

ADAPTIV PREDIKTION

BJÖRN WITTENMARK

Rapport RE - 4      aug. 1966

ADAPTIV PREDIKTION

Examensarbete i regleringsteknik

Björn Wittenmark

## ADAPTIV PREDIKTION

### Kap

1. Inledning
    - 1.1 Formulering av problemet
    - 1.2 Några kommentarer till problemet
  2. Konstruktion av prediktor för en process  $y(t) + a_1 y(t-1) + \dots + a_n y(t-n) = e(t)$  då  $a_i$   $i = 1 \dots n$  är kända samt  $E(e(t)) = 0$  och  $E(e(t)^2) = \sigma^2$ 
    - 2.1 Enstegsprediktor
    - 2.2 K-stegsprediktor. Linjärprediktor
  3. Uppskattning av parametrarna  $a_1, a_2, \dots, a_n$  i processen  $y(t) + a_1 y(t-1) + \dots + a_n y(t-n) = e(t)$  då utdata är givna
  4. Adaptiv prediktion och parameteruppskattning
  5. Kodning
  6. Redovisning av exempel
  7. Sammanfattning och slutsatser
- Appendix A - Diagram
- Appendix B - Datamaskinutskriften av stegsvaret

## 1. INLEDNING

### 1.1 Formulering av problemet

Utgångspunkten för detta examensarbete är följande:

Antag att en stokastisk process  $\{y(t)\}$  genereras av ekvationen

$$y(t) + a_1 y(t - 1) + a_2 y(t - 2) + \dots + a_n y(t - n) = e(t) \quad (1.1)$$

där  $\{e(t)\}$  är en svit okorrelerade lika fördelade stokastiska variabler.

I anslutning till detta antagande skall följande tre problem studeras:

- (i)
  - a) Konstruera en enstegsprediktor för en process enligt (1.1) då parametrarna  $a_1, a_2, \dots, a_n$  är kända.
  - b) Konstruera en flerstegsprediktor under samma förutsättningar.
- (ii) Uppskatta parametrarna  $a_1, a_2, \dots, a_n$  i processen (1.1) då ordningstalet  $n$  och en svit värden  $y(1), y(2) \dots y(N)$  är kända.
- (iii) Angiv ett adaptivt förfarande att samtidigt prediktera  $y(t)$  ett steg framåt och att uppskatta parametrarna  $a_1, a_2, \dots, a_n$  i processen (1.1).

### 1.2 Några kommentarer till problemet

Man kan välja flera olika vägar att angripa problemet. Det är framför allt vilka förutsättningar som skall göras angående de stokastiska variablerna  $\{e(t)\}$ .

Man kan anta

1.  $\{e(t)\}$  är normalfördelade.
2.  $\{e(t)\}$  har ett givet första och andra moment.

Det första antagandet får som följd att den "bästa" prediktorn (i någon mening som specificeras längre fram) blir linjär. Antagandet med normalfördelning för med sig att det går att använda maximum-likelihood (ML) uppskattning. Jag kommer att använda mig av denna, ty ML-uppskattningen ger också ett mått på hur noggrann min parameteruppskattning är.

Om man använder det andra antagandet om  $\{e(t)\}$  så för detta med sig att man måste postulera en linjär prediktor.

Vad är villkoret för att processen  $\{y(t)\}$  är stationär?

Med hjälp av z-operatorm definierad genom

$$z y(t) = y(t - 1)$$

kan ekvationen (1) skrivas

$$(1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}) y(t) = e(t)$$

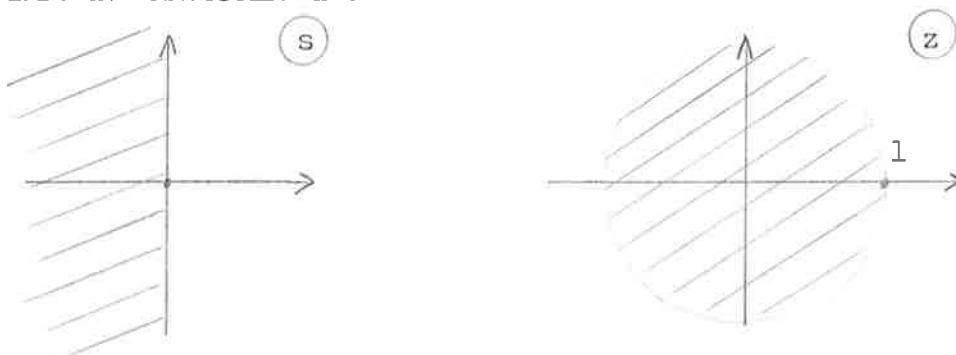
eller

$$A(z^{-1}) y(t) = e(t)$$

Vi kan nu se processen  $y(t)$  som utsignalen från ett filter på vars ingång vi har bruset  $e(t)$ . I z-transformeln har då filtret överföringsfunktionen

$$\frac{1}{A(z^{-1})}$$

När det gäller kontinuerliga system sysslar man med Laplace-transformation och då skall rötterna till överföringsfunktionens karakteristiska ekvation ligga strängt inne i vänstra halvplanet för att systemet skall vara asymptotiskt stabilt. Nu gäller följande samband mellan  $z$  och  $s$  (argumentet i Laplacetransformationen)  $z = e^{sT}$ . Dvs vänstra  $s$ -halvplanet avbildas på det inre av enhetscirkeln.



För att processen skall vara asymptotiskt stabil fordras att rötterna till  $A(z^{-1})$  skall ligga strängt innanför enhetscirkeln.

Meningen med att titta på detta problem är att kunna använda lösningen i ett adaptivt system, dvs i ett system där man ändrar styrlagen allteftersom nya data strömmar in och allteftersom processen ändrar sig.

Min avsikt med detta arbete är dock bara att lösa prediktions- och parameteruppskattningsproblemet och testa lösningar på ett antal exempel för att se hur pass bra det går att identifiera olika processer med autoregressioner dvs då prediktorn är ett glidande medelvärde av ett antal föregående observationer.

2. KONSTRUKTION AV PREDIKTOR FÖR EN PROCESS  $y(t) + a_1 y(t - 1) + \dots + a_n y(t - n) = e(t)$  DÅ  $a_i$ ,  $i = 1 \dots n$  ÄR KÄNDA SAMT  $E(e(t)) = 0$  OCH  $E(e(t)^2) = \sigma^2$ .

### 2.1 Enstegsprediktor.

Problemet är nu att göra en prediktion av  $y(t + 1)$  då  $y(t)$ ,  $y(t - 1) \dots$  är kända, så att variansen av prediktionsfelet blir så liten som möjligt.

Prediktorn betecknas med  $\hat{y}(t + 1|t)$ . I formelspråk blir då problemet att söka minimum av  $E(\hat{y}(t + 1|t) - y(t + 1))^2$ .

Ansätt en olinjär prediktor

$$\begin{aligned} \hat{y}(t + 1|t) = & \alpha_1 y(t) + \alpha_2 y(t - 1) + \alpha_3 y(t - 2) + \dots + \\ & + \alpha_{n+1} y(t - n) + \alpha_{n+2} y(t - n - 1) + \dots + \\ & + \beta_1 y(t) y(t) + \beta_2 y(t) y(t - 1) + \dots \end{aligned}$$

Felet  $F$  blir då

$$\begin{aligned} F = \hat{y}(t + 1|t) - y(t + 1) = & \alpha_1 y(t) + \dots + \beta_1 y(t)^2 + \dots + \\ & + a_1 y(t) + a_2 y(t - 1) + \dots + \\ & + a_n y(t - n + 1) - e(t + 1) = \\ = & K - e(t + 1) \end{aligned}$$

Om nu  $e(t)$ , som är okorrelerade, är normalfördelade så är de även oberoende. Således är  $e(t + 1)$  oberoende av  $e(t)$ ,  $e(t - 1)$ ,  $\dots$  och då även av  $y(t)$ ,  $y(t - 1) \dots$

Detta medför att

$$E(F^2) = E(K^2) + E(e(t+1)^2)$$

De blandade termerna försvinner ty  $E(e(t+1)) = 0$ .

Men om  $e(t)$  ej är normalfördelade utan bara har känt första och andra moment och är okorrelerade så måste man förutsätta en linjär prediktor för att de blandade termerna skall försvinna. De blandade termerna försvinner nu ty  $e(t+1)$  är okorrelerad med  $e(t)$ ,  $e(t-1)$  och således även med  $y(t)$ ,  $y(t-1)$ ....

Detta i sin tur medför att  $E(e(t+1)y(t)) = E(e(t+1)) \cdot E(y(t)) = 0$ .

Således är även i detta fall

$$E(F^2) = E(K^2) + E(e(t+1)^2)$$

Minimum av  $E(F^2)$  fås då  $E(K^2) = 0$ . I fallet med normalfördelade  $e(t)$  fås att

$$\alpha_1 = -a_1$$

$$\alpha_2 = -a_2$$

$\vdots$

$$\alpha_n = -a_n$$

$$\alpha_i = 0 \quad i \geq n+1$$

$$\beta_i = 0 \quad i = 1, 2, 3, \dots$$



I fallet med normalfördelade fel har jag då visat att den optimala prediktorn ( i minimal-variations-mening ) är linjär.

I det andra fallet då linjär prediktorn var postulerad fås att

$$\alpha_1 = -a_1$$

$$\alpha_2 = -a_2$$

$$\vdots$$

$$\alpha_n = -a_n$$

$$\alpha_i = 0 \quad i \geq n + 1$$

Jag har således visat att i bägge fallen blir den optimala prediktorn en linjärkombination av de föregående n stycken observationerna

$$\hat{y}(t + 1|t) = -a_1 y(t) - a_2 y(t - 1) - \dots - a_n y(t - n + 1)$$

Heuristiskt skulle man kunna sluta sig till att detta är den bästa prediktorn, ty medelvärdet av  $e(t)$  är noll. Det bästa jag då kan göra är att handla som om  $e(t)$  var identiskt lika med noll och då fås samma prediktor som ovan.

## 2.2 K-stegsprediktor. Linjär prediktor

Det går lätt att generalisera till en k-stegsprediktor. Vi har följande process

$$y(t + k) = \frac{1}{A(z^{-1})} e(t + k)$$

Vidare känner vi  $y(t)$ ,  $y(t-1)$ ... . Följande identitet införes

$$1 = E_{k-1}(z^{-1}) A(z^{-1}) + z^{-k} F_{n-1}(z^{-1})$$

där

$$E_{k-1}(z^{-1}) = 1 + e_1 z^{-1} + \dots + e_{k-1} z^{-(k-1)}$$

$$F_{n-1}(z^{-1}) = f_0 + f_1 z^{-1} + \dots + f_{n-1} z^{-(n-1)}$$

Processen kan då skrivas som

$$y(t+k) = \underbrace{E_{k-1}(z^{-1}) e(t+k)}_{\text{beror av } e(t+k) \dots} + \underbrace{\frac{F_{n-1}(z^{-1})}{A(z^{-1})} \cdot z^{-k} e(t+k)}_{\text{beror av } e(t), e(t-1) \dots}$$

...e(t+1)

Den andra delen kan uttryckas exakt

$$\begin{aligned} \frac{F_{n-1}(z^{-1})}{A(z^{-1})} \cdot z^{-k} e(t+k) &= \frac{F_{n-1}(z^{-1})}{A(z^{-1})} \cdot e(t) = \\ &= \frac{F_{n-1}(z^{-1})}{A(z^{-1})} \cdot A(z^{-1}) y(t) = \\ &= F_{n-1}(z^{-1}) y(t) \end{aligned}$$

Medan den första delen är beroende av de stokastiska variablerna  $\{e(j)\}_{t+1}^{t+k}$  och den bästa prediktorn blir då

$$\hat{y}(t+k|t) = F_{n-1}(z^{-1}) y(t)$$

och felet blir

$$E_{k-1}(z^{-1}) e(t+k)$$

Medelkvadratfelet kommer då att bli

$$E(E_{k-1}(z^{-1}) e(t+k)) = \sigma^2(1 + e_1^2 + \dots + e_{k-1}^2)$$

Det gäller nu att få fram koefficienterna i polynomen  $E_{k-1}(z)$  och  $F_{n-1}(z)$ .

Enligt den ovan införda identiteten gäller

$$1 \equiv (1 + e_1 z + e_2 z^2 + \dots + e_{k-1} z^{k-1})(1 + a_1 z + a_2 z^2 + \dots + a_n z^n) + z^k (f_0 + f_1 z + \dots + f_{n-1} z^{n-1})$$

Identifiering av koefficienterna ger

$$\left\{ \begin{array}{l} 1 = 1 \\ a_1 + e_1 = 0 \\ e_2 + a_2 + a_1 e_1 = 0 \\ e_3 + a_3 + a_1 e_3 + e_1 a_2 = 0 \\ \vdots \\ a_n e_{k-1} + f_{n-1} = 0 \end{array} \right.$$

Man ser att ekvationerna kan lösas rekursivt. Ur den andra fås  $e_1$  som insatt i den tredje ger  $e_2$  osv. Koefficienterna  $f_0 \dots f_{n-1}$  kommer ej in i ekvationerna förrän alla  $e_i$   $i = 1 \dots k-1$  är kända och  $f_0 \dots f_{n-1}$  kan också lösas rekursivt ur ekvationerna. Ekvationerna kan sammanfattas i följande tre formler då  $e_0 = a_0 = 1$

$$\left\{ \begin{array}{ll} \sum_{i=0}^j a_i e_{j-i} = 0 & 0 \leq j \leq k-1 \\ \sum_{i=j-k+1}^j a_i e_{j-i} + f_{j-k} = 0 & k \leq j \leq n \\ \sum_{i=j-k+1}^n a_i e_{j-i} + f_{j-k} = 0 & n+1 \leq j \leq n+k-1 \end{array} \right.$$

Sammanfattningsvis fås för k-stegsprediktorn att

$$\begin{aligned} \hat{y}(t+k|t) &= F_{n-1}(z^{-1}) y(t) = \\ &= f_0 y(t) + f_1 y(t-1) + \dots + f_{n-1} y(t-n+1) \end{aligned}$$

$$E((\hat{y}(t+k|t) - y(t+k))^2) = \sigma^2(1 + e_1^2 + \dots + e_{k-1}^2)$$

Där  $E_{k-1}(z^{-1})$  och  $F_{n-1}(z^{-1})$  fås ur identiteten

$$1 \equiv A(z^{-1}) E_{k-1}(z^{-1}) + z^{-k} F_{n-1}(z^{-1})$$

3. UPPSKATTNING AV PARAMETRARNA  $a_1, a_2 \dots a_n$  I PROCESSEN

$$y(t) + a_1 y(t-1) + \dots + a_n y(t-n) = e(t) \text{ DÅ UTDATA ÄR GIVNA}$$

Jag kommer i det följande att hålla mig till förutsättningen att  $e(t)$  är normalfördelade med medelvärdet 0 och spridningen  $\sigma$ . När denna förutsättning är fastslagen så är det naturligt att vid uppskattningen av parametrarna använda sig av maximum-likelihood eller minsta kvadratmetoden. Jag kommer att använda mig av en maximum-likelihood-uppskattning.

Om felet  $e(t)$  hade varit identiskt lika med noll skulle följande ekvationer skrivna på matrisform vara uppfyllda.

$$\begin{bmatrix} y(n) & y(n-1) & \dots & y(1) \\ y(n+1) & y(n) & & y(2) \\ \vdots & & & \\ y(N-1) & & & y(N-n) \end{bmatrix} \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \\ \vdots \\ \hat{a}_n \end{bmatrix} = \begin{bmatrix} -y(n+1) \\ -y(n+2) \\ \vdots \\ -y(N) \end{bmatrix} \quad (3.1)$$

Vilket med kompaktare beteckningar kan skrivas

$$Y_N \cdot \hat{A}_N = X_N$$

$$\text{Felet } E = \begin{bmatrix} e(n+1) \\ \vdots \\ e(N) \end{bmatrix} = Y_N \hat{A}_N - X_N$$

Nu införes följande beteckning

$$V = \sum_{t=n+1}^N e(t)^2 = E^T E$$

$V$  kallas för förlustfunktionen.

