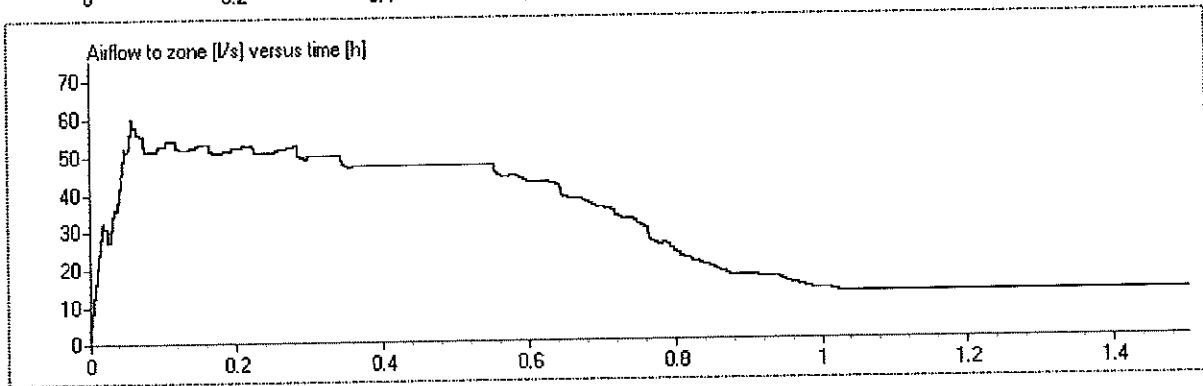
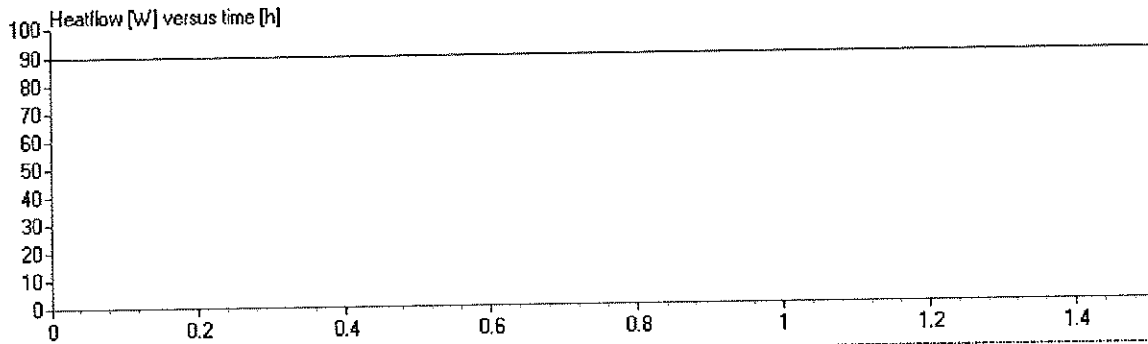
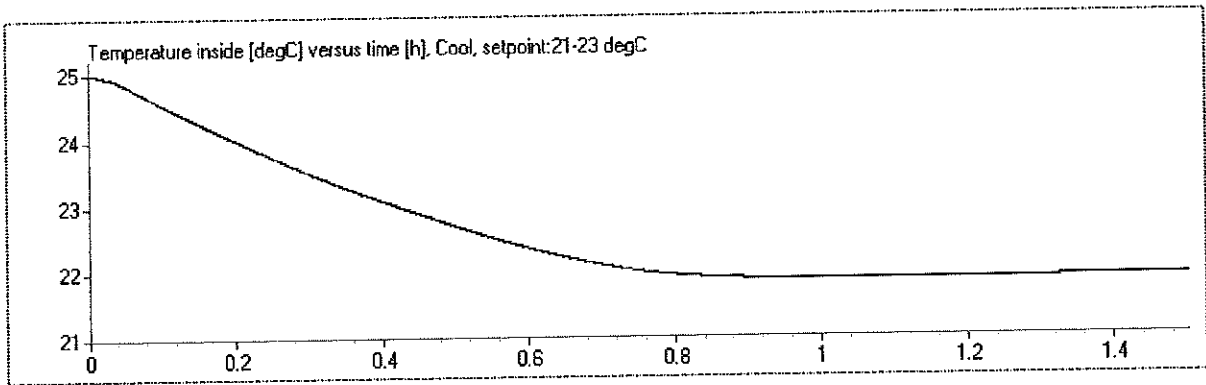
Observera att man ska bortse från de första minuterna av simuleringen, då regulatorn kan bete sig lite konstigt. Det är normalt eftersom den har kanske precis startats eller fortfarande matar ut signaler den räknat ut från en föregående simulering. Man ska vänta några minuter när man gjort nya regulatorinställningar så den kan anpassa sig till de nya förutsättningarna innan själva simuleringsfallet startas för att undvika detta.

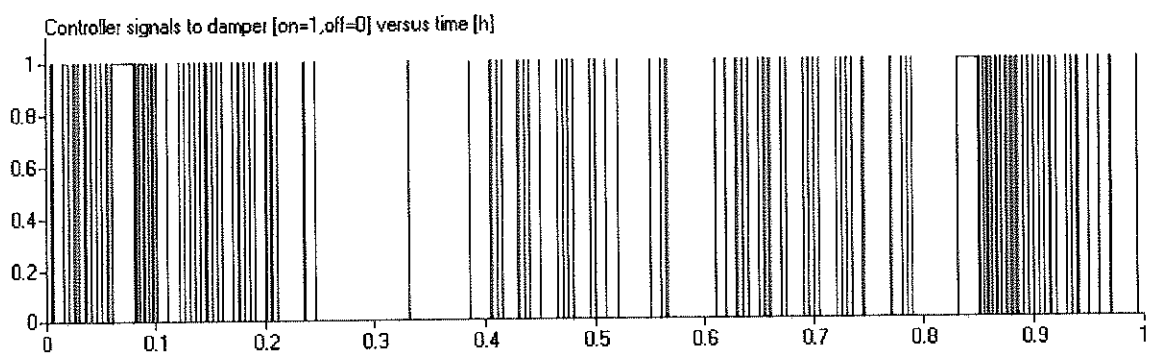
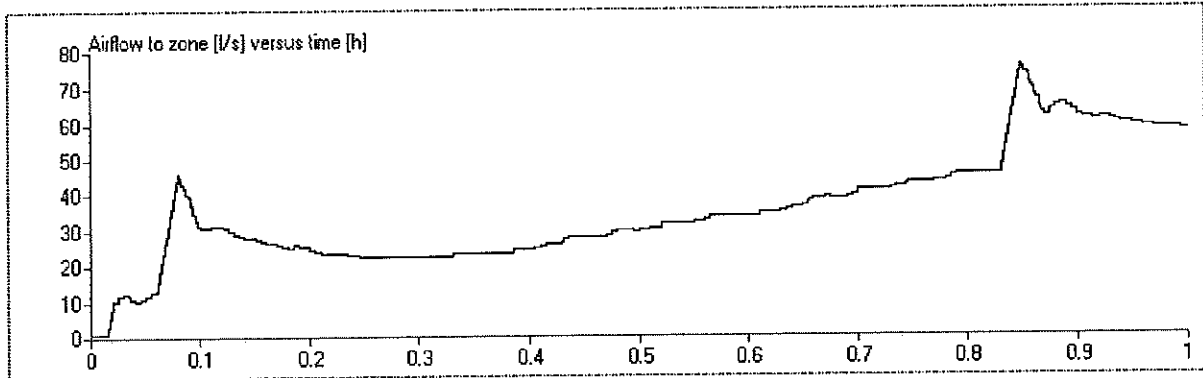
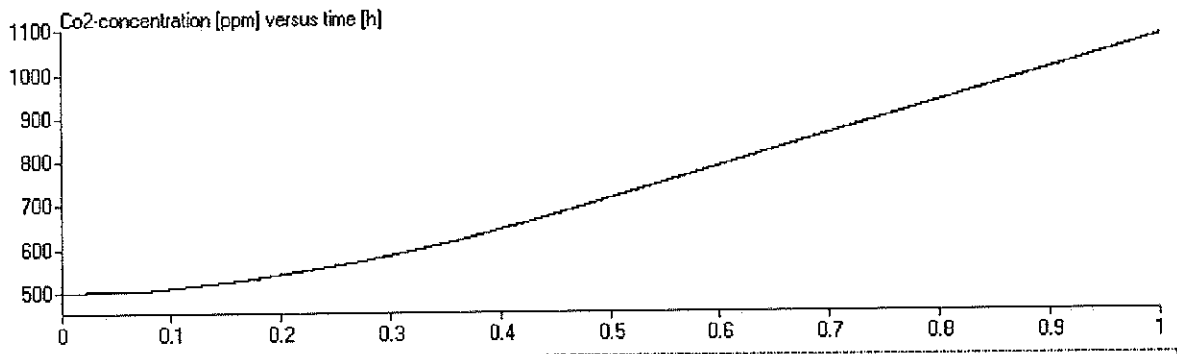
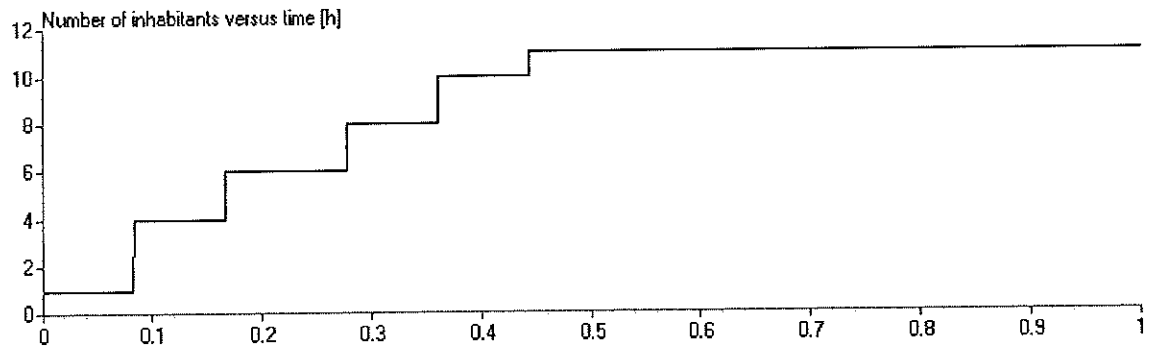
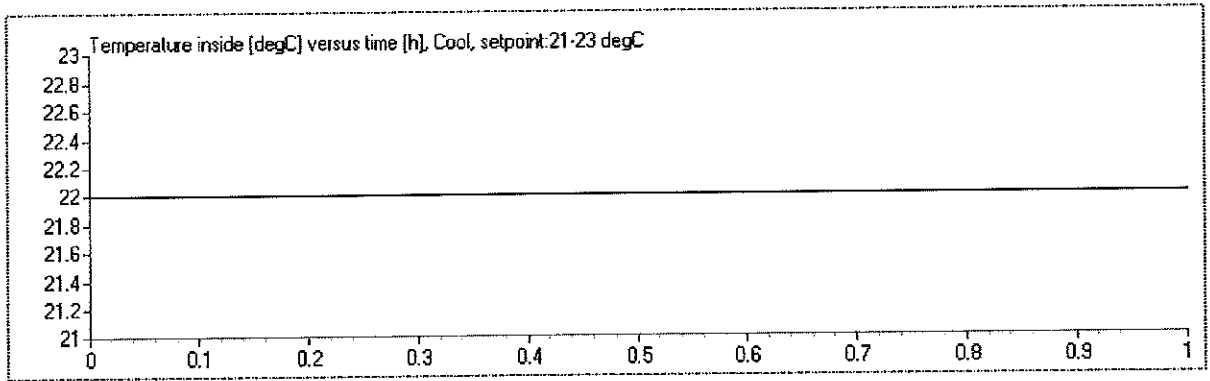


