

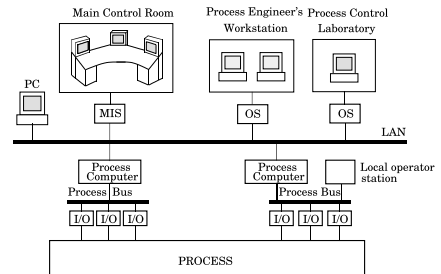
## Processreglering Föreläsning X

- Datorstyrning
- Logik
- Sekvensstyrning

Läsanvisning: *Process Control*: 9.1–9.8

1

## Datorstyrning



Kemiska processer styrs med datorer

- Hur blir PID-regulatorn ett datorprogram?
- Hur startar och stoppar man en process?

2

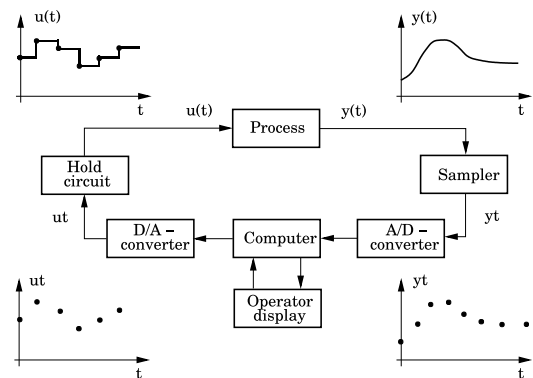
## Datorns egenskaper

- Kan bara göra en sak i taget, men snabbt
- Numera hög tillförlitlighet på hårdvara
- Svårare att skriva tillförlitlig mjukvara, speciellt för parallella uppgifter
- Minne, snabbhet växer

Hur kan datorer användas för reglering?

3

## Samplade regelsystem



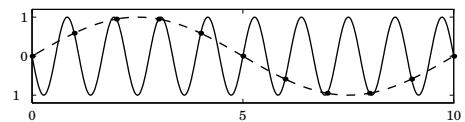
4

## Samplade regelsystem

- Blandning av kontinuerlig och diskret tid – svårt att analysera
  - Förenkling: titta bara i samplingstidpunkterna
- Potentiella problem
  - Vad händer mellan samplingarna?
  - Hur väljs samplingsintervall?
  - Inverkan av kvantisering
  - Hur beskriver man samplade system?

5

## Förlorad information genom sampling

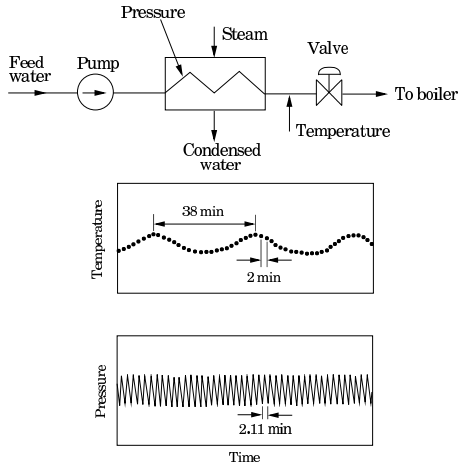


**Sampling:**

- innebär att man mäter i diskreta punkter med ett givet tidsintervall, dvs *samplingsintervall*,  $h$ .
- **Aliasing** kallas det då högre frekvenser tolkas som lägre frekvenser
- *Samplingsteoremet* säger att minst **två sampel per period** krävs för att inte få aliasing

6

### Exempel på aliasing



7

### Matematiska beskrivningar av system

- Tidskontinuerliga system:
  - Differentialekvationer, t.ex.  $T \frac{dy}{dt} + y = Ku$
  - Laplace transform, t.ex.  $Y(s) = \frac{K}{1+Ts} U(s) = G(s)U(s)$
- Samplade (tidsdiskreta) system:

#### - Differensekvationer

$$y(kh + h) + ay(kh) = bu(kh)$$

#### - Skiftoperator: $qy(kh) = y(kh + h)$

$$y(kh) = \frac{b}{q+a} u(kh) = H(q)u(kh)$$

8

### Approximationer av tidskontinuerliga system

Approximera derivatorna med differenser (fungerar för korta samplingsintervall)

- Framåtdifferens (Euler)

$$\frac{dy}{dt} \approx \frac{y(t+h) - y(t)}{h}$$

- Bakåtdifferens

$$\frac{dy}{dt} \approx \frac{y(t) - y(t-h)}{h}$$

- (dessa approximationer kallas *diskretisering*)