Fördelningsfunktionen för  $e(t)$  är

$$\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{e(t)^2}{2\sigma^2}}$$

och logaritmen för likelihoodfunktionen blir

$$\log L = -\frac{N-n}{2} \log 2\pi\sigma^2 - \frac{1}{2\sigma^2} \sum_{t=n+1}^N e(t)^2$$

Att maximera den är detsamma som att minimera förlustfunktionen.

$$\begin{aligned} V &= \sum_{t=n+1}^N e(t)^2 = E^T E = (Y_N \hat{A}_N - X_N)^T (Y_N \hat{A}_N - X_N) = \\ &= \hat{A}_N^T Y_N^T Y_N \hat{A}_N - \hat{A}_N^T Y_N^T X_N - X_N^T Y_N \hat{A}_N + X_N^T X_N = \\ &= X_N^T X_N + \{ \hat{A}_N - \{Y_N^T Y_N\}^{-1} Y_N^T X_N \}^T Y_N^T X_N \cdot \\ &\quad \cdot \{ \hat{A}_N - \{Y_N^T Y_N\}^{-1} Y_N^T X_N \} - X_N^T Y_N \{Y_N^T Y_N\}^{-1} Y_N^T X_N \end{aligned}$$

$V$  är minimum om

$$\hat{A}_N = \{ Y_N^T Y_N \}^{-1} Y_N^T X_N$$

vilket är maximum-likelihood-uppskattningen av parametrarna  $a_1, a_2, \dots, a_n$ .

Utan bevis ger jag ett par egenskaper hos maximum-likelihood-uppskattningen. Närmare detaljer och bevis kan läsas t.ex. i Kendall & Stuart: "The Advanced Theory of Statistics", Kap. 18.

Om de två första derivatorna av likelihoodfunktionen existerar i ett intervall, som innehåller de sanna parametervärdena ( $A^0$ ) och om

$$E \left( \frac{\partial \log L}{\partial A} \right) = 0 \quad \text{och} \quad E \left( \frac{\partial^2 \log L}{\partial A^2} \right)$$

existerar och är skilt från noll i intervallet, så gäller att ML-uppskattningen  $\hat{A}$  är asymptotiskt normalfördelad med medelvärdet  $A_0$  och kovariansmatrisen  $I(A)^{-1}$  där

$$I(A) = - E \left( \frac{\partial^2 \log L}{\partial A^2} \right)$$

$I(A)$  kallas för informationsmatrisen.

Då  $A$  är en kolonnvektor definieras  $\frac{\partial^2 \log L}{\partial A^2}$  som en matris

$$\frac{\partial^2 \log L}{\partial A^2} = \begin{bmatrix} \frac{\partial^2 \log L}{\partial a_1 \partial a_1} & \frac{\partial^2 \log L}{\partial a_1 \partial a_2} & \dots & \frac{\partial^2 \log L}{\partial a_1 \partial a_n} \\ \vdots & & & \\ \frac{\partial^2 \log L}{\partial a_n \partial a_1} & \dots & & \frac{\partial^2 \log L}{\partial a_n \partial a_n} \end{bmatrix}$$

I vårt fall är likelihoodfunktionens logaritm

$$\log L = - \frac{N-n}{2} \log 2\pi \sigma^2 - \frac{1}{2\sigma^2} \sum_{t=n+1}^N e(t)^2$$

Då blir

$$\frac{\partial^2 \log L}{\partial A^2} = - \frac{1}{2\sigma^2} \frac{\partial^2 V}{\partial A^2} = - \frac{1}{\sigma^2} Y^T Y$$

$$\implies I(A) = \frac{1}{\sigma^2} Y^T Y, \quad I(A)^{-1} = \sigma^2 \{ Y^T Y \}^{-1}$$

Asymptotiskt gäller således att

$$\hat{A} \in N(A^0, \sigma^2 \{ Y^T Y \}^{-1})$$

Innebörden i detta är att asymptotiskt kommer vår uppskattning  $\hat{A}$  att ha det riktiga medelvärdet  $A^0$  och att

$$E((\hat{a}_i - a_i^0)^2) = (I_{ii}^{-1})$$

Det vill säga att diagonalelementen i informationsmatrisens invers är medelkvadratfelelen i uppskattningarna av parametrarna  $a_i$ .

Fördelen med ML-uppskattningen är att samtidigt som den ger en uppskattning av parametrarna så ger den ett mått på hur noggrann denna uppskattning är.

Vi skall nu se på problemet ur en annan synvinkel.

Genom att införa så kallade pseudoinverser kan vi få en lösning till systemet

$$Y_N \hat{A}_N = X_N$$

En pseudoinvers  $A^+$  till en rektangulär matris  $A$  definieras genom

$$A A^+ A = A$$



Denna definition ger ej någon entydig pseudoinvers. Pseudoinversen kan dock fås entydig om man tillfogar tre stycken axiom

$$A^+ AA^+ = A^+$$

$$(AA^+)^T = AA^+$$

$$(A^+A)^T = A^+A$$

Penrose har visat att  $x^0 = A^+b$  ger lösningen till systemet  $Ax = b$  om en sådan existerar och den bästa approximationen till en lösning då det ej existerar någon.

Låt oss återgå till vårt ekvationssystem ovan. Om  $Y_N^T Y_N$  ej är singular så är pseudoinversen till  $Y_N$

$$Y_N^+ = \{ Y_N^T Y_N \}^{-1} Y_N^T$$

Detta medför att lösningen eller den bästa approximativa lösningen till  $Y_N \hat{A}_N = X_N$  ges av

$$\hat{A}_N = \{ Y_N^T Y_N \}^{-1} Y_N^T X_N$$

vilket överensstämmer med resultatet som vi kom fram till då ML-uppskattningen användes.

## 4. ADAPTIV PREDIKTION OCH PARAMETERUPPSKATTNING

I avdelning 3 har jag visat att en uppskattning av parametrarna i processen

$$y(t) + a_1 y(t-1) + \dots + a_n y(t-n) = e(t)$$

ges av

$$\hat{A}_N = \{ Y_N^T Y_N \}^{-1} Y_N^T X_N$$

Denna uppskattning används nu för att prediktera  $y(t)$  ett steg framåt, på samma sätt som i avdelning 2.1. dvs

$$\hat{y}(t+1|t) = -\hat{a}_1 y(t) - \hat{a}_2 y(t-1) - \dots - \hat{a}_n y(t-n+1)$$

Hittills har problemet bara varit att göra en parameteruppskattning och en prediktion då ett visst antal utvärden var kända. Nu uppkommer problemet hur man skall göra då man tänker sig detta insatt i ett system, som hela tiden ger nya utdata. Det är då naturligt att använda sig av dessa data för att förbättra den tidigare uppskattningen av systemet. Frågan är nu: Behöver man då ett nytt utdata erhålles göra om hela den föregående beräkningen eller kan man använda denna för en rekursiv beräkning? Jag skall visa att det är möjligt att till stor del använda sig av den föregående beräkningen.

När ett nytt utvärde erhållits kan ännu en ekvation skrivas upp i systemet (3.1) nämligen

$$a_1 y(N) + a_2 y(N-1) + \dots + a_n y(N-n+1) = -y(N+1)$$

så att det nya systemet blir

$$\begin{bmatrix} y(n) & y(n-1) & \dots & y(1) \\ \vdots & & & \vdots \\ y(N-1) & & & y(N-n) \\ \hline y(N) & y(N-1) & \dots & y(N-n+1) \end{bmatrix} \begin{bmatrix} \hat{a}_1 \\ \vdots \\ \hat{a}_n \end{bmatrix} = \begin{bmatrix} -y(n+1) \\ \vdots \\ -y(N) \\ \hline -y(N+1) \end{bmatrix}$$

eller med kompaktare beteckningar

$$Y_{N+1} \hat{A}_{N+1} = X_{N+1}$$

där

$$Y_{N+1} = \begin{bmatrix} Y_N \\ \hline d_{N+1} \end{bmatrix} \quad X_{N+1} = \begin{bmatrix} X_N \\ \hline -y(N+1) \end{bmatrix}$$

och  $d_{N+1} = \{ y(N), y(N-1), \dots, y(N-n+1) \}$

Lösningen till systemet ges nu av

$$\hat{A}_{N+1} = \{ Y_{N+1}^T Y_{N+1} \}^{-1} Y_{N+1}^T X_{N+1}$$

Nu är

$$Y_{N+1}^T Y_{N+1} = \left\{ \begin{array}{c} Y_N^T \\ \vdots \\ d_{N+1}^T \end{array} \right\} \left\{ \begin{array}{c} Y_N \\ \hline d_{N+1} \end{array} \right\} = \{ Y_N^T Y_N + d_{N+1}^T d_{N+1} \}$$

$$Y_{N+1}^T X_{N+1} = \left\{ \begin{array}{c} Y_N^T \\ \vdots \\ d_{N+1}^T \end{array} \right\} \left\{ \begin{array}{c} X_N \\ \hline -y(N+1) \end{array} \right\} = \{ Y_N^T X_N - d_{N+1}^T y(N+1) \}$$

För att få fram  $\{ Y_{N+1}^T Y_{N+1} \}^{-1}$  utan att på nytt behöva invertera någon matris använder jag mig av följande matrisformel

$$\{A + b^T b\}^{-1} = A^{-1} - A^{-1} b^T \{1 + b A^{-1} b^T\}^{-1} b A^{-1}$$

där A är en kvadratisk matris och b är en radvektor.

Att formeln är riktig verifieras lätt

$$\begin{aligned} \{A + b^T b\} \{A + b^T b\}^{-1} &= \{A + b^T b\} \{A^{-1} - A^{-1} b^T \{1 + b A^{-1} b^T\}^{-1} b A^{-1}\} = \\ &= A A^{-1} - A A^{-1} b^T \{1 + b A^{-1} b^T\}^{-1} b A^{-1} + b^T b A^{-1} - \\ &\quad - b^T b A^{-1} b^T \{1 + b A^{-1} b^T\}^{-1} b A^{-1} = \\ &= I - b^T \{1 + b A^{-1} b^T\} \{1 + b A^{-1} b^T\}^{-1} b A^{-1} + b^T b A^{-1} = \\ &= I - b^T b A^{-1} + b^T b A^{-1} = I \end{aligned}$$

Med hjälp av denna formel och genom att införa  $P_N = \{Y_N^T Y_N\}^{-1}$  fås att

$$\begin{aligned} \hat{A}_{N+1} &= \{Y_N^T Y_N + d_{N+1}^T d_{N+1}\}^{-1} \{Y_N^T X_N - d_{N+1}^T y^{(N+1)}\} = \\ &= \{P_N - P_N d_{N+1}^T \{1 + d_{N+1} P_N d_{N+1}^T\}^{-1} d_{N+1} P_N\} \{Y_N^T X_N - d_{N+1}^T y^{(N+1)}\} \end{aligned}$$

Enligt ovan så är  $P_N Y_N^T X_N = \hat{A}_N$  och  $d_{N+1} \hat{A}_N = -\hat{y}^{(N+1|N)}$  då fås att

$$\begin{aligned} \hat{A}_{N+1} &= \underbrace{P_N Y_N^T X_N}_{\hat{A}_N} - P_N d_{N+1}^T y^{(N+1)} - P_N d_{N+1}^T \{1 + d_{N+1} P_N d_{N+1}^T\}^{-1} \underbrace{d_{N+1} P_N Y_N^T X_N}_{-\hat{y}^{(N+1|N)}} \\ &\quad + P_N d_{N+1}^T \{1 + d_{N+1} P_N d_{N+1}^T\}^{-1} d_{N+1} P_N d_{N+1}^T y^{(N+1)} = \\ &= \hat{A}_N + P_N d_{N+1}^T \{(\{1 + d_{N+1} P_N d_{N+1}^T\}^{-1} d_{N+1} P_N d_{N+1}^T - 1) y^{(N+1)} + \\ &\quad + \{1 + d_{N+1} P_N d_{N+1}^T\} \hat{y}^{(N+1|N)}\} \end{aligned}$$

Då  $d_{N+1} P_N d_{N+1}^T$  är en konstant så är

$$\{ 1 + d_{N+1} P_N d_{N+1}^T \}^{-1} = \frac{1}{1 + d_{N+1} P_N d_{N+1}^T} T$$

$$\Rightarrow \hat{A}_{N+1} = \hat{A}_N + P_N d_{N+1}^T \left\{ \frac{d_{N+1} P_N d_{N+1}^T - 1 - d_{N+1} P_N d_{N+1}^T}{1 + d_{N+1} P_N d_{N+1}^T} y(N+1) + \right.$$

$$\left. + \frac{1}{1 + d_{N+1} P_N d_{N+1}^T} \hat{y}(N+1|N) \right\}$$

Således har jag nu kommit fram till följande rekursiva formler:

$$\hat{A}_{N+1} = \hat{A}_N + P_N d_{N+1}^T \{ 1 + d_{N+1} P_N d_{N+1}^T \}^{-1} (\hat{y}(N+1|N) - y(N+1))$$


---

och

$$P_{N+1} = P_N - P_N d_{N+1}^T \{ 1 + d_{N+1} P_N d_{N+1}^T \}^{-1} d_{N+1} P_N$$


---

Termerna i de rekursiva formlerna kan nu tolkas fysikaliskt.

$$\sigma^2 P_N = \sigma^2 \{ Y_N^T Y_N \}^{-1}$$

är informationsmatrisens invers och

$$\hat{y}(N+1|N) - y(N+1)$$

är felet mellan det predikterade värdet och det verkliga. Man ser att den nya uppskattningen av parametrarna fås genom att ta det gamla värdet och korrigerar det med en term, som är proportionell mot prediktionsfelet.

Problemlösningen är nu väl anpassad för att kunna användas som räkneprocedur i ett adaptivt system.

## 5. KODNING

Om man direkt utgår från formlerna framtagna i avdelning 4 och startar räkneprocessen då  $N$  stycken utdata erhållits, så har detta ett par nackdelar. För det första får man lagra dessa  $N$  stycken utdata och för det andra, vilket är allvarligare, så måste man ha en invertionsprocedur i programmet. Vid en första anblick kan det synas vara svårt att komma ifrån detta, ty matrisen  $P_N$  har ej mening som variansmatris förrän  $N - n \geq 1 \Leftrightarrow N \geq n + 1$ . Men om man ser på detta tillsammans med startvärdesproblemet så ser man att ett lämpligt val av  $d_1$  är en nollvektor, dvs man förutsätter att  $y(0) = y(-1) = \dots = y(-n+1) = 0$ . Då existerar ej  $P_0$ , men man kan visa att följande gäller:

Sats  $P_n$  konstruerad rekursivt enligt

$$P_{N+1} = P_N - P_N d_{N+1}^T \{ 1 + d_{N+1}^T P_N d_{N+1} \}^{-1} d_{N+1} P_N$$

$P_0 = I \times a$  där  $I$  är enhetsmatris av ordningen  $n \times n$  konvergerar mot det sanna  $P_N$  värdet  $= \left( \sum_{s=1}^N d_s^T d_s \right)^{-1}$  då  $N \geq n + 1$  och  $a \rightarrow \infty$ .

Bevis Satsen bevisas med hjälp av induktion

$$P_1 = P_0 - P_0 d_1^T (1 + d_1^T P_0 d_1)^{-1} d_1 P_0 = \{ P_0^{-1} + d_1^T d_1 \}^{-1}$$

$$P_2 = P_1 - P_1 d_2^T (1 + d_2^T P_1 d_2)^{-1} d_2 P_1 = \{ P_1^{-1} + d_2^T d_2 \}^{-1}$$

$$\text{men } P_1^{-1} = P_0^{-1} + d_1^T d_1 \implies$$

$$P_2 = \{ P_0^{-1} + \sum_{s=1}^2 d_s^T d_s \}^{-1}$$

Antag det gäller för  $k$ , visa att det då gäller för  $k + 1$

$$\begin{aligned}
 P_{k+1} &= P_k - P_k d_{k+1}^T (1 + d_{k+1}^T P_k d_{k+1})^{-1} d_{k+1} P_k = \\
 &= \{ P_k^{-1} + d_{k+1}^T d_{k+1} \}^{-1} = \\
 &= \{ P_0^{-1} + \sum_{s=1}^k d_s^T d_s + d_{k+1}^T d_{k+1} \}^{-1} = \\
 &= \{ P_0^{-1} + \sum_{s=1}^{k+1} d_s^T d_s \}^{-1} \\
 \therefore P_N &= \{ P_0^{-1} + \sum_{s=1}^N d_s^T d_s \}^{-1}
 \end{aligned}$$

Att visa att  $P_N \rightarrow \left( \sum_{s=1}^N d_s^T d_s \right)^{-1}$  är ekvivalent med att visa att

$$P_N^{-1} \rightarrow \sum_{s=1}^N d_s^T d_s$$

$$\begin{aligned}
 P_0^{-1} &= \frac{1}{a} \cdot I \implies P_N^{-1} = P_0^{-1} + \sum_{s=1}^N d_s^T d_s = \\
 &= \frac{1}{a} \cdot I + \sum_{s=1}^N d_s^T d_s \rightarrow \sum_{s=1}^N d_s^T d_s
 \end{aligned}$$

då  $a \rightarrow \infty$  vsb

Med hjälp av satsen ovan har jag kommit fram till att det är lämpligt att välja  $P_0$  till en diagonalmatris med stora värden i diagonalen. En anmärkning kan här vara på sin plats. Man får vara försiktig då man väljer konstanten  $a$  så att den inte väljs för stor, för då kan det hända att datamaskinens begränsade noggrannhet gör att beräkningen av matrisen blir felaktig. Det kan t.ex. hända att matrisen, som skall vara positivt semidefinit, kan bli negativt definit, vilket kan förorsaka fel i den fortsatta räkningen.