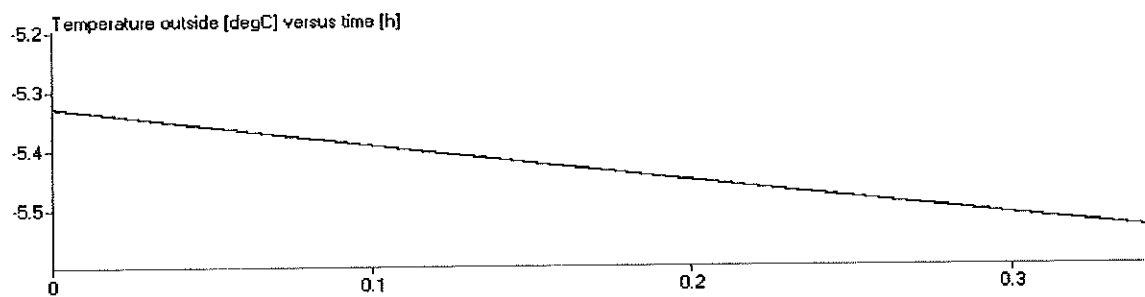
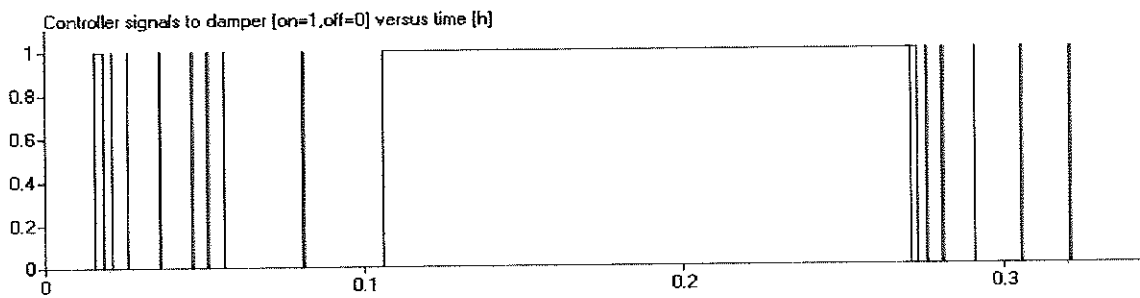
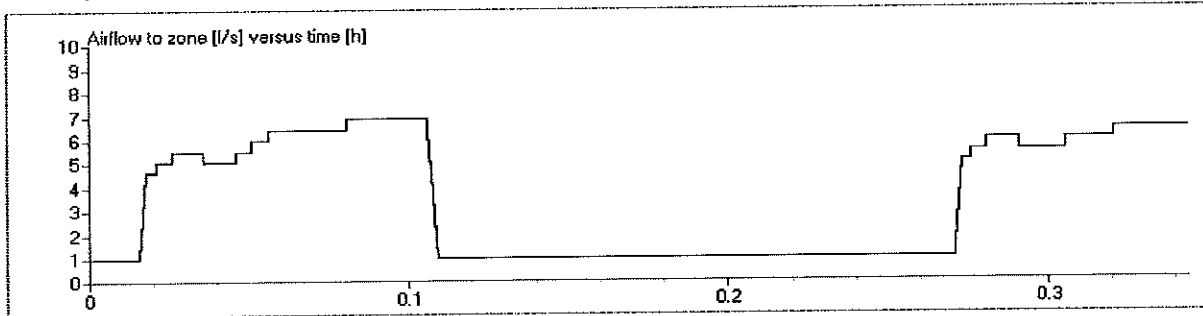
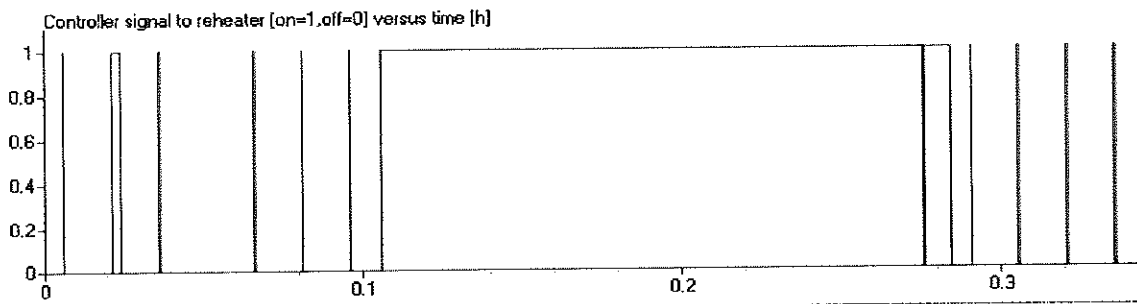
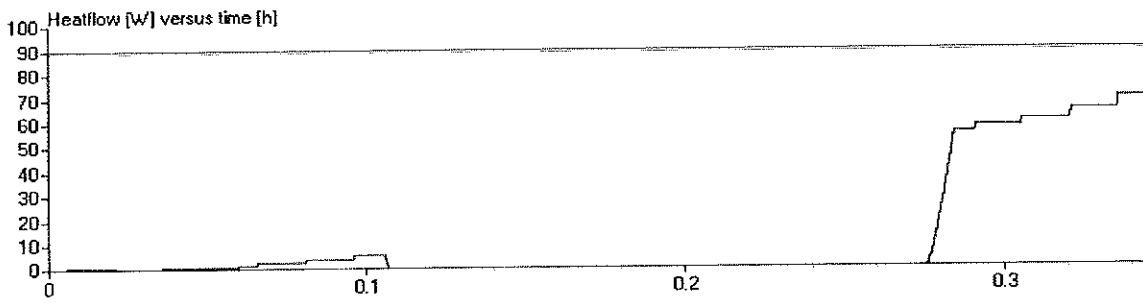
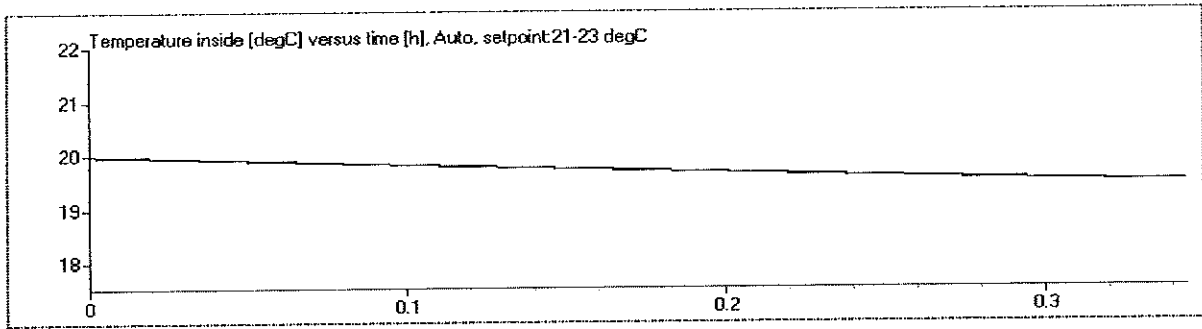