OBS! Det finns fler diskretiseringsmetoder (jämför med numerisk integration)

9

### Exempel

Diskretisera det kontinuerliga systemet  $\frac{dy(t)}{dt} = -3y(t) + 2u(t)$

- Framåtdifferens:

$$\frac{dy}{dt} \approx \frac{y(kh+h) - y(kh)}{h} = -3y(kh) + 2u(kh)$$

$$y(kh+h) = (1-3h)y(kh) + 2hu(kh)$$

- Bakåtdifferens:

$$\frac{dy}{dt} \approx \frac{y(kh) - y(kh-h)}{h} = -3y(kh) + 2u(kh)$$

$$y(kh) = \frac{1}{1+3h} y(kh-h) + \frac{2h}{1+3h} u(kh)$$

10

### Analys av stabilitet

Stabilitet för en skalär differensekvation:

$$y(kh+h) = ay(kh) + bu(kh)$$

Antag  $u(kh) = 0$ .  $y(kh) = a^k y(0)$

- $y(\infty) = 0$  om  $|a| < 1$
- $|y(\infty)| = \infty$  om  $|a| > 1$
- (poler innanför enhetscirkeln medför asymptotisk stabilitet)

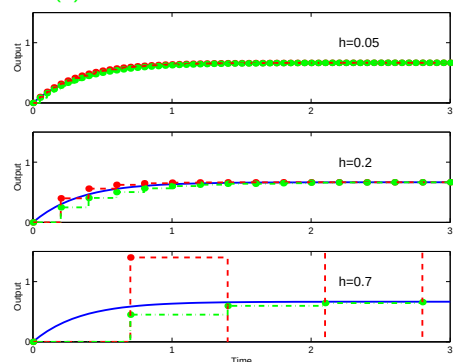
**Exemplet:** Stabilitetskrav

- Framåtdifferens  $|1-3h| < 1 \Rightarrow 0 < h < 2/3$
- Bakåtdifferens  $|\frac{1}{1+3h}| < 1 \Rightarrow h > 0$

11

### Simulering

$G(s) = \frac{2}{s+3}$ . Exakt lösning (—) Framåtdifferens (Euler) (- -) Bakåtdifferens (-.)



12

## Diskretisering av PI-regulatorn

### 1. Proportionaldel (ingen approximation behövs)

$$P(kh) = K e(kh)$$

### 2. Integraldel (framåtdifferens)

$$I(t) = \frac{K}{T_i} \int_0^t e(\tau) d\tau$$

$$\frac{dI(t)}{dt} = \frac{K}{T_i} e(t)$$

$$\frac{I(kh+h) - I(kh)}{h} = \frac{K}{T_i} e(kh)$$

$$I(kh+h) = I(kh) + \frac{Kh}{T_i} e(kh)$$

13

Intelgraldel med uppvridningsskydd (förhindrar att I-delen fortsätter att växa vid mättad styrsignal):

$$I(kh+h) = I(kh) + \frac{Kh}{T_i} e(kh) + \frac{h}{T_r} (u(kh) - v(kh))$$

- $v(kh)$  är den signal regulatorn *vill* ställa ut
- $u(kh)$  är den mättade signal regulatorn *kan* ställa ut
- $T_r$  är *tracking-konstanten*

(Mer om detta i föreläsning 9.)

14

## Implementering av PI-regulatorn – pseudokod

```

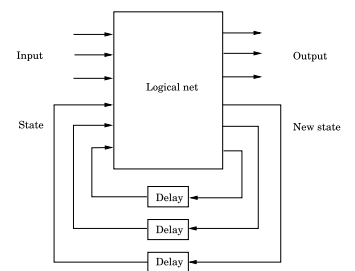
LOOP
  WaitForClockTick();
  r = ADIn(1);
  y = ADIn(2);
  P = K*(r-y)
  v = P + I;
  IF v < umin
    u = umin;
  ELSEIF v > umax
    u = umax;
  ELSE
    u = v;
  END
  DAOut(1,u);
  I = I + K*h/Ti*(r-y) + h/Tr*(u-v);
END

```

15

## Logik och diskret styrning

- Diskreta signaler
  - Mät signaler: *sant* eller *falskt*
  - Styr signaler: **på** eller **av**
- Logiknät
  - Statiskt nät
  - Ex: Alarm
  - Kallas *förreglingar*
- Sekvensnät
  - Dynamiskt nät
  - Ex: Uppstart/stopp, satsvisa förlopp



16

## George Boole (1815–1864)



Boole approached logic in a new way reducing it to a simple algebra, incorporating logic into mathematics. He also worked on differential equations, the calculus of finite differences and general methods in probability.