Då  $P_0^{-1} = \frac{1}{a} \cdot I$  är proportionell mot informationsmatrisen är det naturligt att elementen i  $P_0^{-1}$  skall vara små då informationen innan något utvärde har observerats är = 0. Att diagonalelementen ej också valts till 0 beror på att inversen  $P_0$  måste existera.

Sammanfattningsvis är det följande formler, som skall programmeras:

$$\left\{ \begin{array}{l} \hat{A}_{N+1} = \hat{A}_N + P_N d_{N+1}^T \{1 + d_{N+1}^T P_N d_{N+1}\}^{-1} (\hat{y}(N+1|N) - y(N+1)) \\ P_{N+1} = P_N - P_N d_{N+1}^T \{1 + d_{N+1}^T P_N d_{N+1}\}^{-1} d_{N+1} P_N \\ \hat{y}(N+1|N) = - d_{N+1} \hat{A}_N \end{array} \right.$$

Som begynnelsevärden har valts:

$$\hat{A}_0 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \text{av ordningen } n \times 1$$

$$d_1 = \begin{bmatrix} 0, 0, \dots, 0 \end{bmatrix} \quad \text{" " } 1 \times n$$

$$P_0 = I \times a \quad \text{" " } n \times n$$

De data som behöver lagras mellan två beräkningar är

matrisen $\hat{A}_N$	$n$	tal
" $d_{N+1}$	$n$	"
" $P_N$	$n^2$	"
" $\hat{y}(N+1 N)$	$1$	"



Om man skulle få svårigheter med minnesutrymmet kan man utnyttja att  $P_N$  är en symmetrisk matris. Då kan  $P_N$  lagras i  $\frac{n^2 + n}{2}$  minnesceller.

Programmet, som återges nedan, innehåller ej några programmerings-tekniska finesser utan är en rättfram kodning av de tidigare härledda formlerna.

Då man på den datamaskin, som stod till förfogande (SMIL i Lund) ej kan programmera med dynamiska fält har programmet delats upp i två delar. Den första deklarerar matriserna, processens ordning samt antalet värden för vilka beräkningen skall göras. Den andra remsan innehåller resten av programmet. Det blir således bara värden på den första remsan som ändras då beräkningen utföres för processer av olika ordning.

Kommentaren i programmet ger förklaring till de använda beteckningarna.

För varje steg sker utskrift av stegnumret  $i$ , det verkliga  $y$ -värdet,  $y(i)$ , det predikterade  $y$ -värdet,  $\hat{y}(i|i-1)$ , den beräknade variansen samt matriserna  $A_N$  och  $P_N$ .

Då programmet endast används för att identifiera en process är det tillräckligt att endast skriva ut slutresultaten.

Några ytterligare anmärkningar angående programmet torde ej behövas.

IGEN:

begin array yu[1:100],a,d,K[1:3],p[1:3,1:3]; integer n,N;

n:=3;

N:=100;;

comment Identifiering av parametrarna i en process som beskrivs av  
 $y[t]+a[1]\times y[t-1]+ \dots +a[n]\times y[t-n]=e[t]$ , då felet  $e[t]$   
 är normalfördelat med medelvärdet 0 och variansen var.

V: förlustfunktionen

a: kolonnvektor med parameteruppsättningarna

d: radvektor med utdata

p: n×n matris = 1/var×(informationsmatrisen)

K: p×d (transponat)

var: uppsatt värde på felets varians

yb: prediterat utdata

n: processens ordning

N: antalet utvärden för vilka beräkningarna skall göras;

integer i,j,k; real yb,y,H,V,var;

begin

for i:=1 step 1 until N do yu[i]:=read;

comment inläsning av utdata;

for i:=1 step 1 until n do

begin a[i]:=0;d[i]:=0;

for k:=1 step 1 until n do p[i,k]:=0;p[i,i]:=10<sup>-5</sup>;

end begynnelsevärden;

yb:=0;V:=0;

i:=1;

OM: y:=yu[i];punch(1);print(4,0,i);print(-15,3,y);print(-3,3,yb);

V:=V+(y-yb)<sup>2</sup>;var:=V/i;print(var);punch(1);

comment utskrift av i, y[i], yb[i|i-1] och var;

for j:=1 step 1 until n do

begin K[j]:=0; for k:=1 step 1 until n do

K[j]:=K[j]+p[j,k]×d[k]

end beräkning av K;

H:=0; for j:=1 step 1 until n do H:=H+d[j]×K[j];H:=1/(1+H);

for j:=1 step 1 until n do

begin a[j]:=a[j]+K[j]×H×(yb-y);print(-3,3,a[j]);

for k:=1 step 1 until n do

begin p[j,k]:=p[j,k]-H×K[j]×K[k];print(-5,3,p[j,k])

end;punch(1);

end utskrift av a och p;

punch(1);

for j:=n step -1 until 2 do d[j]:=d[j-1]; d[1]:=y;

yb:=0;

for j:=1 step 1 until n do yb:=yb-d[j]×a[j]; i:=i+1;

if i<N+1 then go to OM else

begin punch(1);print(4,0,i);print(-15,3,yb);

punch(3)

end

end;go to IGEN

end

## 6. REDOVISNING AV EXEMPEL

De exempel som skall redovisas är av två typer:

$$a) A(z^{-1}) y(t) = e(t)$$

$$b) A(z^{-1}) y(t) = C(z^{-1}) e(t) \text{ där } C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \dots$$

Det är processer av typ a, som programmet är konstruerat för, men för att visa begränsningarna i programmet har ett antal processer av typ b identifierats.

Den bästa linjära prediktorn för typ b ges av (för härledning se kompendium för fortsättningskursen i regleringsteknik):

$$\hat{y}(t+1|t) = \frac{F_{n-1}(z^{-1})}{C(z^{-1})} y(t) \quad (6.1)$$

dvs

$$\begin{aligned} \hat{y}(t+1|t) = & -c_1 \hat{y}(t|t-1) - c_2 \hat{y}(t-1|t-2) - \dots + \\ & + f_0 y(t) + f_1 y(t-1) + \dots + f_{n-1} y(t-n+1) \end{aligned}$$

där  $F_{n-1}(z^{-1})$  fås ur identiteten

$$C(z^{-1}) = A(z^{-1}) + z^{-1} F_{n-1}(z^{-1})$$

(Jämför härledning av k-stegsprediktorn i avdelning 2.2). I prediktorn ingår en linjärkombination av ett antal föregående predikt.

Den prediktor som programmet ger består endast av en linjärkombination av föregående y-värden, dvs den är en autoregression. Detta gör att det ej är möjligt att tillfredsställande identifiera processer av typ b annat än i vissa speciella fall. Se exempel 5-8.

En annan begränsning av prediktorn är vid identifiering av processer då  $E(y(t)) \neq 0$ . Med en prediktor av typ (6.1) får man en integration, som plockar upp medelvärdet. Den bildar således medelvärdet av samtliga föregående värden. Denna utväg finns ej i det här redovisade programmet, utan man får använda sig av ett glidande medelvärde av ett begränsat antal föregående värden. För vidare kommentarer till detta se exempel 4.

I samband med redovisningen av exemplen kan vissa generella iakttagelser göras.

För det första vid identifiering som första ordningens system blir koefficienten om det verkliga systemet ej är av första ordningen, ungefär -1. Rent intuitivt är det klart att det bästa, som man då kan göra är att sätta prediktorn lika med det föregående värdet.

Vidare är parameteruppskattningen i de första stegen dålig och parametervärdena fluktuerar väldigt för att efter cirka tio steg ställa in sig på ett ganska konstant värde. Detta för med sig att prediktionsfelen i de första stegen är stora och att förlustfunktionen får stora bidrag i början av beräkningen.

Då man skall bedöma hur bra system av olika ordning identifierar processen kan man använda sig av två mått. För det första kan man använda kovariansmatrisen och se hur stor den beräknade spridningen är på de olika parametrarna  $a_i$ . Detta är en bra metod då man känner den verkliga processen. Då denna däremot är okänd kan man använda förlustfunktionen  $V(t)$ . Förlustfunktionen definieras enligt avdelning 3 som

$$V(t) = \sum_{i=1}^t (\hat{y}(i|i-1) - y(i))^2$$

För att komma ifrån inverkan av identifieringens insvängningsförlopp så används förlustfunktionens värde i slutpunkten minskat med förlustfunktionens värde vid  $t = 20$ , dvs  $V(N) - V(20)$ , vid jämförelse mellan olika system.

Diagrammen, som åskådliggör de olika exemplen, finns i appendix A. Då parametrarnas insvängningsförlopp är likartat för alla exempel, så har det bara redovisats i diagramform för exempel 1.

### Exempel 1

Processen

$$y(t) - 1.5y(t - 1) + 0.7y(t - 2) = e(t), \quad N = 100$$

identifierad som första, andra och tredje ordningens system.

Programmet gav följande resultat:

Verkliga värden                      Beräknade värden på koefficienterna  $a_i$  och spridningen  $\sigma$

$a_i$	1:a ordn.		2:a ordn.		3:e ordn.	
	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$
- 1.5	-0.8758	0.0389	-1.5182	0.0775	-1.5727	0.1171
0.7	-	-	0.7496	0.0783	0.8620	0.1973
0	-	-	-	-	0.0746	0.1202
$V(100)-V(20)$	144.17		63.29		65.93	

Predikten och koefficienternas variation under beräkningens gång redovisas i diagram 1.2-1.11. De verkliga värdena ges av diagram 1.1. När processen har identifierats med 2:a och 3:e ordningens system har resultaten i stort sätt blivit desamma. Koefficienterna ligger inom  $1\sigma$ -gränsen. Dock är spridningen mindre vid identifikation som andra ordningens autoregression.

Exempel 2

Processen

$$y(t) - 0.6y(t - 1) = e(t) ; N = 100$$

identifierad som första och andra ordningens system.

De verkliga värdena ges i diagram 2.1, de predikterade för första ordningens system i diagram 2.2 och för andra ordningens system i diagram 2.3. Som synes har programmet svårt att prediktera processen. Detta beror på att processen är av första ordningen och att bruset då har stor inverkan. Men programmet identifierar koeficienterna bra, se tabell.

Resultat:

Verkliga värden	Beräknade värden			
	1:a ordn.		2:a ordn.	
	$a_i$	$\sigma$	$a_i$	$\sigma$
-0.6000	-0.6202	0.0905	-0.6766	0.1165
0.0000	-		0.0904	0.1173
V(100)-V(20)	62.43		63.78	

De verkliga värdena ligger inom  $1\sigma$ -gränsen. På förlustfunktionen ser man att prediktionen blir lika bra vare sig processen identifieras med första eller andra ordningens system.

Exempel 3

Stegsvaret för första och andra ordningens system.

Stegsvaret för programmet kan relativt lätt undersökas analytiskt för första och andra ordningens system.

Om steget göres i noll dvs  $y(0) = y(1) = \dots = b$  och matrisen  $P$  ges begynnelsevärdet  $a \times I$  där  $I$  är enhetsmatris, så kan man med induktion visa att följande generella formler gäller:

Första ordningens system:

$$\left\{ \begin{array}{l} \hat{a}(t) = - \frac{t ab^2}{1 + t ab^2} \\ P(t) = \frac{a}{1 + t ab^2} \\ \hat{y}(t|t-1) = \frac{(t-1) ab^3}{1 + (t-1)ab^2} \end{array} \right.$$

Samt för andra ordningens system:

$$\hat{A}(t) = - \begin{bmatrix} \frac{(t-1) a^2 b^4 + t ab^2}{1 + (2t-1) ab^2 + (t-1) a^2 b^4} \\ \frac{(t-1) ab^2}{1 + (2t-1) ab^2 + (t-1) a^2 b^4} \end{bmatrix}$$

$$P(t) = \begin{bmatrix} \frac{a(1 + (t-1) ab^2)}{1 + (2t-1) ab^2 + (t-1) a^2 b^4} & \frac{-(t-1) a^2 b^2}{1 + (2t-1) ab^2 + (t-1) a^2 b^4} \\ \frac{-(t-1) a^2 b^2}{1 + (2t-1) ab^2 + (t-1) a^2 b^4} & \frac{a(1 + t ab^2)}{1 + (2t-1) ab^2 + (t-1) a^2 b^4} \end{bmatrix}$$

$$\hat{y}(t|t-1) = \frac{b\{(2t-3)ab^2 + (t-2) a^2 b^4\}}{1 + (2t-3)ab^2 + (t-2) a^2 b^4}$$

Som synes av formlerna har programmet den trevliga egenskapen att det är globalt asymptotiskt stabilt åtminstone för stegstörning. Vidare kryper  $\hat{y}(t|t-1)$  upp mot det riktiga värdet utan överslängar. Ett problem som borde undersökas är om detta gäller för system av godtycklig ordning. Formlerna ovan tyder på att så skulle vara fallet. Enligt bifogade utskrifter från datamaskin, se appendix B, ses att prediktet snabbt kryper upp mot det riktiga värdet, särskilt för stora steg. I utskrifterna är redovisat steget  $b$ , ordningsnummer (på tidsögonblicket), prediktet  $\hat{y}(t|t-1)$ , förlustfunktionen samt kovariansmatrisen i sista steget. Stegsvaret ges för stegen 1, 10 och 50, för första och andra ordningens system då startvärdet på  $P(t)$  är  $10^5 \times I$ .

#### Exempel 4

Identifiering av processer då utsignalens medelvärde är skilt från noll.

a. Processen i exempel 1 har identifierats som första, andra, tredje och fjärde ordningens system då medelvärdet av utsignalen  $E y(t)$  har varit = 20. Beräkningarna gav följande resultat:

Verkliga värden	Beräknade värden							
	1:a		2:a		3:e		4:e	
$a_i$	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$
-1.5000	-0.9964	0.0049	-1.1890	0.0417	-1.1800	0.0426	-1.1820	0.0427
0.7000	-		0.1948	0.0418	0.1345	0.0658	0.1294	0.0661
0	-		-		0.0520	0.0431	0.1030	0.0663
0	-		-		-		-0.0446	0.0436
$V(100)-V(20)$	152.07		131.42		130.45		129.96	



b. Processen i exempel 2 har identifierats som andra ordningens system då  $E y(t) = 5,20$  och  $100$ . Beräkningarna gav följande resultat:

Verkliga värden	Beräknade värden					
	b = 5		b = 20		b = 100	
	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$
-0.6000	-0.8935	0.0935	-0.9714	0.0452	-0.9965	0.0134
0.0000	-0.0932	0.0938	-0.0280	0.0454	-0.0035	0.0135
$V(100)-V(20)$	75.69		76.00		76.21	

Man ser att programmet inte alls kan identifiera processerna. Anmärkningsvärt är att trots koefficienterna skiljer sig så från de verkliga värdena så är de beräknade värdena på  $\sigma$  små.

Vi skall nu titta lite närmare på de båda fallen från olika synvinklar.

#### Fall a:

Betrakta vektorn

$$\hat{A} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}$$

som en stokastisk variabel med medelvärdet  $m$  och kovariansmatrisen  $E(\hat{A} - A^0)(\hat{A} - A^0)^T = P$ . Låt  $T$  bestå av egenvektorerna till  $\hat{A}$  dvs

$$T = \{ e_1 \ e_2 \ \dots \ e_n \} \quad \text{där} \quad e_i = \begin{bmatrix} e_i^1 \\ e_i^2 \\ \vdots \\ e_i^n \end{bmatrix}$$

Då gäller under förutsättning att  $P$  har distinkta egenvärden att  $T T^T = I$  ty egenvektorerna har normerats och då  $P$  är symmetrisk så är de också ortogonala dvs  $T^{-1} = T^T$ . Transformera  $P$  på följande sätt

$$T^{-1} P T = T^T P T$$

Då gäller att

$$T^T P T = \Lambda = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_n \end{bmatrix}$$

där  $\lambda_i$  är egenvärdena till kovariansmatrisen  $P$ .

$$\text{Bevis: } T^T P T = \Lambda \iff P T = T \Lambda$$

Betrakta  $P$  som sammansatt av  $n$  radvektorer av ordningen  $1 \times n$

$$P = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} \quad \text{där } p_i = \{ p_i^1 \ p_i^2 \ \dots \ p_i^n \}$$

då gäller att

$$P T = \begin{bmatrix} p_1 e_1 & p_1 e_2 & \dots & p_1 e_n \\ p_2 e_1 & & & \\ \vdots & & & \\ p_n e_1 & \dots & & p_n e_n \end{bmatrix}$$

$$\text{där } p_i e_j = \sum_{k=1}^n p_i^k e_j^k$$

För varje kolumn gäller

$$\begin{bmatrix} p_1 e_i \\ p_2 e_i \\ \vdots \\ p_n e_i \end{bmatrix} = P e_i = \lambda_i e_i$$

på grund av egenskaperna hos egenvärden och egenvektorer.