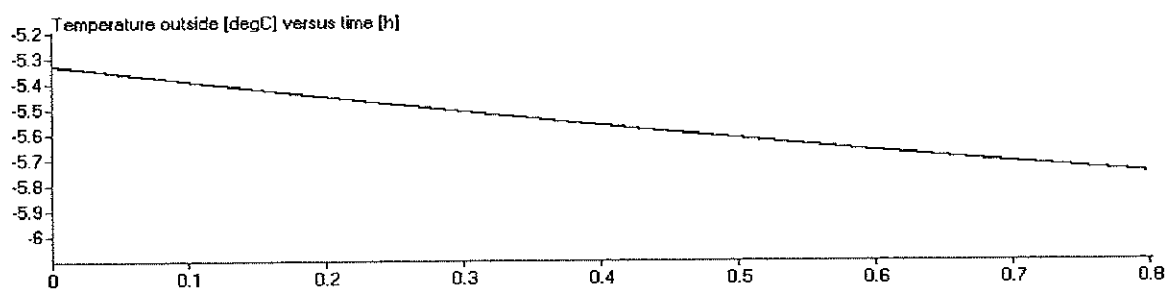
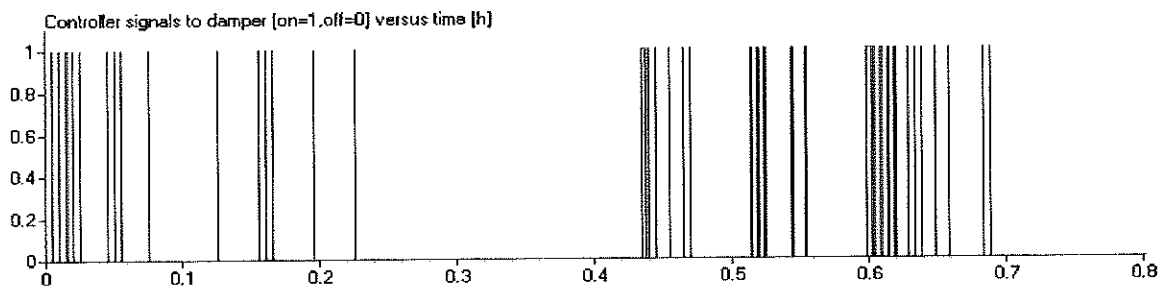
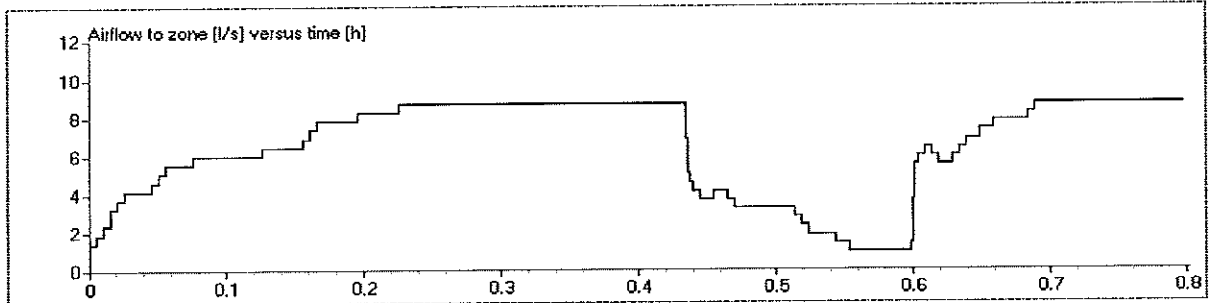
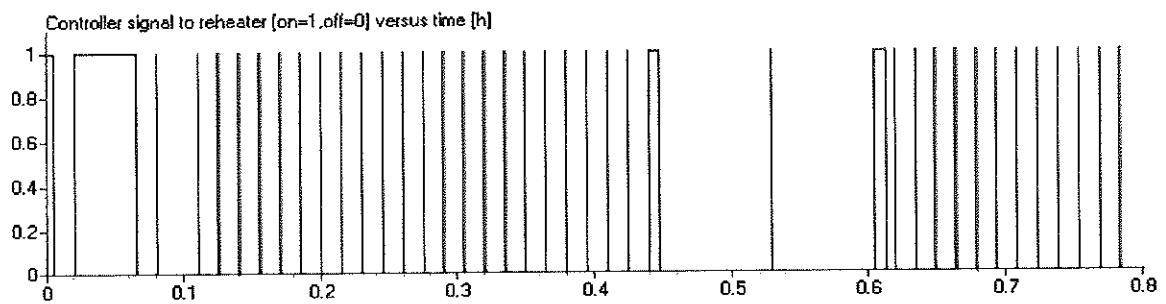
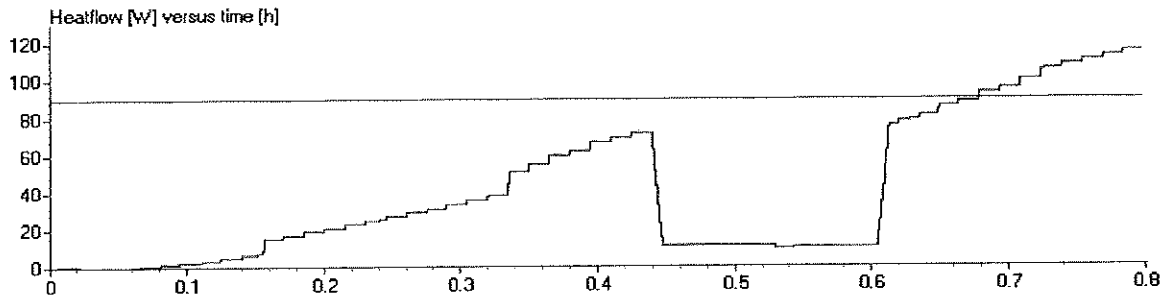
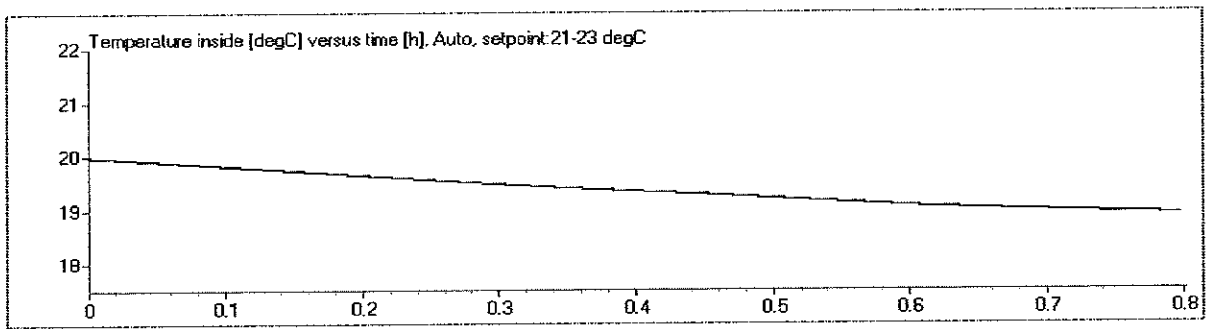