*An investigation into the Laws of Thought, on Which are founded the Mathematical Theories of Logic and Probabilities* (1854)

17

## Operationer och symboler

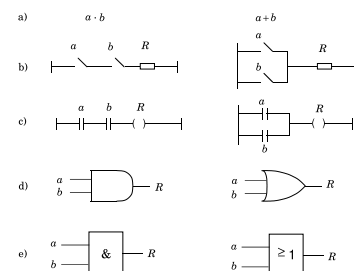
Tre operationer:

and:  $a \cdot b$   $a$  and  $b$   $a \wedge b$

or:  $a + b$   $a$  or  $b$   $a \vee b$

not:  $\bar{a}$  not  $a$   $\neg a$

Symboler: (svensk och amerikansk standard)



18

## Logiklagar

Boolesk algebra:

- Ex:  $1 + a = 1$  och  $0 + a = a$
- Ex:  $1 \cdot a = a$  och  $0 \cdot a = 0$
- Ex:  $a + \bar{a} = 1$  och  $a \cdot \bar{a} = 0$

Logiska lagar:

- Kommutativ  
 $a \cdot b = b \cdot a$ ,  $a + b = b + a$
- Associativ  
 $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ ,  $a + (b + c) = (a + b) + c$
- Distributiv  
 $a \cdot (b + c) = a \cdot b + a \cdot c$
- de Morgans lag  
 $a + \bar{b} = \bar{a} \cdot b$ ,  $a \cdot \bar{b} = \bar{a} + b$

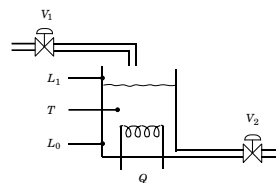
19

## Exempel

**Alarm för en satsreaktor:**

Ge alarm om temperaturen i tanken är för hög,  $T$ , samt kylventilen är stängd,  $Q$ , eller då temperaturen är hög och inflödesventilen är öppen,  $V_1$ .

Sanningstabell:



| $T$ | $Q$ | $V_1$ | $y = \text{alarm}$ |
|-----|-----|-------|--------------------|
| 0   | 0   | 0     | 0                  |
| 1   | 0   | 0     | 1                  |
| 0   | 1   | 0     | 0                  |
| 1   | 1   | 0     | 0                  |
| 0   | 0   | 1     | 0                  |
| 1   | 0   | 1     | 1                  |
| 0   | 1   | 1     | 0                  |
| 1   | 1   | 1     | 1                  |

20

## Sekvensstyrning

Sekvensnät

- har minne (eller tillstånd), signalernas ordningsföljd är väsentlig
- Olika beskrivningar
  - Finite state machine (automatateori)
  - Petrinät
  - GRAFCET är ett slags Petrinät
- Sekvenser i kemiteknik
  - Uppstart och stopp
  - Batch-recept och lab-procedurer
  - Vissa analysinstrument

21

## GRAFCET – Steg och övergångar

Steg:

- Aktiv respektive inaktiv



S1.x = 1 when active

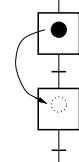
S1.T = number of time units since the step last became active



Initial step

Övergång:

avfyras då föregående steg är aktivt samt övergångens villkor är eller blir uppfyllt.



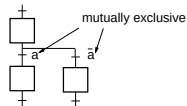
condition true and/or event occurred + previous step active

22

## GRAFCET – Kontrollstrukturer

- Alternativa vägar:

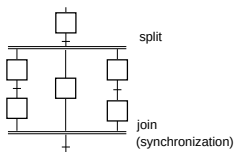
1. Förgrening (ömsesidigt uteslutande)



2. Repetition

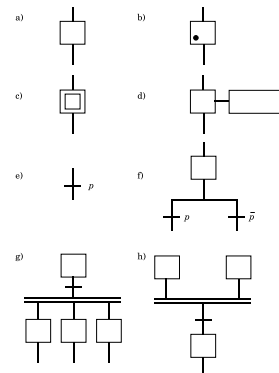


- Parallella vägar med synkroniserad avslutning



23

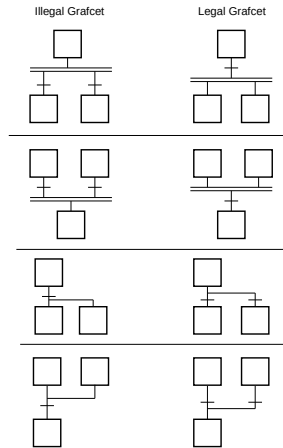
## GRAFCET – Grundläggande symboler



- (a) Inaktivt steg, (b) Aktivt steg, (c) Initialsteg, (d) Steg med åtgärd, (e) Övergång, (f) Förgrening med ömsesidigt uteslutande alternativ, (g) Förgrening i parallella vägar, (h) Synkronisering