Dvs

$$PT = \{ \lambda_1 e_1, \lambda_2 e_2 \dots \lambda_n e_n \}$$

Vidare gäller att

$$T^{-1} \Lambda = \{ e_1, e_2 \dots e_n \} \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_n \end{bmatrix} =$$

$$= \{ \lambda_1 e_1, \lambda_2 e_2, \dots, \lambda_n e_n \}$$

Således är  $T^{-1} PT = \Lambda$  vsb

Betrakta nu följande transformtion av  $\hat{A}$

$$z = T^{-1} \hat{A}$$

$$\text{Då gäller att } Ez = E(T^{-1} \hat{A}) = T^{-1} E \hat{A} = T^{-1} m$$

vidare att

$$E\{(z - T^{-1} m)(z - T^{-1} m)^T\} = T^{-1} E\{(\hat{A} - m)(\hat{A} - m)^T\} T = T^{-1} PT = \Lambda$$

Detta innebär att man får fram linjärkombinationer av  $a_1 \dots a_n$  som är okorrelerade.

Det visar sig nu att för både andra och tredje ordningens system så gäller att det minsta egenvärdet har egenvektorena

$$\begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \quad \text{respektive} \quad \begin{bmatrix} \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \end{bmatrix}$$

Således plockar programmet fram linjärkombinationerna  $a_1 + a_2$  och  $a_1 + a_2 + a_3$  bra. För systemet av andra ordningen är siffervärdena

$$\begin{cases} \lambda_1 = 0.000023 \\ \lambda_2 = 0.003513 \end{cases} \quad e_1 = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \quad e_2 = \begin{bmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$$

vilket ger

$$\sigma(a_1 + a_2) = 0.0068$$

$$\sigma(-a_1 + a_2) = 0.0839$$

De ovan gjorda beräkningarna visar att  $a_1$  och  $a_2$  är korrelerade medan vissa linjärkombinationer av dem är okorrelerade. Programmet försöker plocka upp medelvärdet genom att placera en rot till karakteristiska ekvationen i  $z = 1$ . Om  $z = 1$  sätts in i karakteristiska ekvationen fås

$$1 + a_1 + a_2 + \dots + a_n = 0$$

För att detta skall vara uppfyllt måste uppskattningen av  $a_1 + a_2 + \dots + a_n = -1$  vara bra. Dvs  $\sigma(a_1 + a_2 + \dots + a_n)$  skall vara liten. Vilket enligt ovan stämmer för de gjorda beräkningarna.

Fall b

Om man tittar lite närmare på tabellen ovan så ser man att tendensen är att ju större steget är desto mer närmar sig  $a_1$  och  $a_2$   $-1$  respektive  $0$ .

De i exempel 3 givna formlerna för stegsvaret ger för stora  $t$  att

$$\hat{A}(t) \approx - \begin{bmatrix} 1 - \frac{1}{2 + ab^2} \\ \frac{1}{2 + ab^2} \end{bmatrix}$$

där  $b$  är steget och  $a$  är en stor konstant (i detta fallet  $10^5$ ).

Man ser att den ovan gjorda iakttagelsen skulle stämma om processens utvärde var konstant. Men då steget blir stort så att variationen på grund av processdynamiken ej nämnvärt spelar in då kommer det att gälla även om utvärdet ej är konstant.

Detta har den tråkiga konsekvensen att den här konstruerade prediktorn ej kan användas på processer vars utvärden har ett medelvärde skilt ifrån noll. Då det endast gäller identifieringsproblemet kan svårigheten kringgås genom att först bilda medelvärdet och dra ifrån detta från alla värdena och sedan använda programmet. Men i ett adaptivt system kan ej den konstruerade prediktorn användas utan den måste modifieras så att den klarar av processer då  $E_y \neq 0$ .

Exempel 5

Processen

$$y(t) = e(t) + 0.99 e(t - 1) \quad N = 100$$

identifierad som ett första, andra och tredje ordningens system.

Denna process är av typ b (se inledningen till denna avdelning) och den optimala minimumvarians prediktorn är

$$\hat{y}(t+1|t) = \frac{0.99}{1 + 0.99z^{-1}} y(t)$$

Genom att serieutveckla kvoten  $\frac{0.99}{1 + 0.99z^{-1}}$  fås

$$\frac{0.99}{1 + 0.99z^{-1}} = 0.99 - 0.98z^{-1} + 0.97z^{-2} - 0.96z^{-3} \dots$$

Om processen skall identifieras med en autoregression måste man ta med så många termer att resten kan försummas. Vi ser att i detta exemplet är det svårt eftersom termerna i början är av samma storleksordning.

Resultat:

Beräknade värden					
1:a		2:a		3:e	
$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$
-0.4985	0.0669	-0.6915	0.0771	-0.7546	0.0836
		0.3884	0.0771	0.5011	0.0964
				-0.1631	0.0837
V(100)-V(20)	148.25	131.74		134.38	

Av diagram 5.2, 5.3 och 5.4 ses att prediktorn inte blir särskilt bra, vilket man också kan se genom att studera förlustfunktionen som i alla tre fallen är stor.

#### Exempel 6

Processen

$$y(t) = e(t) + 0.3e(t-1) \quad N = 100$$

identifierad som ett första, andra, tredje och fjärde ordningens system.

Den optimala minimumvariansprediktorn till processen är (se inledning till avdelning 6)

$$\hat{y}(t+1|t) = \frac{0.3}{1+0.3z^{-1}} y(t)$$

Serieutveckling av kvoten ger

$$\begin{aligned} \hat{y}(t+1|t) = & 0.3y(t) - 0.09y(t-1) + 0.027y(t-2) - 0.0081y(t-3) + \\ & + 0.0024y(t-4) - \dots \end{aligned}$$

Då termerna snabbt avtar skulle man kunna förmoda att processen tillfredsställande kan identifieras som en autoregression.

Resultat:

Förväntade värden	Beräknade värden							
	1:a		2:a		3:e		4:e	
	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$
-0.3000	-0.3458	0.1091	-0.3869	0.1169	-0.3766	0.1173	-0.3746	0.1185
0.0900			0.1148	0.1174	0.0644	0.1255	0.0648	0.1258
-0.0270					0.1376	0.1208	0.1324	0.1288
0.0081							0.0142	0.1222
V(100)-V(20)	62.17		63.55		63.22		64.22	

Man ser att de förväntade värdena ligger inom  $1\sigma$  gränsen utom för  $a_3$  i 3:e och 4:e ordningens system, som ligger inom  $2\sigma$  gränsen. Av förlustfunktionen att döma är predikten lika bra för alla fyra systemen.

Då man jämför den verkliga processen, diagram 6.1, med de predikterade värdena, diagram 6.2-6.5, ser man att prediktorn ej är riktigt bra.

Exempel 7

Processen

$$y(t) = x_1(t) + x_2(t) \quad \text{då}$$

$$x_1(t+1) = x_1(t) + e_1(t)$$

$$x_2(t+1) = 0.8x_2(t) + e_2(t) \quad N = 100$$

identifierad som ett första, andra och tredje ordningens system.

Om man jämför de verkliga värdena, diagram 7.1, med de predikterade, diagram 7.2-7.4, finner man att för alla tre fallen är prediktorn dålig. Man ser också en egenskap hos prediktorn, som är karakteristisk, nämligen att  $a_1$  blir ungefär  $-1$  och de andra parametervärdena blir ungefär lika med noll. Det vill säga prediktorn använder, då den ej kan plocka upp processdynamiken, det föregående utvärdet som uppskattning.

Resultat:

Beräknade värden					
1:a		2:a		3:e	
$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$
-0.9930	0.0145	-1.0293	0.0801	-1.0275	0.0801
		0.0374	0.0809	-0.0179	0.1166
				0.0553	0.0844
V(100)-V(20) 129.36		132.95		135.33	



Exempel 8

Processen

$$y(t) = x_1(t) + x_2(t) \quad \text{där}$$

$$x_1(t) = 1.90x_1(t-1) - 0.9925x_1(t-2) + e_1(t)$$

$$x_2(t) = 1.50x_2(t-1) - 0.70x_2(t-2) + e_2(t) \quad N = 500$$

identifierad som ett första, andra, tredje, fjärde och femte ordningens system.

Processen är ingen autoregression, men den är sammansatt av två processer som är det.  $x_1(t)$  är den dominerande processen, ty rötterna till den karakteristiska ekvationen ligger närmast enhetscirkelns rand, den är också mer lågfrekvent än  $x_2$ .

I diagrammen 8.1-8.6 ses att processen predikteras bra för andra, tredje, fjärde och femte ordningens system. I tabellen ser man att programmet plockar fram parametervärden, som ungefär är parametervärdena i processen  $x_1(t)$ .

Förlustfunktionen är stor, men detta beror på att processen har utvärden, som är stora jämfört med tidigare exempel.

Resultat:

Beräknade värden

	1:a		2:a		3:e		4:e		5:e	
	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$	$a_i$	$\sigma$
	-0.9529	0.0025	-1.8676	0.0082	-1.8734	0.0291	-1.8740	0.0291	-1.8591	0.0294
	-		0.9600	0.0082	0.9713	0.0551	1.0913	0.0619	1.0631	0.0623
	-		-		-0.0061	0.0292	-0.2382	0.0620	-0.1083	0.0697
	-		-		-		0.1241	0.0292	-0.0998	0.0624
	-		-		-		-		0.1198	0.0295
V(500)-V(400)	2374.42		244.36		245.73		257.44		252.06	

## 7. SAMMANFATTNING OCH SLUTSATSER

I examensarbetet har följande stokastiska process studerats

$$y(t) + a_1 y(t-1) + a_2 y(t-2) + \dots + a_n y(t-n) = e(t)$$

där  $\{e(t)\}$  är en svit okorrelerade lika fördelade stokastiska variabler.

Avsikten har varit att konstruera ett identifieringsprogram för parametrarna  $a_i$   $i = 1 \dots n$  och att konstruera en prediktor. Därvid har stor vikt lagts vid att identifieringen och prediktorn skall kunna användas i ett adaptivt system, dvs att identifieringsprogrammet skall modifiera parametervärdena allteftersom nya värden från processen erhålles.

I avdelning 2 har formler härletts för 1-steps- och k-steps-prediktorn då parametrarna  $a_1, a_2, \dots, a_n$  var kända.

Därefter har i avdelning 3 identifieringsproblemet studerats. Det har attackerats utifrån den förutsättningen att störningen  $e(t)$  är normalfördelad med medelvärdet noll. Med denna förutsättning har det varit lämpligt att använda en maximum-likelihooduppskattning för parametrarna. Denna har den fördelen att samtidigt som den ger en beräkning av parametervärdena så beräknas en kovariansmatris, som ger ett mått på hur noggrann beräkningen är.

I avdelning 4 har formlerna omformats så att de blir användbara i ett adaptivt system utan att alltför betungande räkningar skall behöva göras för varje steg. Det visar sig att den nya uppskattningen kan fås genom att använda den gamla och till den lägga en term, som beror på felet, dvs skillnaden mellan det predikerade och det verkliga utvärdet. Dessa formler är väl avpassade att kodas och användas i ett adaptivt system.

Begynnelsevärdesproblemet och kodningen diskuteras i avdelning 5.

I avdelning 6 redovisas ett antal exempel som illustrerar prediktorns för- och nackdelar. Ur exemplen kan följande slutsatser dras:

1. Programmet identifierar och predikterar autoregressioner bra. Dvs att programmet fungerar bra för de processer för vilka det är konstruerat.
2. Stegsvaret är globalt asymptotiskt stabilt.
3. Programmet klarar ej av processer för vilka utsignalens medelvärde är skilt ifrån noll, se exempel 4. För sådana processer uppfattar programmet ej processdynamiken utan strävar efter att som prediktor använda den sist erhållna utsignalen. Detta är en stor nackdel, ty bland de egenskaper, som man vill ha hos en prediktor hör just att den skall kunna plocka upp ett medelvärde och den skall kunna följa en process som driver.
4. Programmet har svårt att klara av processer, som ej är autoregressioner utom i speciella fall. Se exemplen 5, 6, 7 och 8.
5. När programmet ej kan uppfatta processdynamiken använder det som prediktor den sista utsignalen. I diagrammen ger det sig tillkänna genom att den predikterade processen ligger förskjuten en tidsenhet efter den verkliga processen. Detta kan iakttagas t.ex. i exempel 7.

Ett fortsatt studium av adaptiv prediktion på den här inslagna vägen får framförallt gå ut på att konstruera om prediktorn, så att den får sådana egenskaper att den kan följa processer, som har utsignalens medelvärde skilt från noll, samt processer som driver. Nästa steg får sedan bli att angripa problemet med en processmodell som är mer komplex, t.ex. processer där störningen kommer in på ett mer komplicerat sätt och processer som kan styras med hjälp av någon insignal.

## APPENDIX B

Stegsvaret för första och andra ordningens system, sidorna B1-3  
respektive B4-6, för stegen 1, 10 och 50.

$b = 1.10^+ u$	$\hat{q}(t t-1)$	$v(t)$
$t = 1$	$0.00000000_{10} + 0$	$1.00000000_{10} + 0$
$t = 2$	$9.99990000_{10} - 1$	$1.00000000_{10} + 0$
$t = 3$	$9.99995000_{10} - 1$	$1.00000000_{10} + 0$
$t = 4$	$9.99996667_{10} - 1$	$1.00000000_{10} + 0$
$t = 5$	$9.99997500_{10} - 1$	$1.00000000_{10} + 0$
$t = 6$	$9.99998001_{10} - 1$	$1.00000000_{10} + 0$
$t = 7$	$9.99998334_{10} - 1$	$1.00000000_{10} + 0$
$t = 8$	$9.99998572_{10} - 1$	$1.00000000_{10} + 0$
$t = 9$	$9.99998751_{10} - 1$	$1.00000000_{10} + 0$
$t = 10$	$9.99998889_{10} - 1$	$1.00000000_{10} + 0$
$t = 11$	$9.99999000_{10} - 1$	$1.00000000_{10} + 0$
$t = 12$	$9.99999091_{10} - 1$	$1.00000000_{10} + 0$
$t = 13$	$9.99999167_{10} - 1$	$1.00000000_{10} + 0$
$t = 14$	$9.99999231_{10} - 1$	$1.00000000_{10} + 0$
$t = 15$	$9.99999286_{10} - 1$	$1.00000000_{10} + 0$
$t = 16$	$9.99999334_{10} - 1$	$1.00000000_{10} + 0$
$t = 17$	$9.99999375_{10} - 1$	$1.00000000_{10} + 0$
$t = 18$	$9.99999412_{10} - 1$	$1.00000000_{10} + 0$
$t = 19$	$9.99999444_{10} - 1$	$1.00000000_{10} + 0$
$t = 20$	$9.99999474_{10} - 1$	$1.00000000_{10} + 0$
$t = 21$	$9.99999500_{10} - 1$	$1.00000000_{10} + 0$
$t = 22$	$9.99999524_{10} - 1$	$1.00000000_{10} + 0$
$t = 23$	$9.99999546_{10} - 1$	$1.00000000_{10} + 0$
$t = 24$	$9.99999565_{10} - 1$	$1.00000000_{10} + 0$
$t = 25$	$9.99999584_{10} - 1$	$1.00000000_{10} + 0$
$t = 26$	$9.99999600_{10} - 1$	$1.00000000_{10} + 0$
$t = 27$	$9.99999616_{10} - 1$	$1.00000000_{10} + 0$
$t = 28$	$9.99999630_{10} - 1$	$1.00000000_{10} + 0$
$t = 29$	$9.99999643_{10} - 1$	$1.00000000_{10} + 0$
$t = 30$	$9.99999655_{10} - 1$	$1.00000000_{10} + 0$

$P(t) \quad 3.33333334_{10} - 2$

	$\hat{y}(t t-1)$	$V(t)$
$b = 1.0 \times 1$		
$t = 1$	0.000000000 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
$t = 2$	9.99999900 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	3 9.99999950 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	4 9.99999967 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	5 9.99999976 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	6 9.99999981 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	7 9.99999984 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	8 9.99999986 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	9 9.99999988 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	10 9.99999990 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	11 9.99999991 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	12 9.99999991 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	13 9.99999992 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	14 9.99999993 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	15 9.99999994 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	16 9.99999994 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	17 9.99999994 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	18 9.99999994 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	19 9.99999995 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	20 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	21 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	22 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	23 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	24 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	25 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	26 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	27 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	28 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	29 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
	30 9.99999996 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2