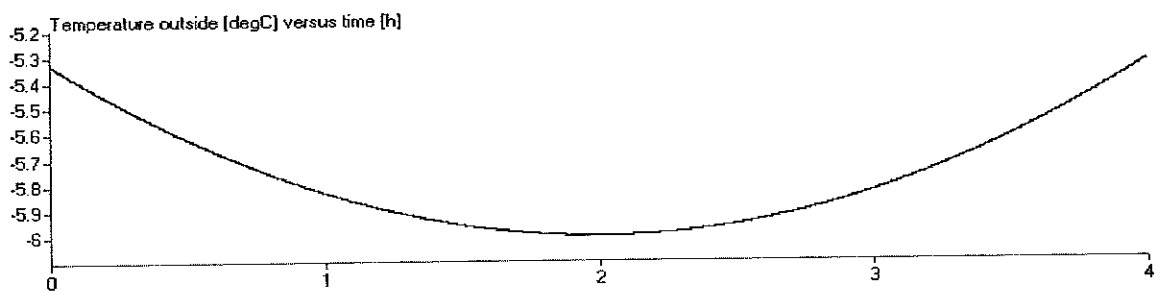
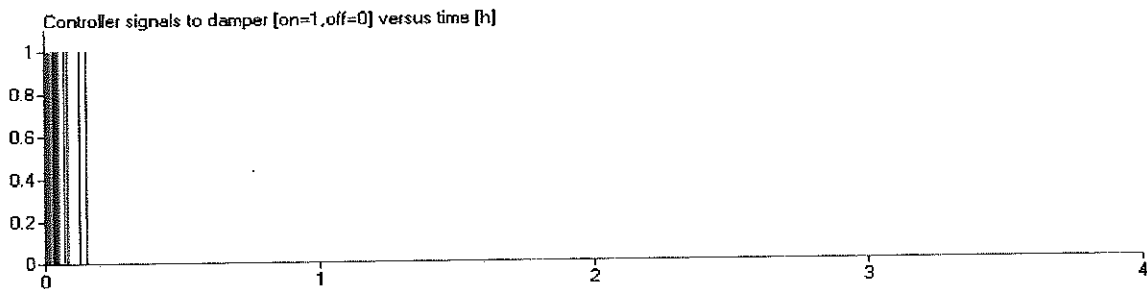
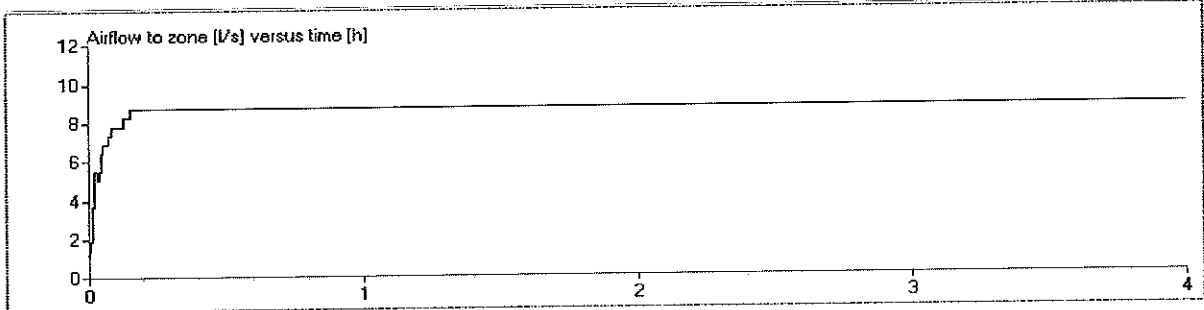
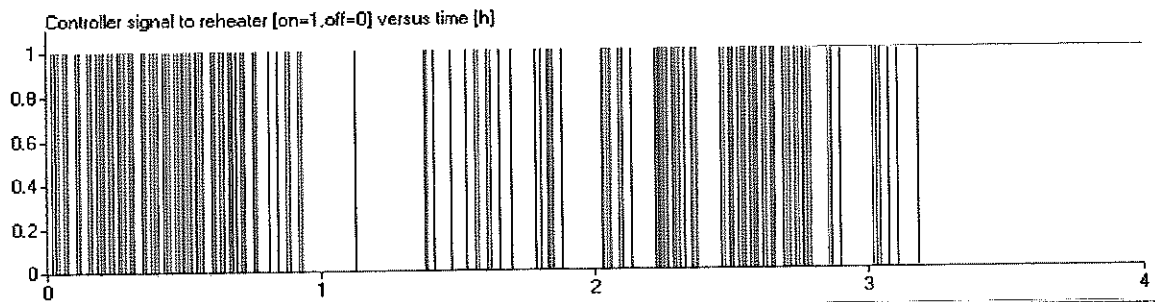
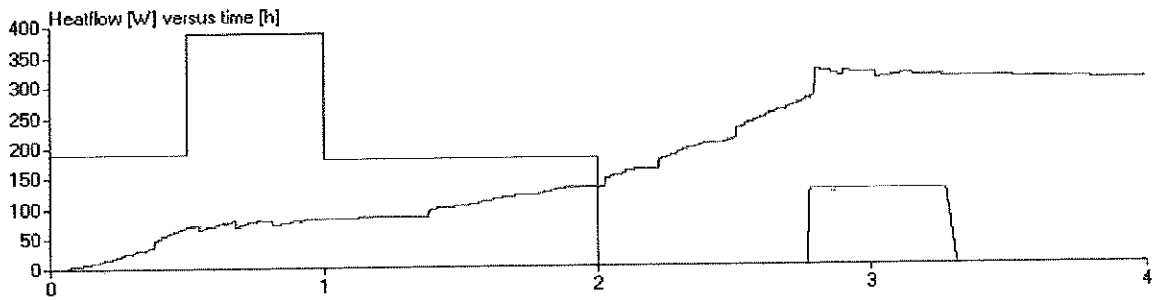
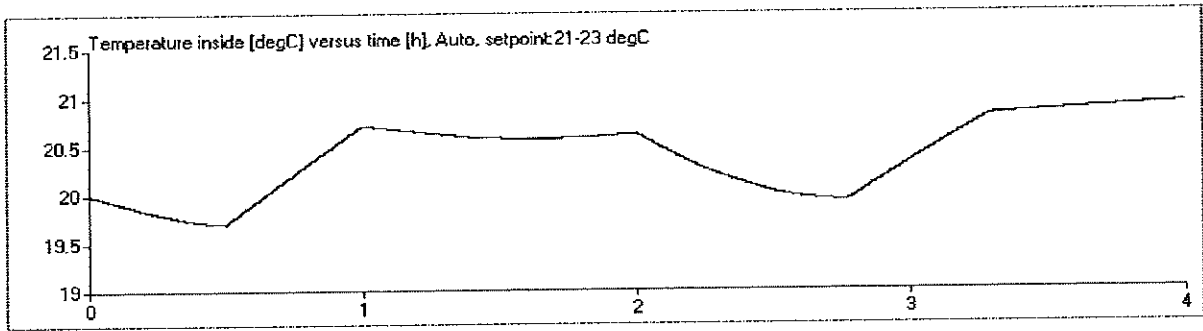