24

## GRAFCET – Några exempel



25

## GRAFCET – Exekveringsprinciper

GRAFCETs exekveringsregler:

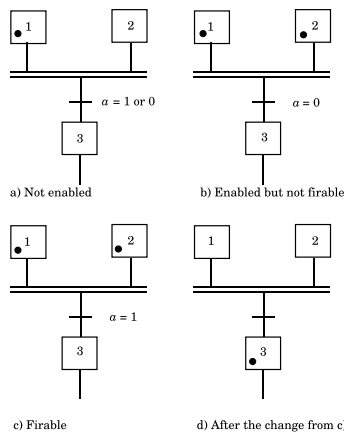
- Initialsteget är aktivt när sekvensnätet är initierat.
- En övergång är avfyringsbar om:
  - alla steg före övergången är aktiva
  - övergångsvillkoret är sant

En avfyringsbar övergång måste avfyras

- alla stegen före övergången inaktiveras, och alla följande steg aktiveras, när en övergång avfyras.
- alla avfyringsbara övergångar avfyras samtidigt
- när ett steg både måste inaktiveras och aktiveras så kvarstår steget som aktivt utan avbrott

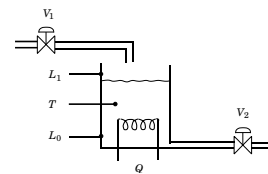
26

## GRAFCET – Exempel på avfyring



27

## Exempel: specifikationer

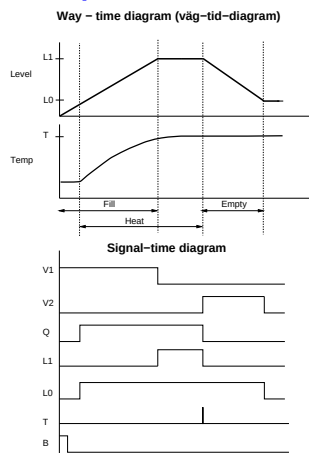


Verbal beskrivning:

1. Starta sekvensen med att trycka på knappen  $B$
2. Öppna ventil  $V_1$  för påfyllning av vatten till övre nivå  $L_1$
3. Värm vattnet tills temperaturen är högre än  $T$ . Värmning kan starta så fort det finns vatten över nivå  $L_0$
4. Töm tanken genom att öppna ventil  $V_2$  tills nivån når  $L_0$ .
5. Stäng ventilerna och gå till 1) för att vänta på en ny startorder.

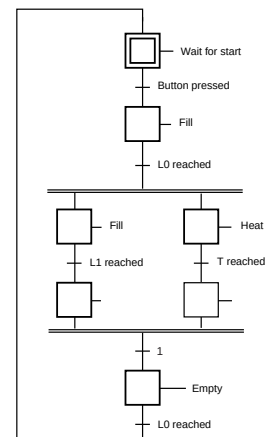
28

## Exempel: tidsfunktioner



29

## Exempel: Sekvensnät

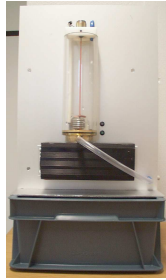


30

## Laboration X

Sekventiellt styrprogram för en satsreaktor

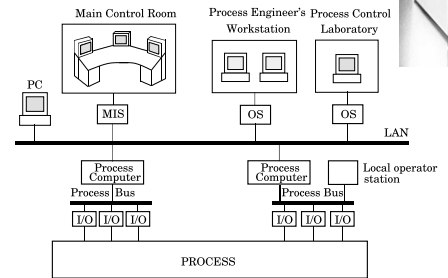
1. Sekvensstyrning och GRAFCET
2. JGrafchart (Java-baserad implementation av GRAFCET)
3. Digital implementering av PI-regulator för temperatur
4. Test genom simulering
5. Test på vattenvärmningsprocess



31

## Industriella styrsystem

- Digitala regulatorer
- PLC (programable logical controller) för logik och sekvensstyrning
- Processdatorsystem



32