$P(t) = 3.33276084 \times 10^{-4}$

$$P(t) = 1.33617366e - 5$$

30	5.00000000e+1	2.50000000e+3
29	5.00000000e+1	2.50000000e+3
28	5.00000000e+1	2.50000000e+3
27	5.00000000e+1	2.50000000e+3
26	5.00000000e+1	2.50000000e+3
25	5.00000000e+1	2.50000000e+3
24	5.00000000e+1	2.50000000e+3
23	5.00000000e+1	2.50000000e+3
22	5.00000000e+1	2.50000000e+3
21	5.00000000e+1	2.50000000e+3
20	5.00000000e+1	2.50000000e+3
19	5.00000000e+1	2.50000000e+3
18	5.00000000e+1	2.50000000e+3
17	5.00000000e+1	2.50000000e+3
16	5.00000000e+1	2.50000000e+3
15	5.00000000e+1	2.50000000e+3
14	5.00000000e+1	2.50000000e+3
13	5.00000000e+1	2.50000000e+3
12	5.00000000e+1	2.50000000e+3
11	5.00000000e+1	2.50000000e+3
10	5.00000000e+1	2.50000000e+3
9	5.00000000e+1	2.50000000e+3
8	5.00000000e+1	2.50000000e+3
7	5.00000000e+1	2.50000000e+3
6	5.00000000e+1	2.50000000e+3
5	4.99999999e+1	2.50000000e+3
4	4.99999999e+1	2.50000000e+3
3	4.99999999e+1	2.50000000e+3
2	4.99999998e+1	2.50000000e+3
1	0.00000000e+0	2.50000000e+3

$$b = 5.0e + 1$$

$$q(t+1)$$

$$v(t)$$



	$g(t t-1)$	$V(t)$
$b = 1_{10} + 0$		
$t = 1$	$0.00000000_{10} + 0$	$1.00000000_{10} + 0$
$t = 2$	$9.99990000_{10} - 1$	$1.00000000_{10} + 0$
3	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
4	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
5	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
6	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
7	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
8	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
9	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
10	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
11	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
12	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
13	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
14	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
15	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
16	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
17	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
18	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
19	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
20	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
21	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
22	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
23	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
24	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
25	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
26	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
27	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
28	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
29	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$
30	$1.00000000_{10} + 0$	$1.00000000_{10} + 0$

$$P(t) = \begin{matrix} 9.99990001_{10} - 1 & -9.99989655_{10} - 1 \\ -9.99989655_{10} - 1 & 1.03447203_{10} + 0 \end{matrix}$$

	$\ddot{y}(t(t-1))$	$V(t)$
$b = 1.10^+ 1$		
$t = 1$	0.000000000 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
$t = 2$	9.999999900 <sub>10</sub> + 0	1.000000000 <sub>10</sub> + 2
$t = 3$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 4$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 5$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 6$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 7$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 8$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 9$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 10$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 11$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 12$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 13$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 14$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 15$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 16$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 17$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 18$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 19$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 20$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 21$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 22$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 23$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 24$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 25$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 26$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 27$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 28$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 29$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2
$t = 30$	1.000000000 <sub>10</sub> + 1	1.000000000 <sub>10</sub> + 2

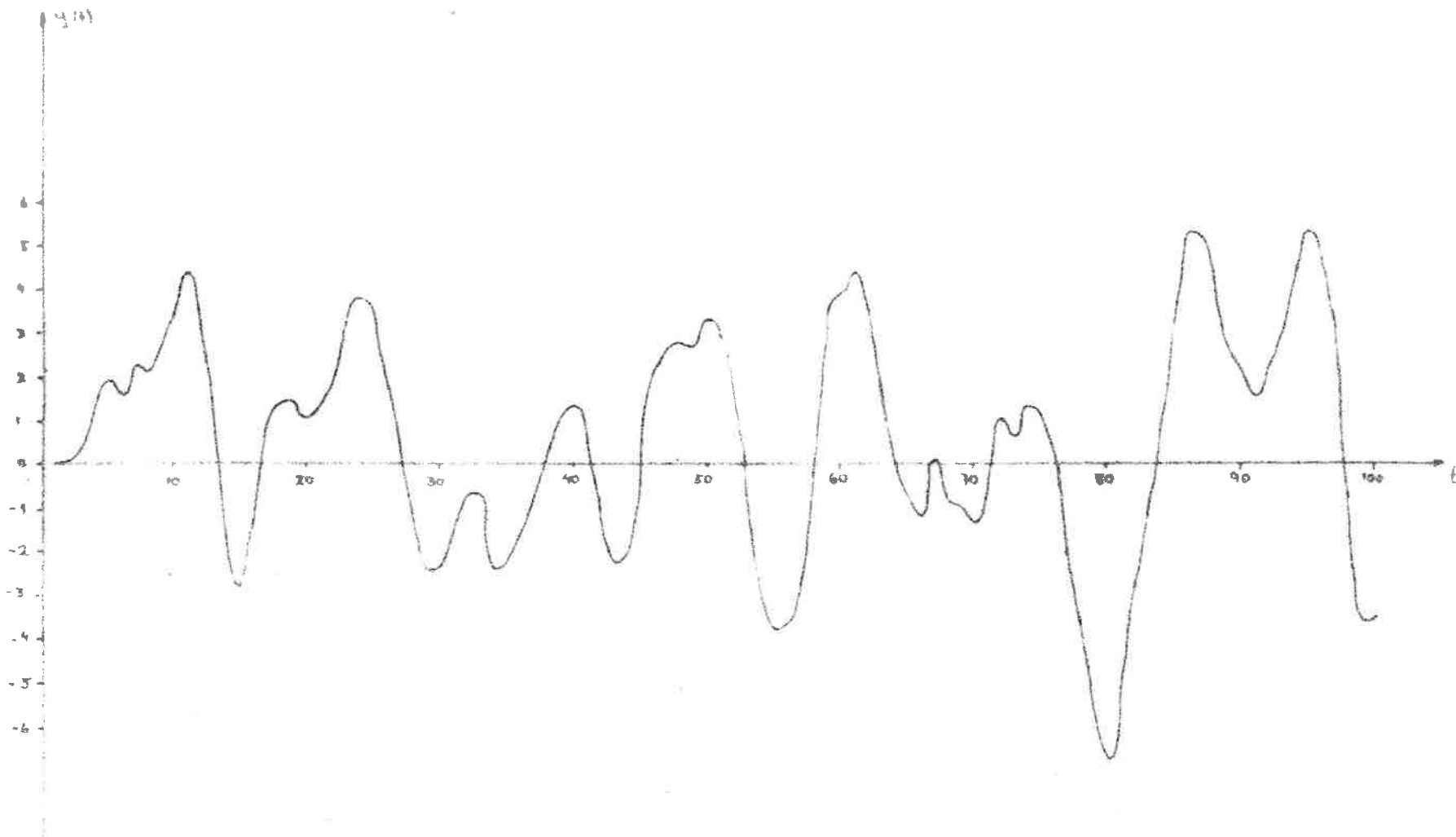
$$P(t) = \begin{matrix} 9.94872948_{10} - 3 & -9.94872943_{10} - 3 \\ -9.94872942_{10} - 3 & 1.02935686_{10} - 2 \end{matrix}$$

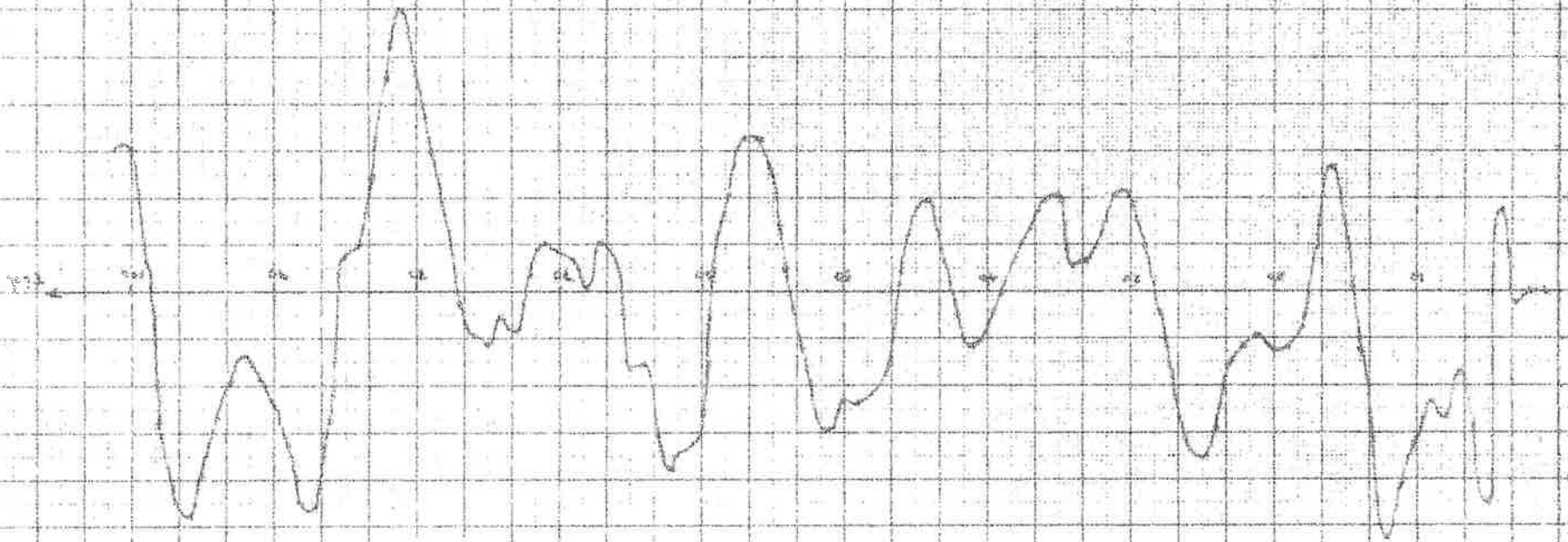
	$g(t t-1)$	$v(t)$
$b = 5_{10} + 1$		
$t = 1$	$0.00000000_{10} + 0$	$2.50000000_{10} + 3$
$t = 2$	$4.99999998_{10} + 1$	$2.50000000_{10} + 3$
$t = 3$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 4$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 5$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 6$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 7$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 8$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 9$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 10$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 11$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 12$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 13$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 14$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 15$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 16$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 17$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 18$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 19$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 20$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 21$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 22$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 23$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 24$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 25$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 26$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 27$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 28$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 29$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$
$t = 30$	$5.00000000_{10} + 1$	$2.50000000_{10} + 3$

$$P(t) = \begin{matrix} 4.27246092_{10} - 4 & -4.27246092_{10} - 4 \\ -4.27246092_{10} - 4 & 4.41069593_{10} - 4 \end{matrix}$$

$$y(t) = 1.5y(t-1) - 0.2y(t-2) + e(t)$$
$$E_y = 0$$

Diagram 1.1  
Est





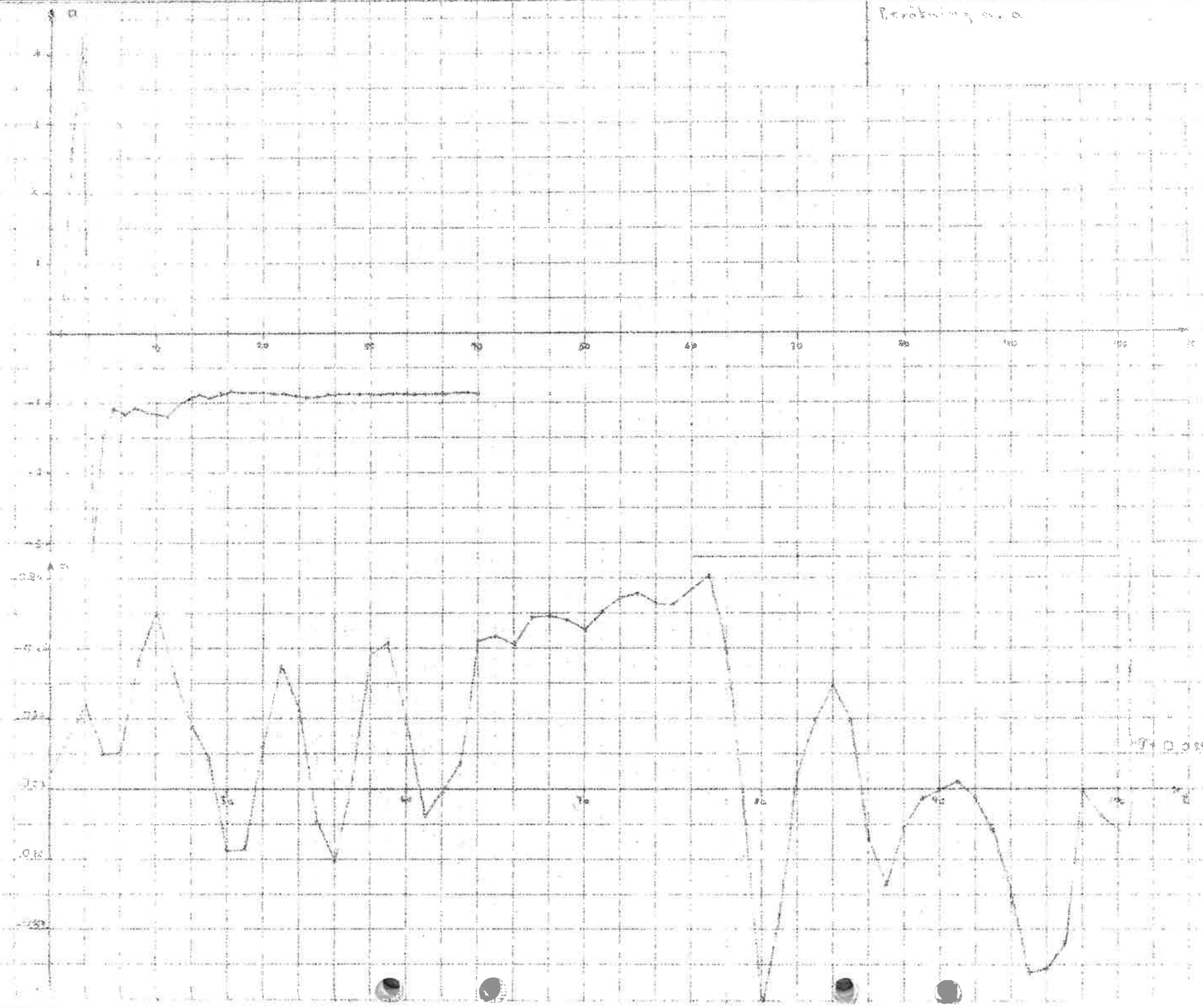
Handwritten text at the bottom right of the page, possibly a date or label.

Diagram 1/4  
 Ex. 8  
 8/2/01

Handwritten notes at the bottom left, including the word "system" and some illegible text.

Berating a. a

Diagram 13  
Ex 1  
20/21 1

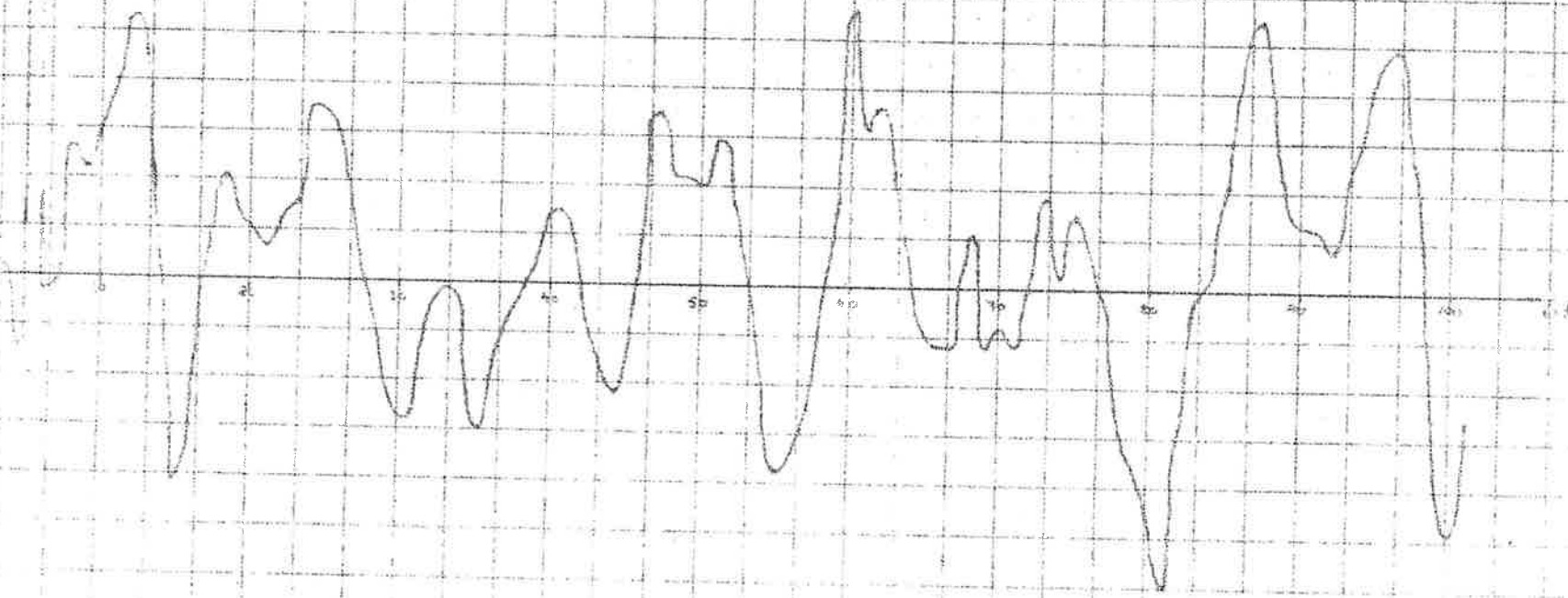


$$u(t) = 100 \sin(2\pi t) \text{ and } y(t) = 100 \cos(2\pi t)$$

Plot the response of the system.