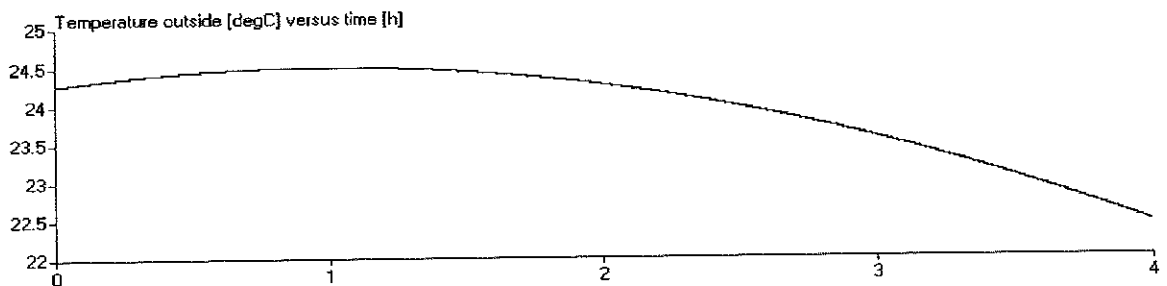
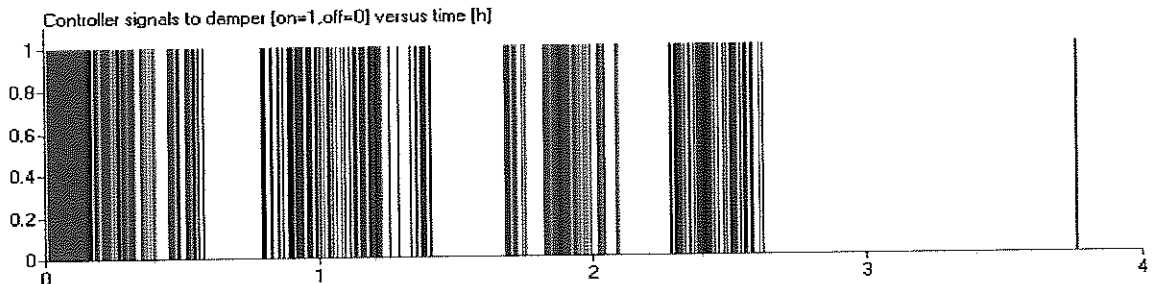
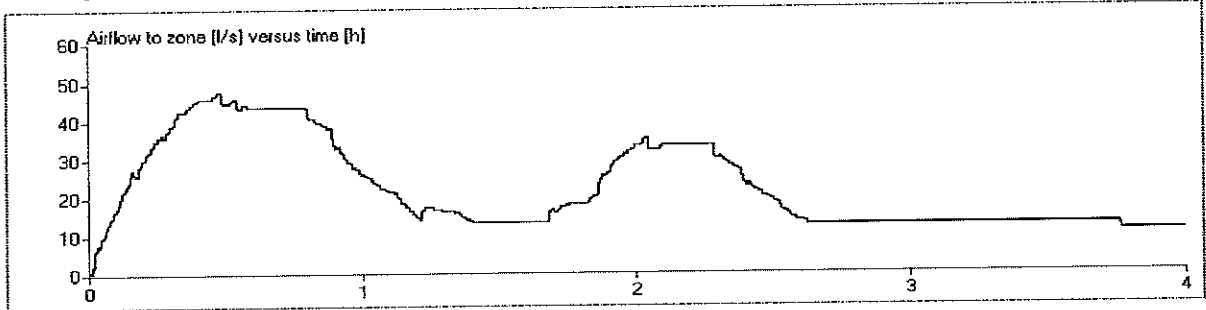
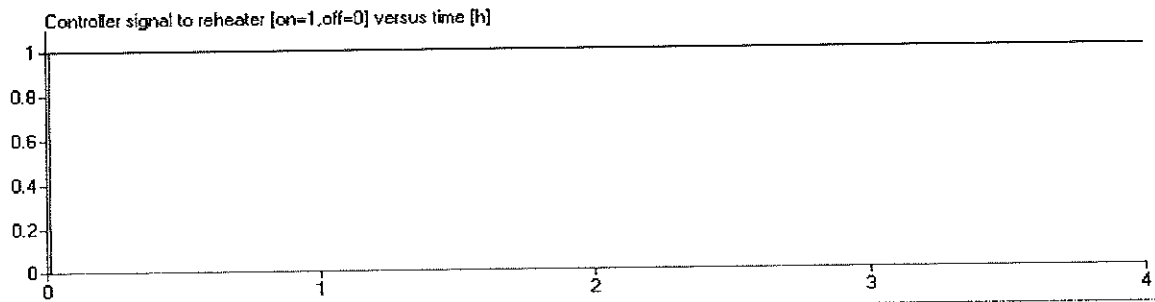
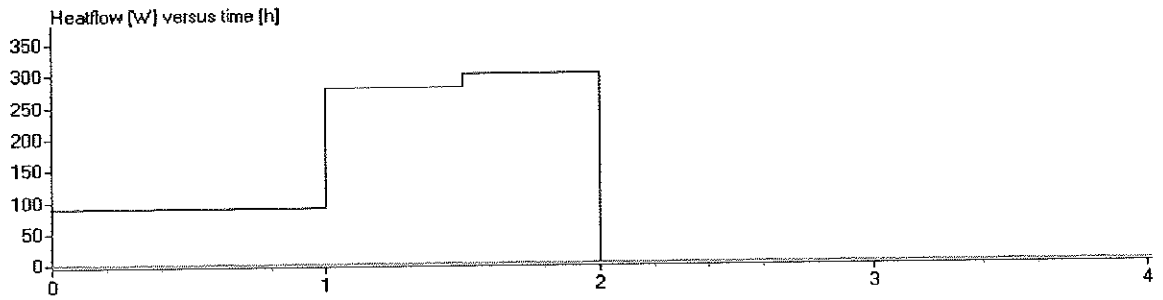
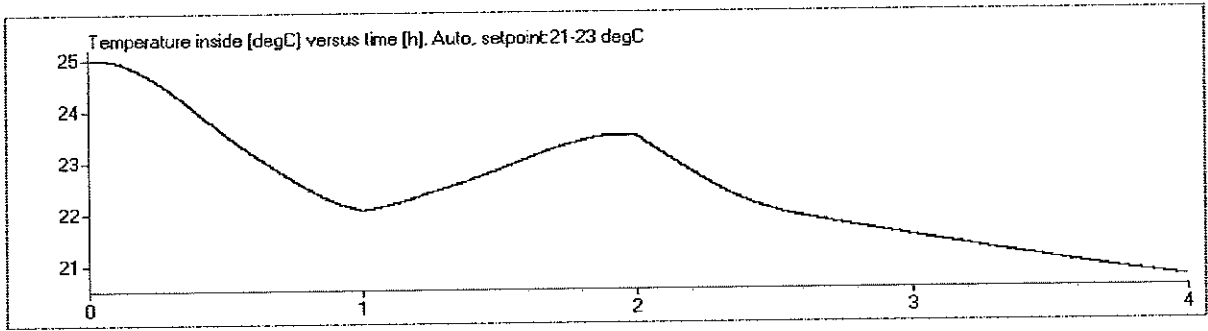


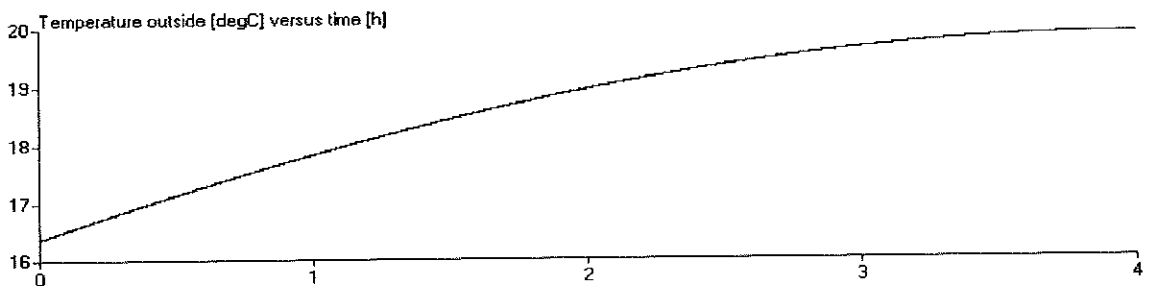
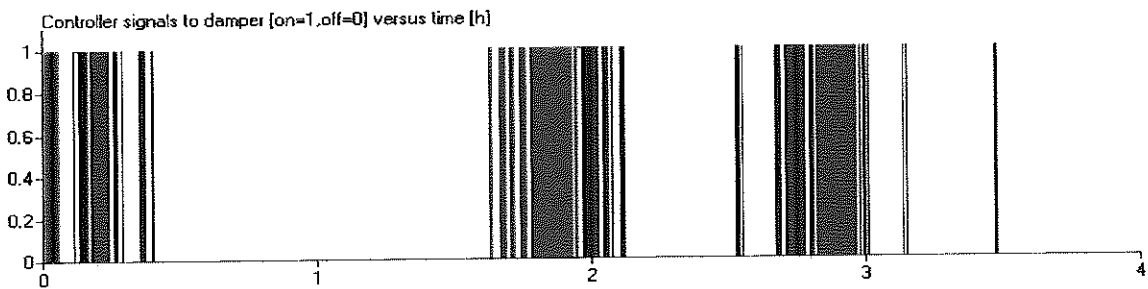
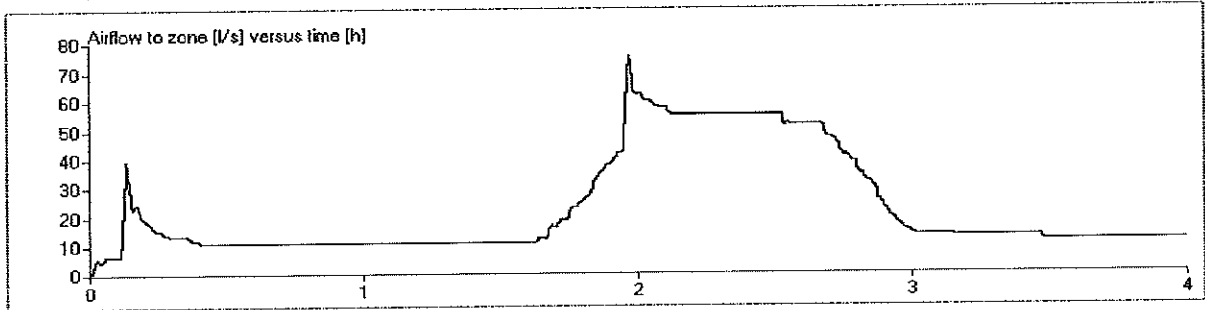
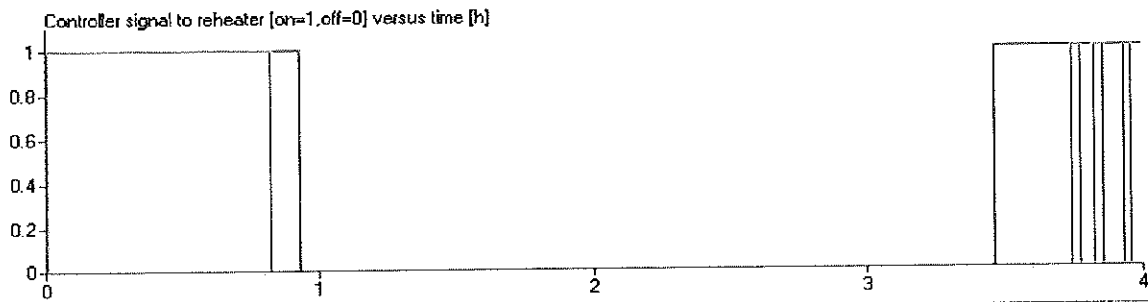
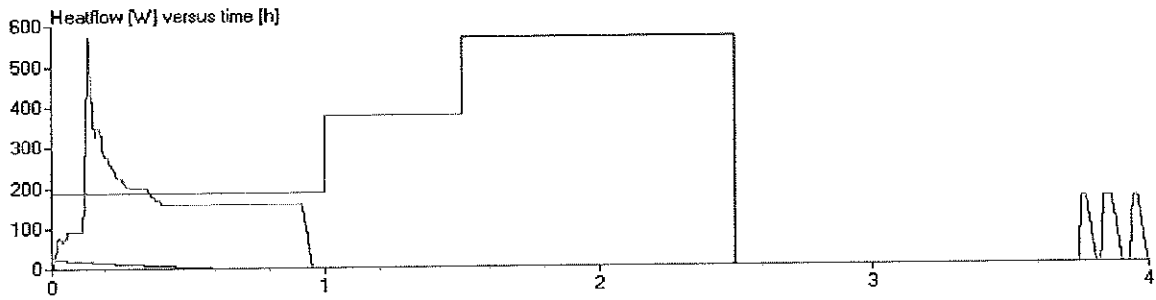
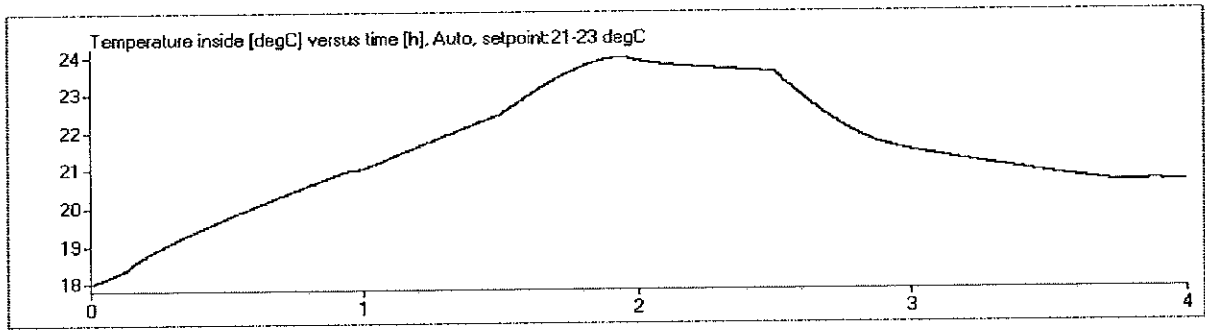