Page No. 1/1
Ex 1
R.F. 1

Figure



Resistor no. 1, 2, 100

Diagram 15

Ex 1

20/02

10

20

30

40

50

60

70

80

90

100

t







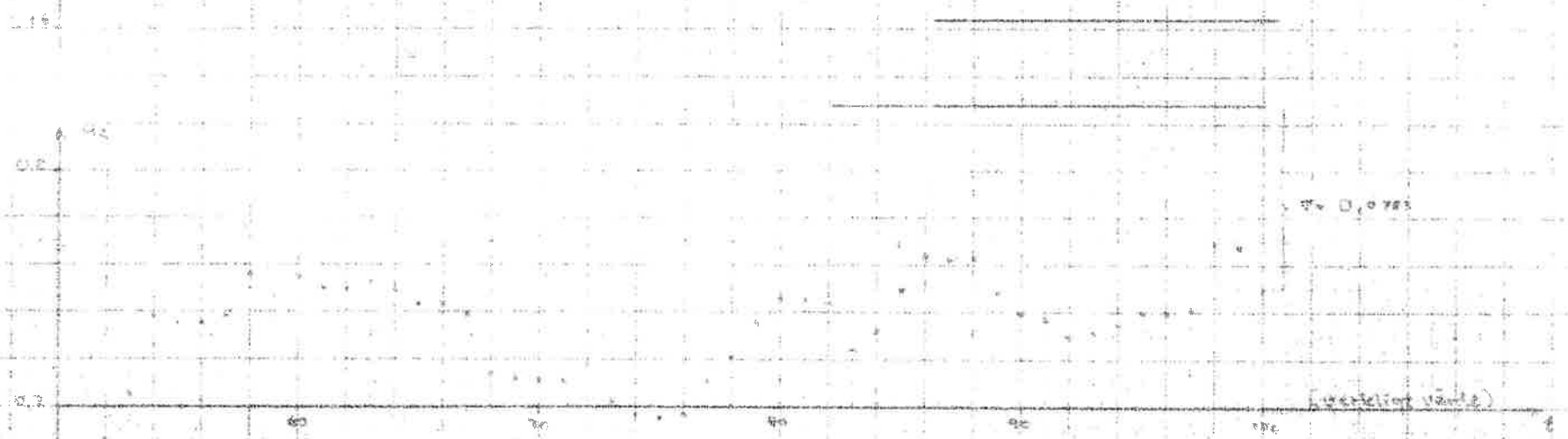
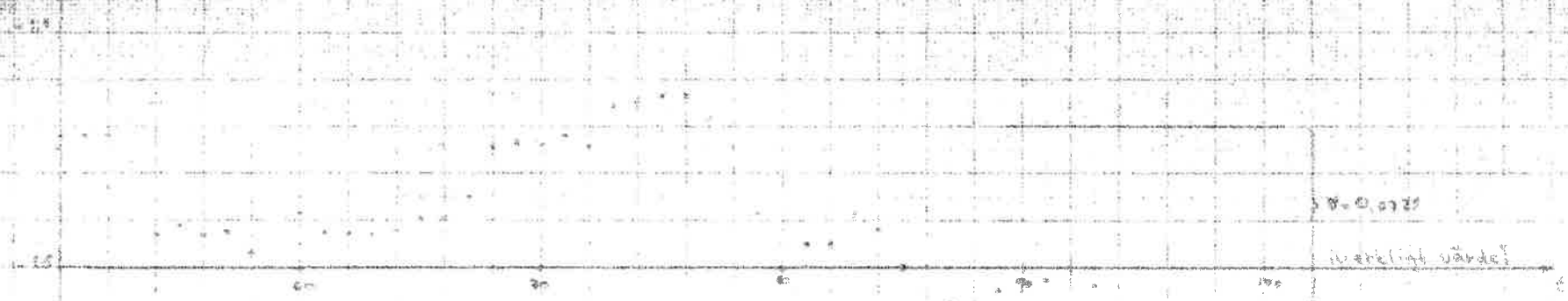
10/87  
 EX 1  
 Diagrams

800-100-20-1-100

100

Bestämning av  $\sigma_1$  och  $\sigma_2$   
t = 50 - 100

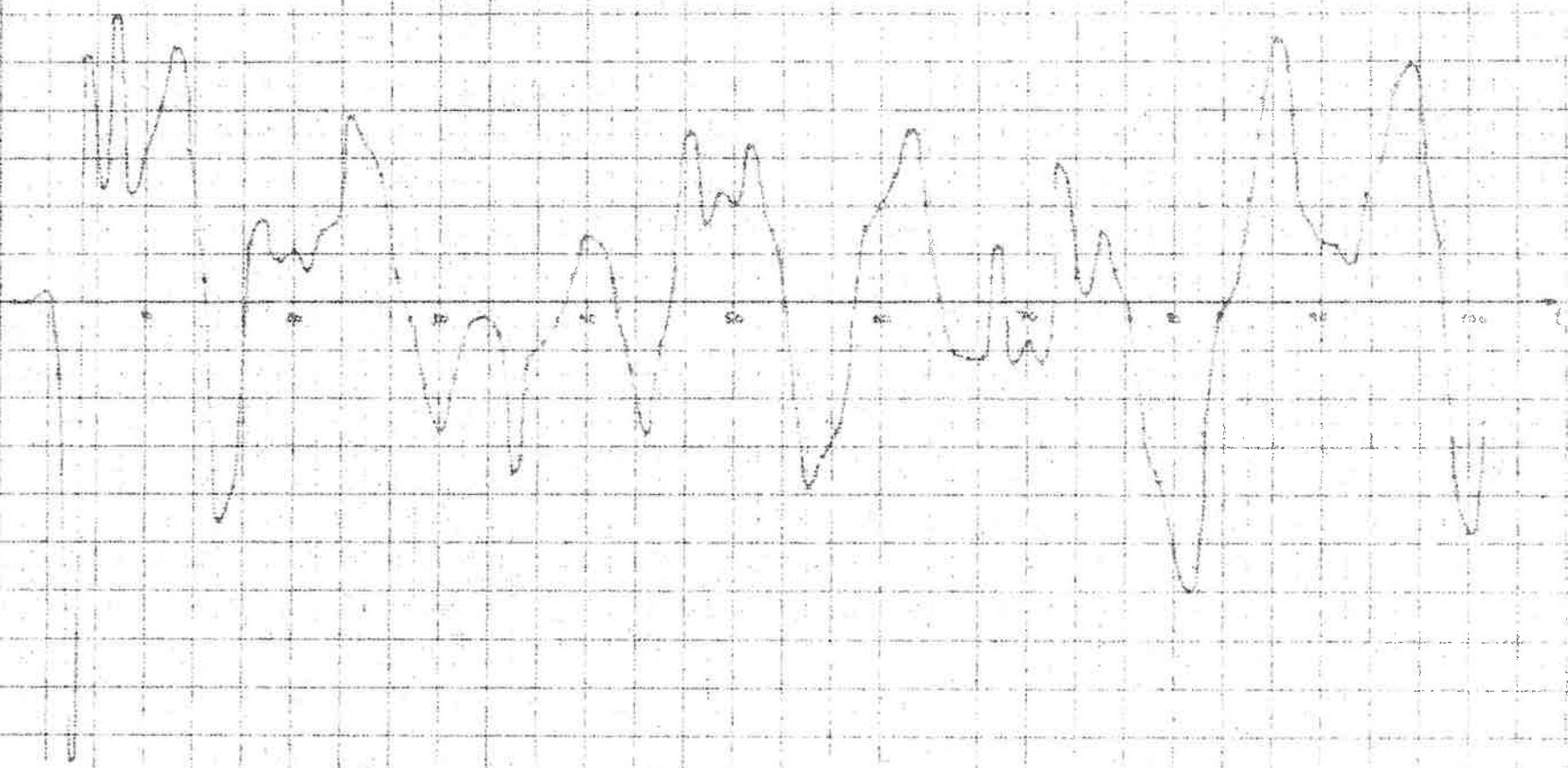
Skala  
R 2/21



4th order of Bessel function  
for  $x = 0$  from the following  
graph

Date	10/10/22
Ex	7
Roll No.	

Y-axis



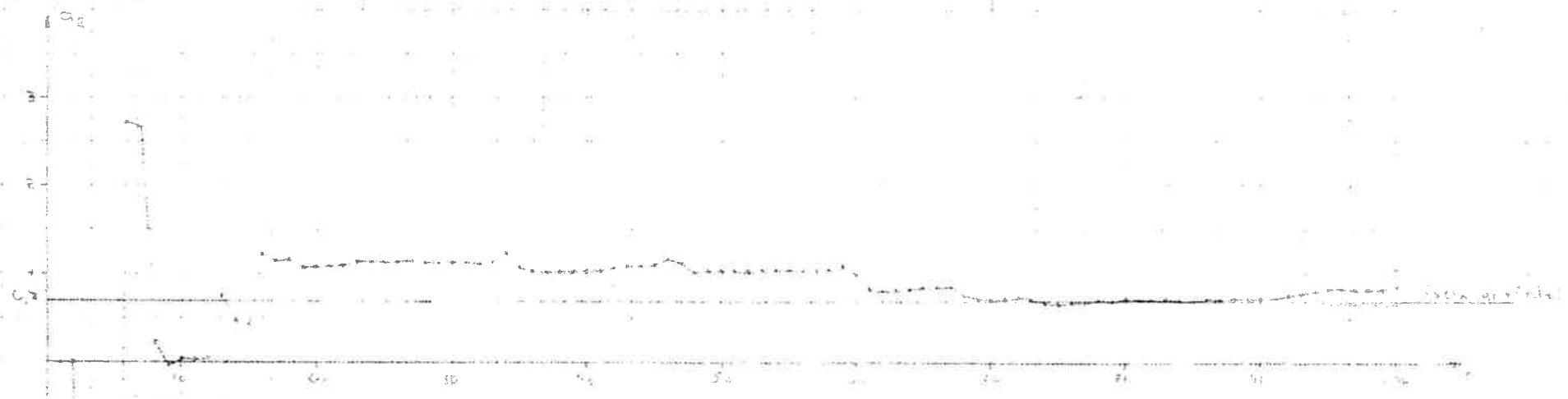
Beräkning av  $\alpha$ .

0.000000
Ex 1
X11/P1.3



Bestätigung

Diagram 1.10  
Ex 1  
R11/913

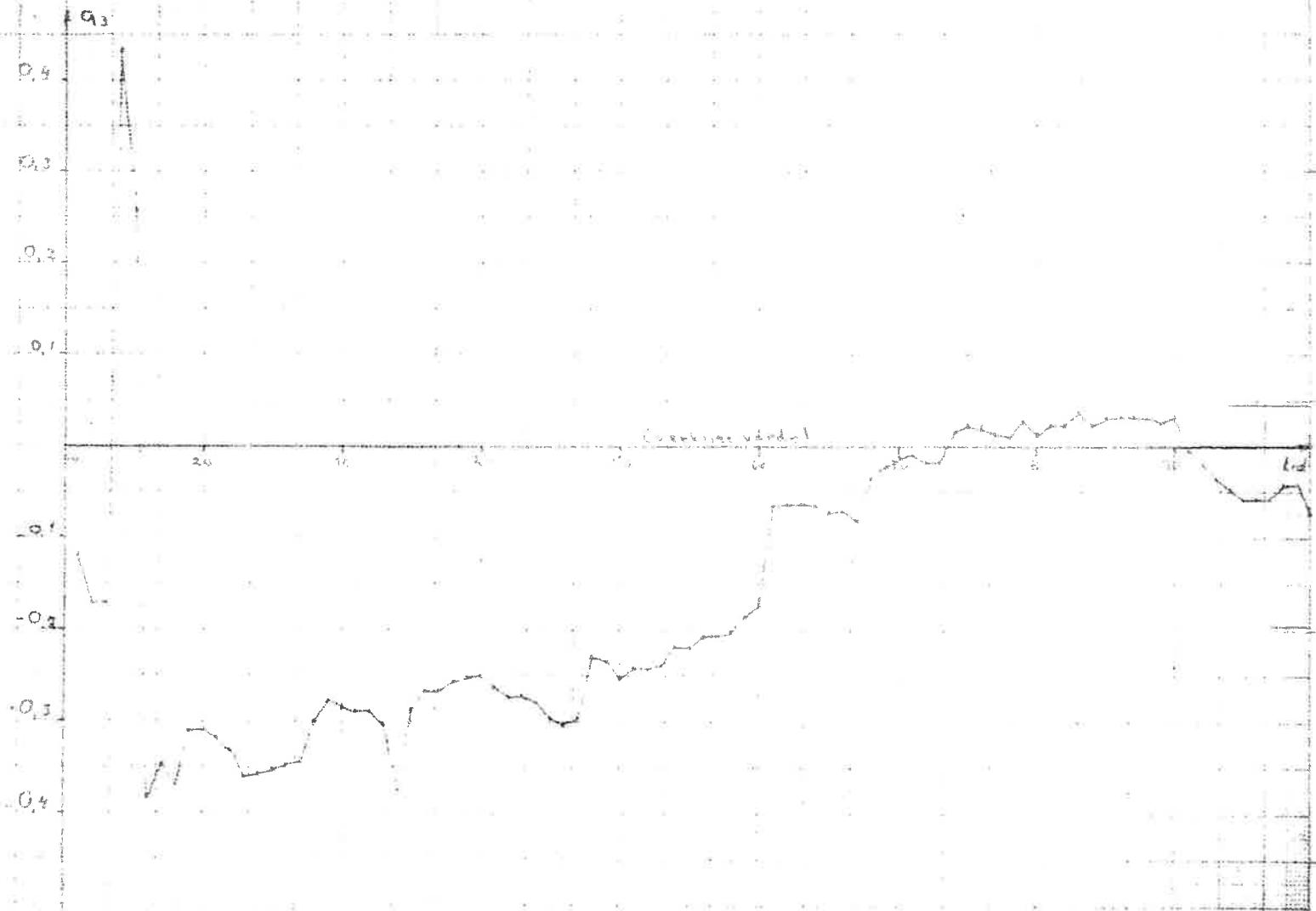
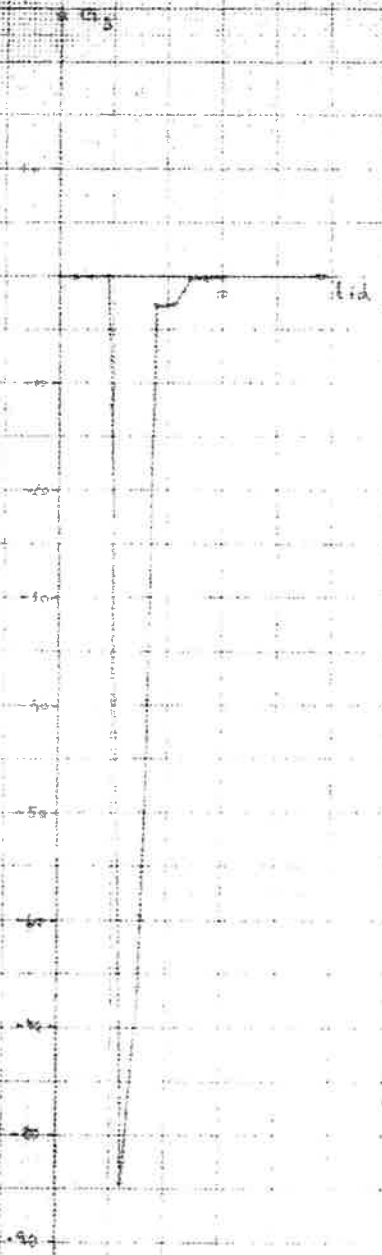