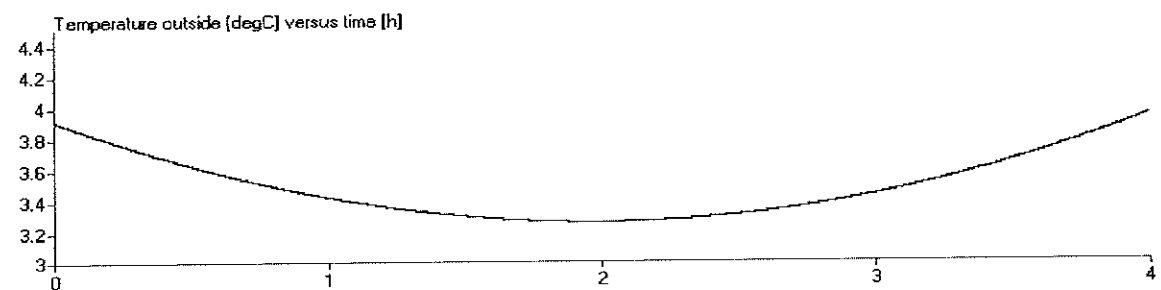
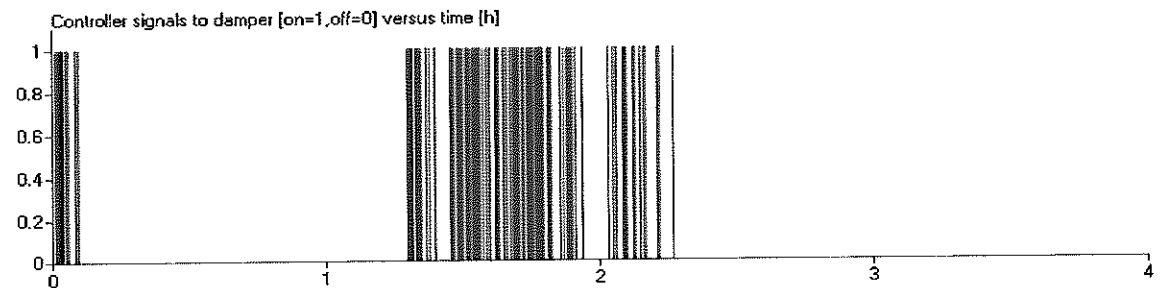
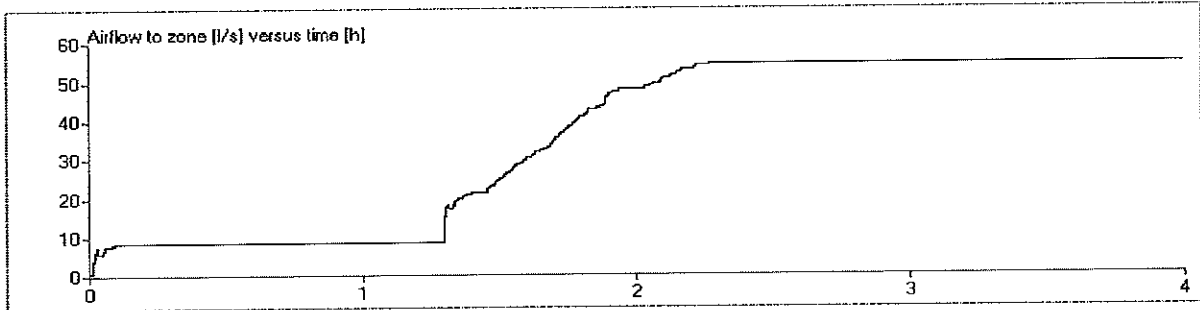
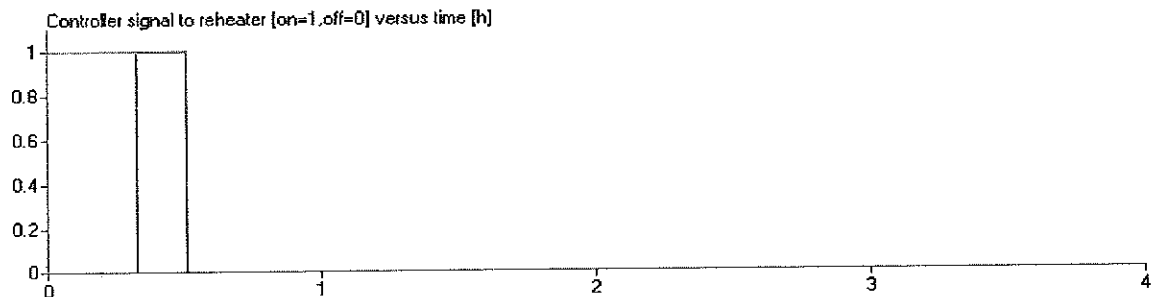
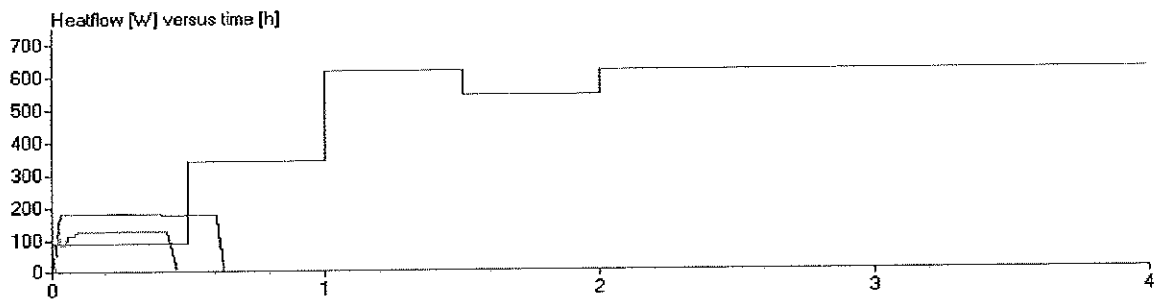
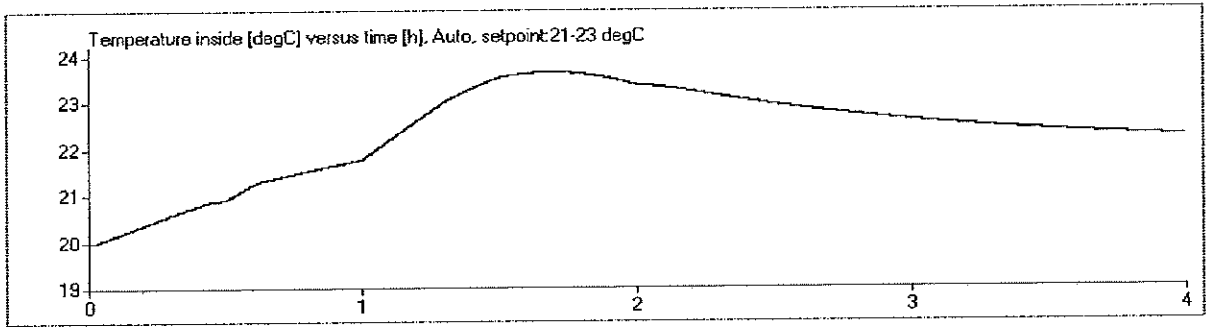