Beräkning av  $a_3$

Diagram III

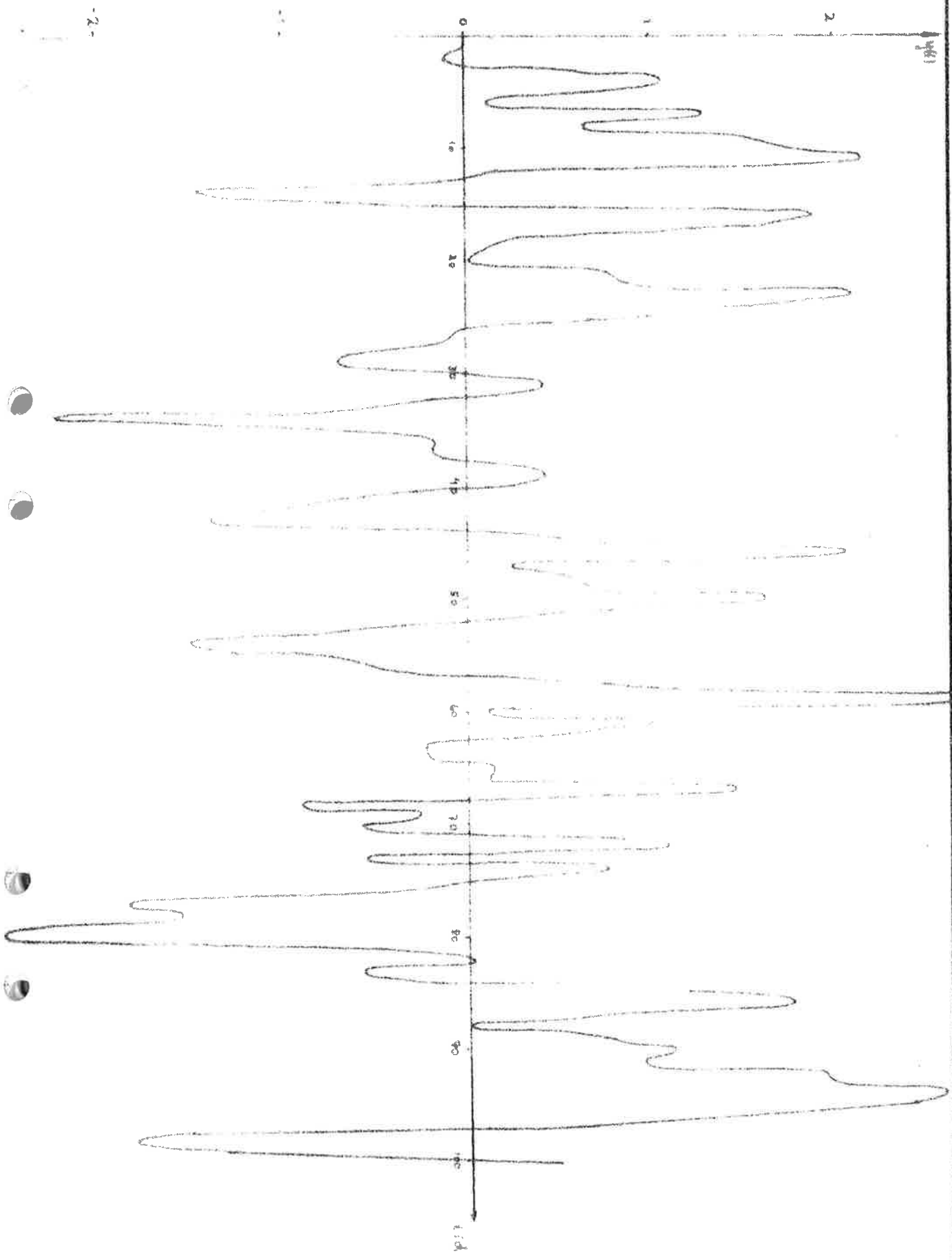
Ex 1

20/01/3



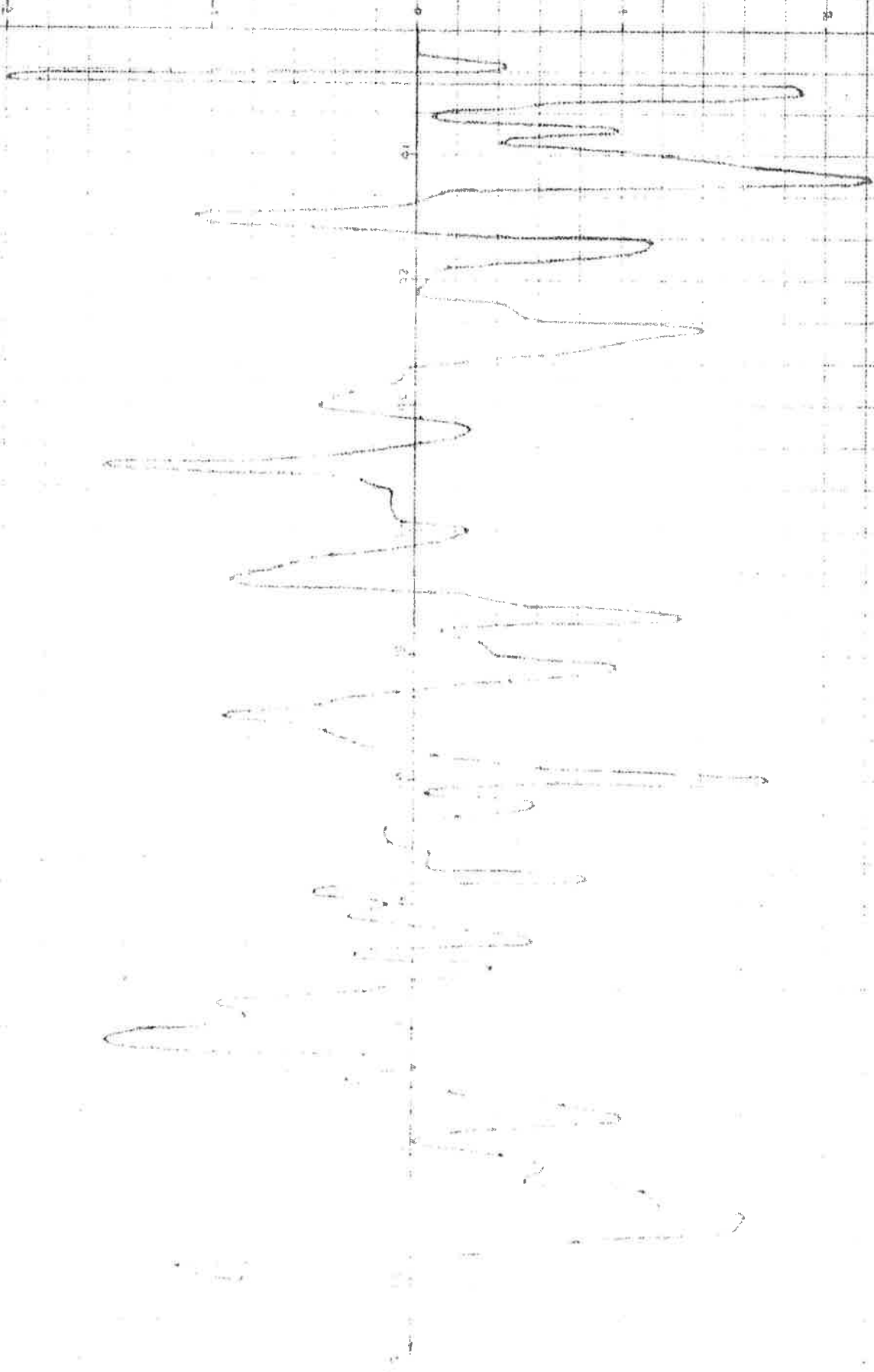
Data genererade av processen  
 $y(t) = 0.6 \sin(t) + e(t)$