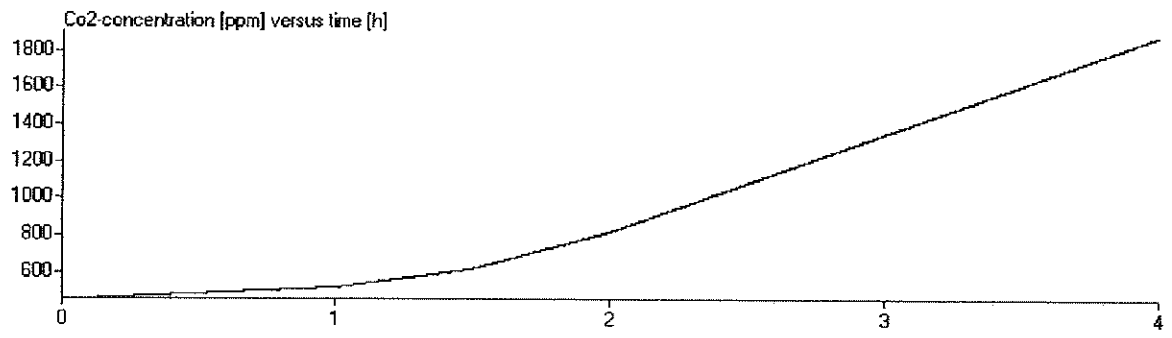
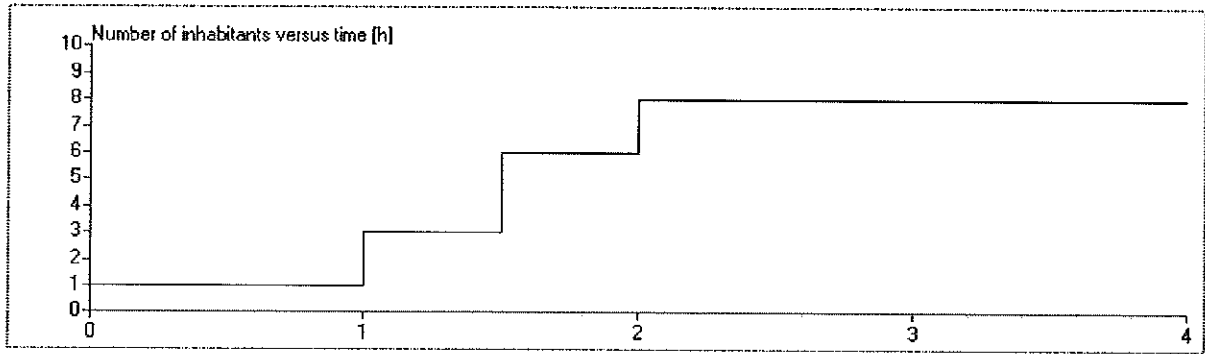












11 Sammanfattning

Testbänken är färdig att användas även om vissa kompletteringar kan behövas. Utgåendes från denna kan ganska lätt nya testbänkar sättas upp för andra regulatorer i serien.

En del buggar har hittats i Simnon, bland annat:

- Ett makro kan inte startas med valfria parametrar som utlovat
- Mark-kommandot till plotarna fungerar inte
- RPC-kommandon som krånglar ibland
- "Sorting error. Check handler or try increasing MAXSORT"-meddelandet som kommer upp och avbryter ens simulering eftre ett tag när man kör den via RPC-länken. Felet kunde inte förklaras av Simnon-folket. MAXSORT är tydligen en parameter i beräkningskärnan som man inte har tillgång till.
- Om man startar ett makro via RPC-länken, måste man direkt efter starten skriva till exempel "disp" i Simmons kommandofönster för att få igång realtidsritandet.

Dessa buggar kommer att rättas till så småningom av Simnon-leverantören. Till dess finns vid sidan om den riktiga testbänken en tillfällig som är baserad på makron istället för kommandon som fungerar alldeles utmärkt men begränsar ens valmöjligheter. Denna version tillåter användaren att välja standardfall, men tillåter inte övrig användarinblandning när simuleringen är igång. Dessutom kan man inte själv bestämma vad som ska sparas och ritas medan simuleringen pågår.

Intressant hade varit om man hade kunnat jämföra en verklig och en simulerad körning för att se vilka eventuella modeller som skulle behöva anpassas bättre. Som det är nu handlar mycket om praktisk erfarenhet: man tycker att det verkar stämma rätt så bra, har det framkommit efter samtal med medarbetare på företaget.

Jag har försökt att själv lista ut de typiska standardfallen. Vissa är mindre bra, såsom Auto 1, eftersom de inte sätter regulatorn på prov ordentligt, medan andra är högst intressanta, till exempel Auto 4, som under 4 timmar testar både uppvärmning, kylning och koldioxidreglering. Man kan tänka sig en utökning av dessa fall, helst i samarbete med testavdelningen så att riktiga fall tas fram. Ett par exempel är om man vill simulera ett typiskt konferensrum eller ett kontorsrum där man kör en 24-timmars simulering.

Jag har inte hunnit simulera regulatorn i nätverk, så att man i testbänken även implementerar LonWorks-beteende. Det kan istället bli föremål för eventuell vidareutveckling. Man kan implementera detta genom att till exempel lägga en extra sida till noteboken (X102ESTest) som hanterar detta. Man kan även lägga till extra knappar/editeringsrutor på de olika sidorna och knyta dem till olika variabler, såsom begynnelsevärdet för koldioxidhalten, U-värdet för fönster och väggar som nu ligger som fasta värden.

Om problemen med Simnon försvinner föreslår jag en vidareutveckling av testbänken, eftersom Simnon i övrigt är ett mycket bra simuleringsprogram som dessutom är väldigt prisvärt. Om man nöjer sig med att bara köra standardfallen, utan egendefinierade simuleringar, och kan tänka sig att definiera standardfallen som makron, föreslår jag också en vidareutveckling. Annars är min uppfattning att man ska avvakta tills buggarna har fixats.

Öppna filen Simnon.ini, som ligger under 'c:\windows', med till exempel Notepad. Kolla att raden "EnableRPC" är True och INTE False. Om så inte är fallet, ändra det. I slutet på filen ska följande rader läggas till:

```
[AddOns]
X100=X100.dll
RWIO=RWIO.dll
[Protocol]
Protocol=ncalrpc
Endpoint=simnonrpc
```

Steg 4: X102ESTest

Kopiera katalogen X102ESTest och placera den som underkatalog till Simnon-katalogen. Förutom programmet X102ESTest ligger här även Simnon-modellerna.

Kopiera filerna RWIO.dll och X100.dll till Simnon-katalogen.

Kopiera filen W32_8xx.dll från 'c:\analog' till Simnon-katalogen.

Nu är installationen fullständig.

Användarhandbok

Starta Simnon.

I menyn "File" välj "Open" för att ändra hemkatalog. Öppna Katalogen "X102ES" och tryck sedan OK.

Från menyn "Controller" tryck på "X102ES". Detta kommer att starta testprogrammet X102ESTest.

För att få igång kommunikationen mellan Simnon och testprogrammet tryck på knappen "Not connected to Simnon! Push to Connect!" under fliken "RunTimeSettings". OBS! Försöker man köra något annat kommando som försöker kommunicera med Simnon innan dess, leder detta till att programmet låser sig och måste startas om.

Om allt har gått rätt tillväga kan man nu starta olika simuleringar, antingen standardfall, som finns under fliken "StandardCases" eller egna. För att köra igång återgå till fliken "RunTimeSettings" och följ instruktionerna.

OBS! Efter att knappen "Start Simulation!" har tryckts ner, gå in i Simnon och skriv i kommandofönstret "disp", annars fungerar inte bland annat realtids-ritandet pga en bugg i Simnon.

När simuleringen är färdig kan man antingen starta en ny eller stänga av. Innan man stänger av ska man dock se till att koppla ner RPC-länken till Simnon med knappen "Disconnect from Simnon!" annars kommer inte Simnon att fungera som det ska.

Från reläbryggans plint nr.:	Till digitala AnalogDevices kopplingskortets plint nr:
1	Ej inkopplad
2	Ej inkopplad
3	Ej inkopplad
4	Ej inkopplad

Reläbryggan ska kopplas in till digitala signaler för att sköta om koldioxid- och närvarogivaren. För närvarande används en knapp till var och en och sköts manuellt.

HEAT 2:**Regulatorns SNVT:ar ska ställas in som följer:**

Index	Variabel	Värde
18	nviManOccCmd	Occupied
19	nviApplicMode	Heat
20	nviSpaceTemp	Invalid
21	nviSetpoint	Invalid
22	nviSetpntOffset	0
23	nviSpaceCO2	Invalid
24	nviEnergiHoldOff	0%, Off
25	nviManOverride	Off, 0 %, 0 l/s
26	nviEmergCmd	Normal
27	nviHeatSlaveR	0
28	nviOccSensor	Invalid
29	nviAuxInput	Off
30	nviHeatSlaveSec	Off
31	nviBoxFlowSlave	0
32	nciAppOptions	111101100000000
33	nciMinFlow	12
34	ncimaxFlow	55
35	nciBoxScale	5
36	nciBoxOffset	1
37	nciNomFlow	76
38	nciMinFlowHeat	10
39	nciMinFlowStand	0
40	nciSetpoints	oc23, oh21, sc25, sh19, uc28, uh16
41	nciSpaceTempDev	2
42	nciSpaceTempLow	10
43	nciVAVGain	25
44	nciVAVItime	900
45	nciGainDamp	1
46	nciDamperActStTime	165
47	nciGainHeat	25
48	nciItimeHeat	900
49	nciHeatActStTime	165
50	nciSpaceTempOfst	0
51	nciFlowOfstslave	0
52	nciCO2PerVolt	200
53	nciSpaceCO2Low	400
54	nciSpaceCO2High	1000
56	nciSndHrtBt	0
57	nciRcvHrtBt	0

HEAT 4:**Regulatorns SNVT:ar ska ställas in som följer:**

Index	Variabel	Värde
18	nviManOccCmd	Occupied
19	nviApplicMode	Heat
20	nviSpaceTemp	Invalid
21	nviSetpoint	Invalid
22	nviSetpntOffset	0
23	nviSpaceCO2	Invalid
24	nviEnergiHoldOff	0%, Off
25	nviManOverride	Off, 0 %, 0 l/s
26	nviEmergCmd	Normal
27	nviHeatSlaveR	0
28	nviOccSensor	Invalid
29	nviAuxInput	Off
30	nviHeatSlaveSec	Off
31	nviBoxFlowSlave	0
32	nciAppOptions	111101000000000
33	nciMinFlow	12
34	ncimaxFlow	55
35	nciBoxScale	5
36	nciBoxOffset	1
37	nciNomFlow	76
38	nciMinFlowHeat	10
39	nciMinFlowStand	0
40	nciSetpoints	oc23, oh21, sc25, sh19, uc28, uh16
41	nciSpaceTempDev	2
42	nciSpaceTempLow	10
43	nciVAVGain	25
44	nciVAVItime	900
45	nciGainDamp	1
46	nciDamperActStTime	165
47	nciGainHeat	25
48	nciItimeHeat	900
49	nciHeatActStTime	165
50	nciSpaceTempOfst	0
51	nciFlowOfstslave	0
52	nciCO2PerVolt	200
53	nciSpaceCO2Low	400
54	nciSpaceCO2High	1000
56	nciSndHrtBt	0
57	nciRevHrtBt	0

CO2:**Regulatorns SNVT:ar ska ställas in som följer:**

Index	Variabel	Värde
18	nviManOccCmd	Occupied
19	nviApplicMode	Cool
20	nviSpaceTemp	Invalid
21	nviSetpoint	Invalid
22	nviSetpntOffset	0
23	nviSpaceCO2	Invalid
24	nviEnergiHoldOff	0%, Off
25	nviManOverride	Off, 0 %, 0 l/s
26	nviEmergCmd	Normal
27	nviHeatSlaveR	0
28	nviOccSensor	Invalid
29	nviAuxInput	Off
30	nviHeatSlaveSec	Off
31	nviBoxFlowSlave	0
32	nciAppOptions	1111100000000000
33	nciMinFlow	12
34	ncimaxFlow	55
35	nciBoxScale	5
36	nciBoxOffset	1
37	nciNomFlow	76
38	nciMinFlowHeat	10
39	nciMinFlowStand	0
40	nciSetpoints	oc23, oh21, sc25, sh19, uc28, uh16
41	nciSpaceTempDev	2
42	nciSpaceTempLow	10
43	nciVAVGain	25
44	nciVAVItime	900
45	nciGainDamp	1
46	nciDamperActStTime	165
47	nciGainHeat	25
48	nciItimeHeat	900
49	nciHeatActStTime	165
50	nciSpaceTempOfst	0
51	nciFlowOfstslave	0
52	nciCO2PerVolt	200
53	nciSpaceCO2Low	400
54	nciSpaceCO2High	1000
56	nciSndHrtBt	0
57	nciRevHrtBt	0

OCCUPANCY:**Regulatorns SNVT:ar ska ställas in som följer:**

Index	Variabel	Värde
18	nviManOccCmd	Occupied
19	nviApplicMode	Auto
20	nviSpaceTemp	Invalid
21	nviSetpoint	Invalid
22	nviSetpntOffset	0
23	nviSpaceCO2	Invalid
24	nviEnergiHoldOff	0%, Off
25	nviManOverride	Off, 0 %, 0 l/s
26	nviEmergCmd	Normal
27	nviHeatSlaveR	0
28	nviOccSensor	Invalid
29	nviAuxInput	Off
30	nviHeatSlaveSec	Off
31	nviBoxFlowSlave	0
32	nciAppOptions	1111000000000000
33	nciMinFlow	12
34	ncimaxFlow	55
35	nciBoxScale	5
36	nciBoxOffset	1
37	nciNomFlow	76
38	nciMinFlowHeat	10
39	nciMinFlowStand	0
40	nciSetpoints	oc23, oh21, sc25, sh19, uc28, uh16
41	nciSpaceTempDev	2
42	nciSpaceTempLow	10
43	nciVAVGain	25
44	nciVAVtime	900
45	nciGainDamp	1
46	nciDamperActStTime	165
47	nciGainHeat	25
48	nciItimeHeat	900
49	nciHeatActStTime	165
50	nciSpaceTempOfst	0
51	nciFlowOfstslave	0
52	nciCO2PerVolt	200
53	nciSpaceCO2Low	400
54	nciSpaceCO2High	1000
56	nciSndHrtBt	0
57	nciRevHrtBt	0

AUTO 2:**Regulatorns SNVT:ar ska ställas in som följer:**

Index	Variabel	Värde
18	nviManOccCmd	Occupied
19	nviApplicMode	Auto
20	nviSpaceTemp	Invalid
21	nviSetpoint	Invalid
22	nviSetpntOffset	0
23	nviSpaceCO2	Invalid
24	nviEnergiHoldOff	0%, Off
25	nviManOverride	Off, 0 %, 0 l/s
26	nviEmergCmd	Normal
27	nviHeatSlaveR	0
28	nviOccSensor	Invalid
29	nviAuxInput	Off
30	nviHeatSlaveSec	Off
31	nviBoxFlowSlave	0
32	nciAppOptions	1111000000000000
33	nciMinFlow	12
34	ncimaxFlow	55
35	nciBoxScale	5
36	nciBoxOffset	1
37	nciNomFlow	76
38	nciMinFlowHeat	10
39	nciMinFlowStand	0
40	nciSetpoints	oc23, oh21, sc25, sh19, uc28, uh16
41	nciSpaceTempDev	2
42	nciSpaceTempLow	10
43	nciVAVGain	25
44	nciVAVItime	900
45	nciGainDamp	1
46	nciDamperActStTime	165
47	nciGainHeat	25
48	nciItimeHeat	900
49	nciHeatActStTime	165
50	nciSpaceTempOfst	0
51	nciFlowOfstslave	0
52	nciCO2PerVolt	200
53	nciSpaceCO2Low	400
54	nciSpaceCO2High	1000
56	nciSndHrtBt	0
57	nciRevHrtBt	0

AUTO 4:**Regulatorns SNVT:ar ska ställas in som följer:**

Index	Variabel	Värde
18	nviManOccCmd	Occupied
19	nviApplicMode	Auto
20	nviSpaceTemp	Invalid
21	nviSetpoint	Invalid
22	nviSetpntOffset	0
23	nviSpaceCO2	Invalid
24	nviEnergiHoldOff	0%, Off
25	nviManOverride	Off, 0 %, 0 l/s
26	nviEmergCmd	Normal
27	nviHeatSlaveR	0
28	nviOccSensor	Invalid
29	nviAuxInput	Off
30	nviHeatSlaveSec	Off
31	nviBoxFlowSlave	0
32	nciAppOptions	111101000000000
33	nciMinFlow	12
34	ncimaxFlow	55
35	nciBoxScale	5
36	nciBoxOffset	1
37	nciNomFlow	76
38	nciMinFlowHeat	10
39	nciMinFlowStand	0
40	nciSetpoints	oc23, oh21, sc25, sh19, uc28, uh16
41	nciSpaceTempDev	2
42	nciSpaceTempLow	10
43	nciVAVGain	25
44	nciVAVItime	900
45	nciGainDamp	1
46	nciDamperActStTime	165
47	nciGainHeat	25
48	nciItimeHeat	900
49	nciHeatActStTime	165
50	nciSpaceTempOfst	0
51	nciFlowOfstslave	0
52	nciCO2PerVolt	200
53	nciSpaceCO2Low	400
54	nciSpaceCO2High	1000
56	nciSndHrtBt	0
57	nciRcvHrtBt	0