Skapades 2.1  
FA2



-2-

204-10

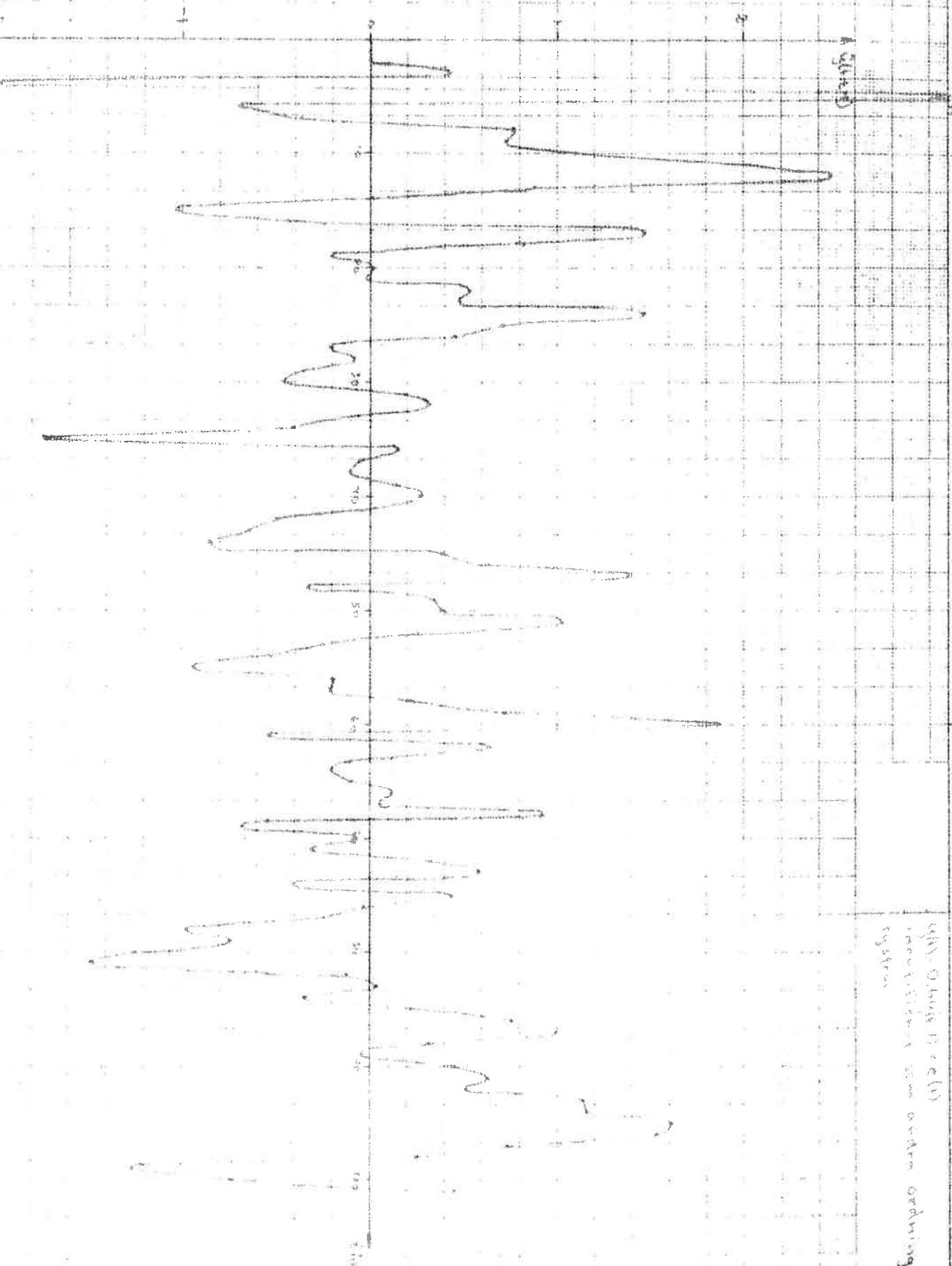


Hand-drawn waveform  
on grid paper

Diagram 2.2  
EX 2  
R01/01/1



A ground



4th Order 1/2 (1)  
non-linear sum order ordingim  
system

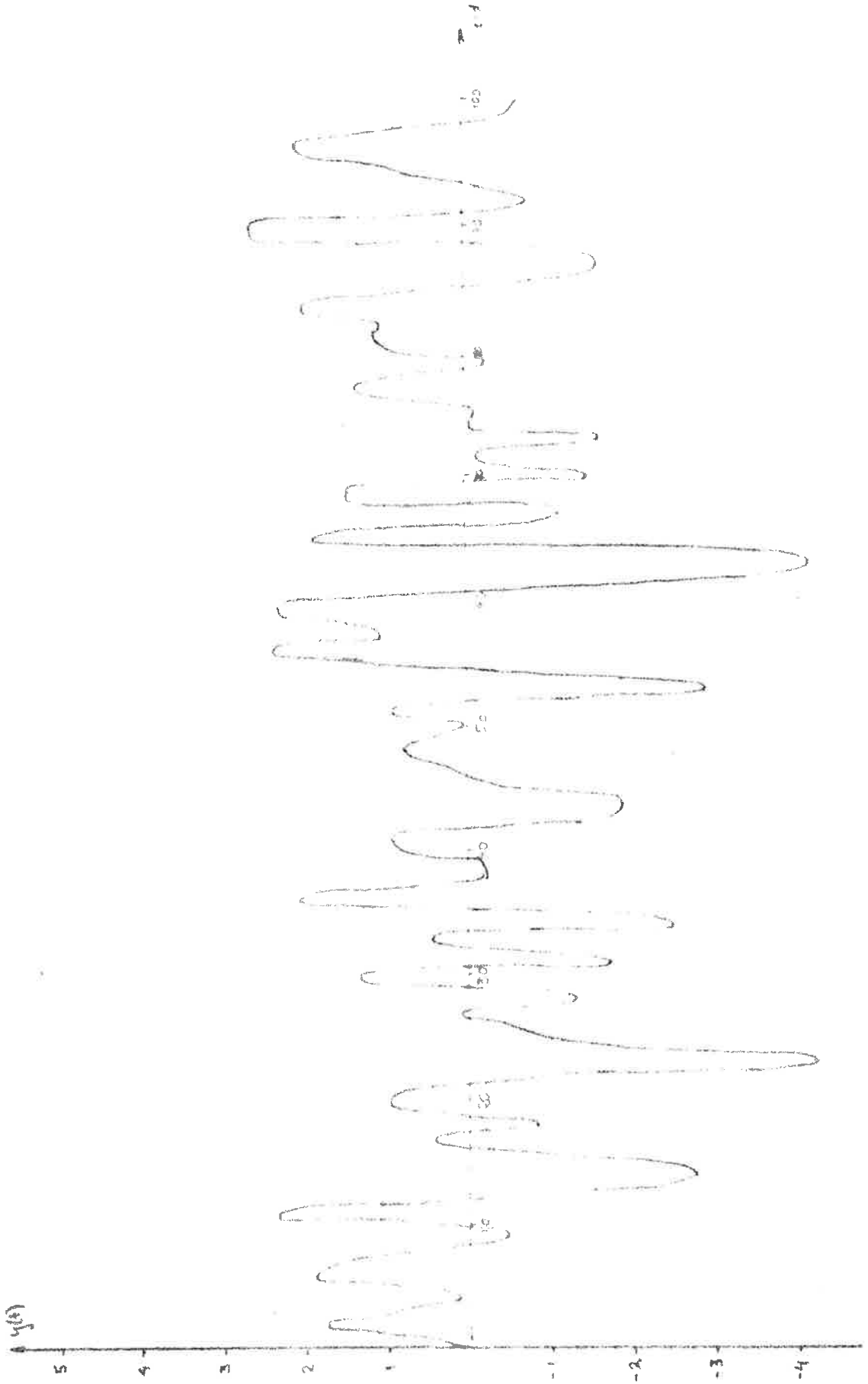
Diagram 2.3  
EX 2  
RS6/P4.2

Diagram 3.1

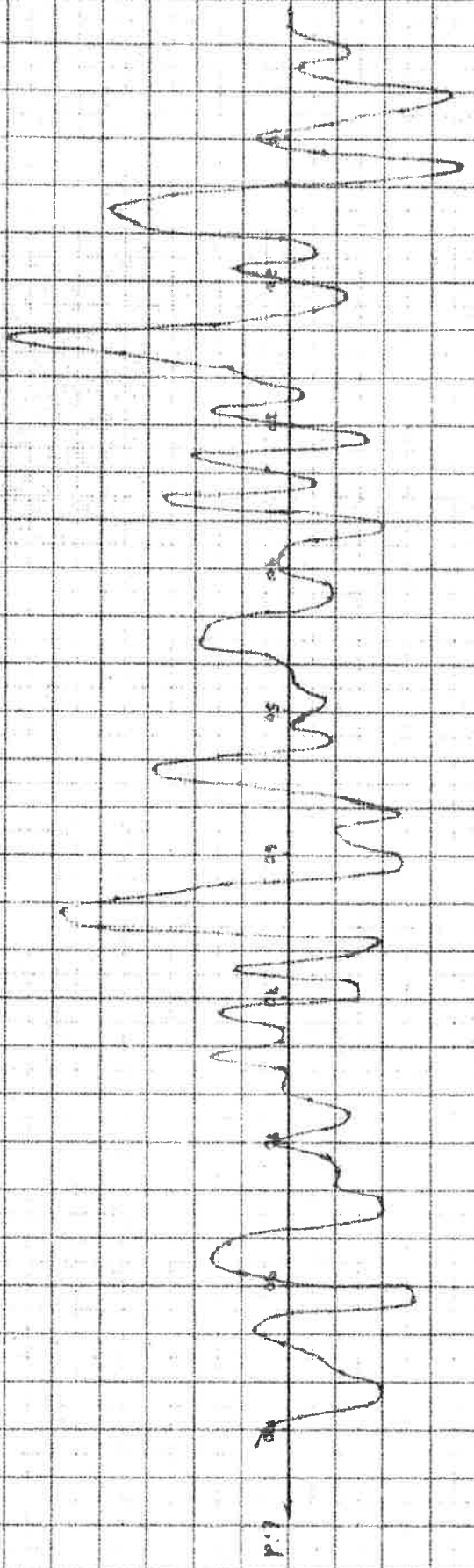
E+5

B16/25

Data generated at Procceser  
mitt. ellipt. Cigarette-17



$y(t) = \sin(t)$



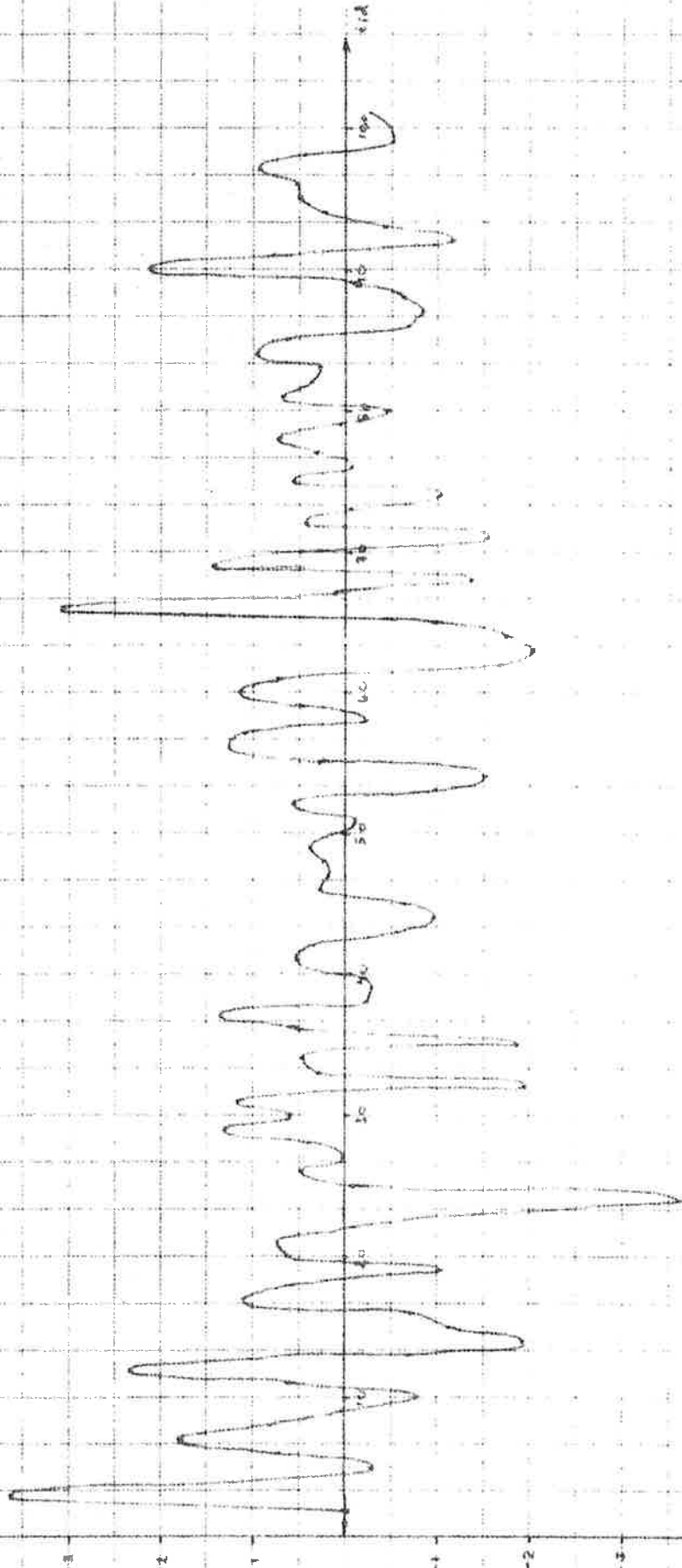
$y(t) = \sin(t)$  and  $x(t) = \cos(t)$  identical  
same in magnitude and phase

Diagram 5-A  
Ex 5  
R21/PH1

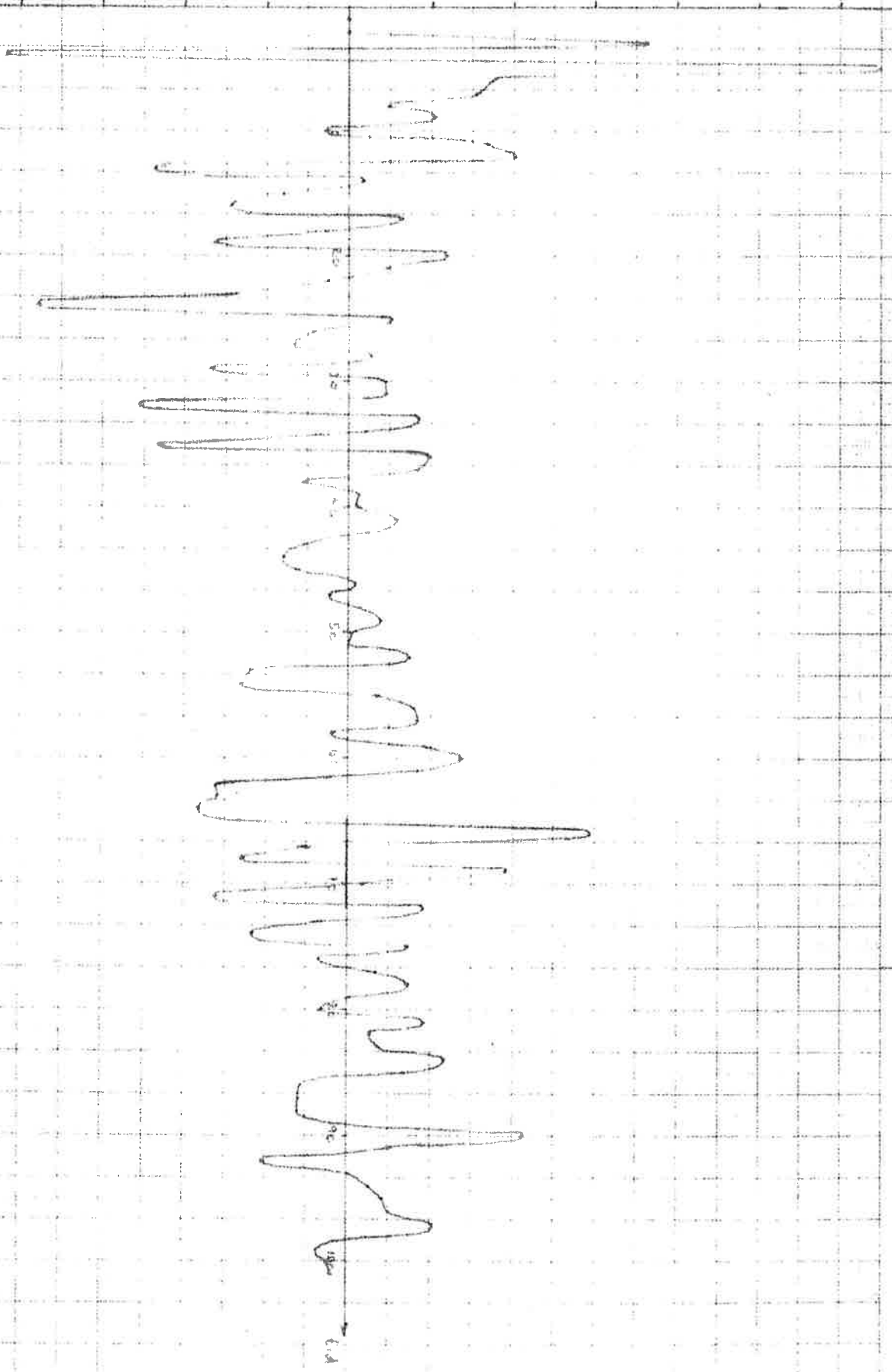
$y(t) = \cos(0.999t)$  identifier system  
som ett 2:a ordningens system

01.5.2008  
EX 3  
R22/P11.2

$y(t)$



$y(t)$



$y(t) = e^{1.1t} \sin(0.99e^{1.1t})$  identifier  
sem 3: e ordingent system

Diagram 5.4

EX 5

23/04/3

Data genererade av processen  
 $y(t) = a(t) \cdot 0.3 a(t-1)$   $t = 1 - 100$

Diagram 6  
Exempel 6  
R34/P7

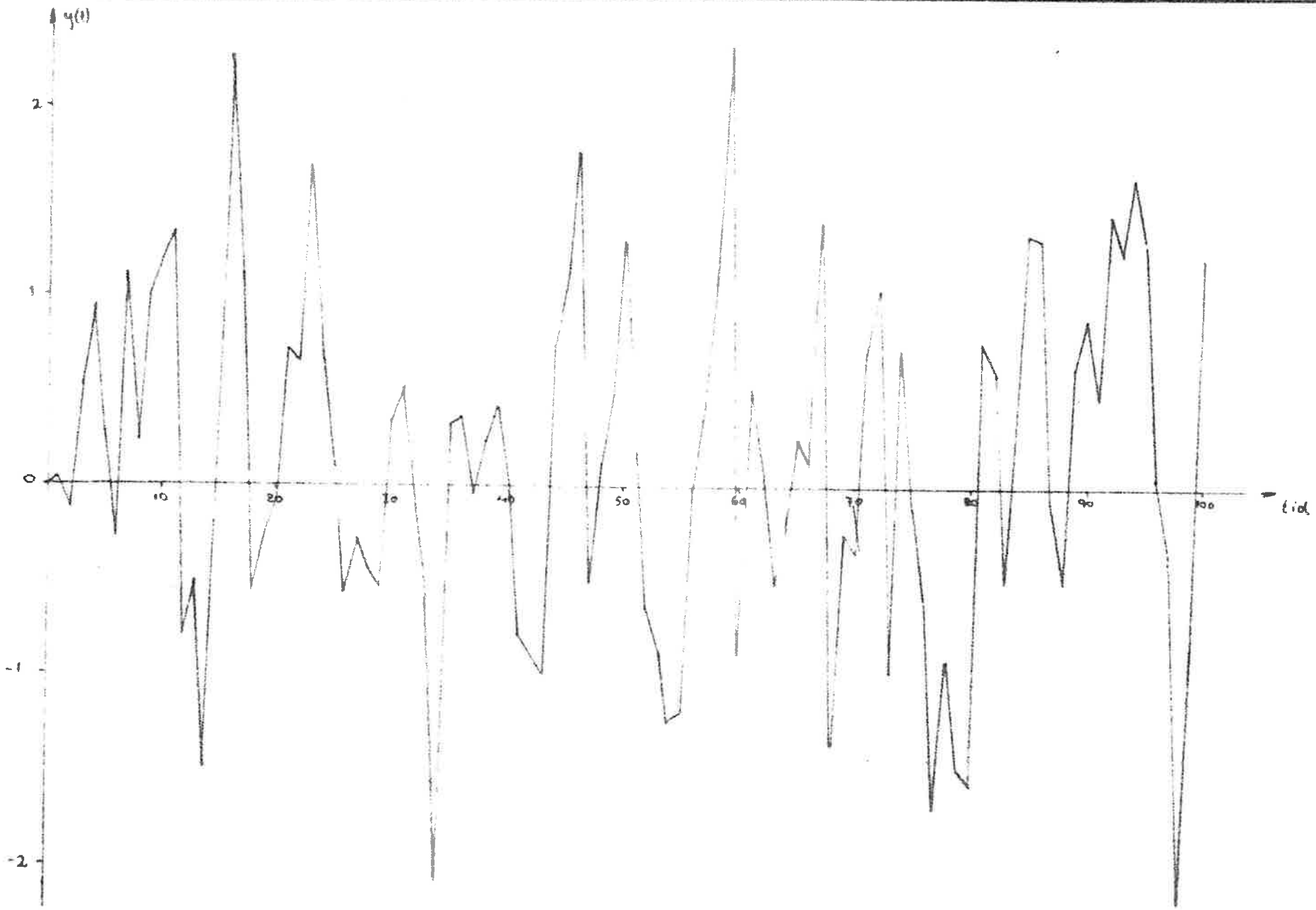
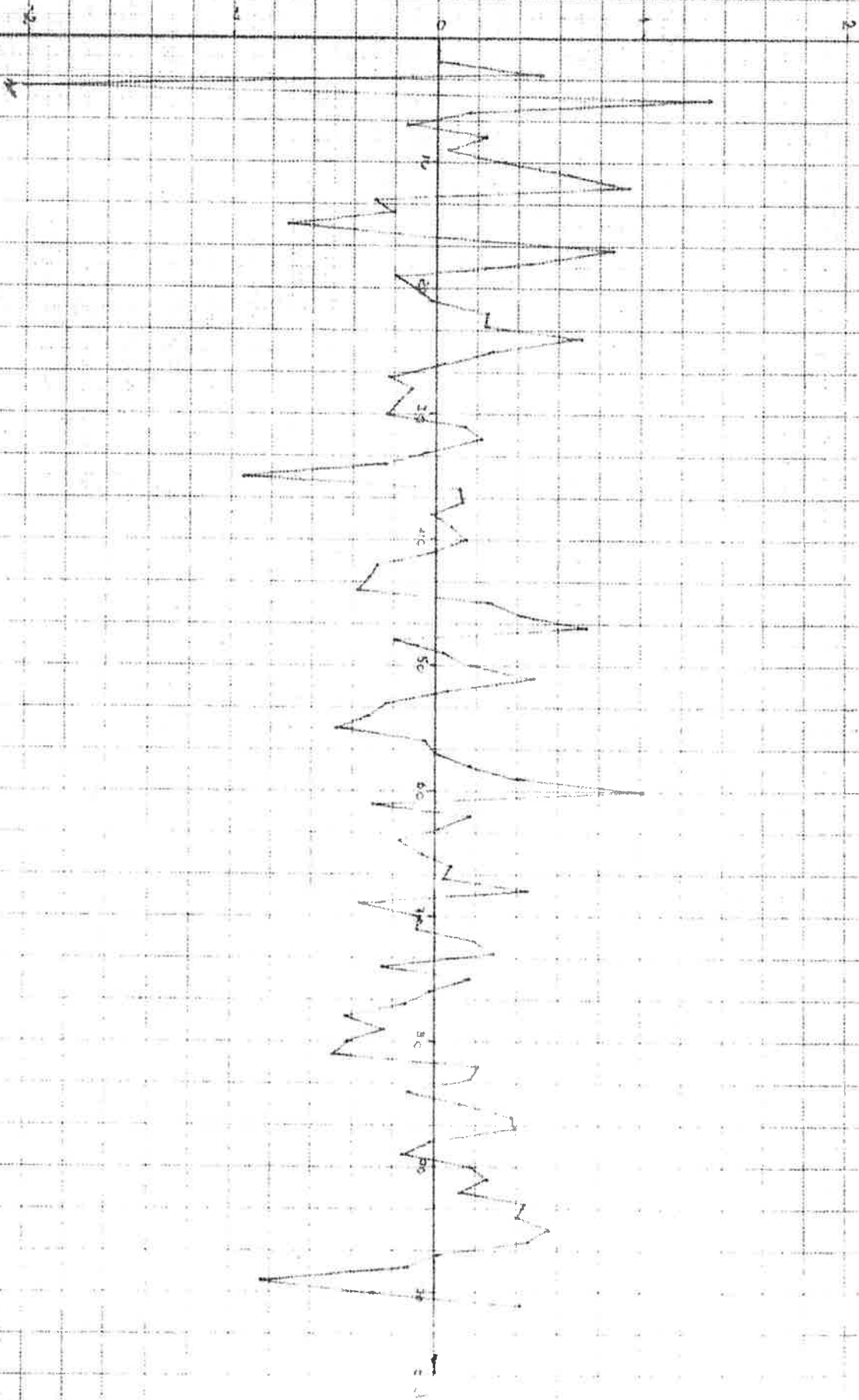


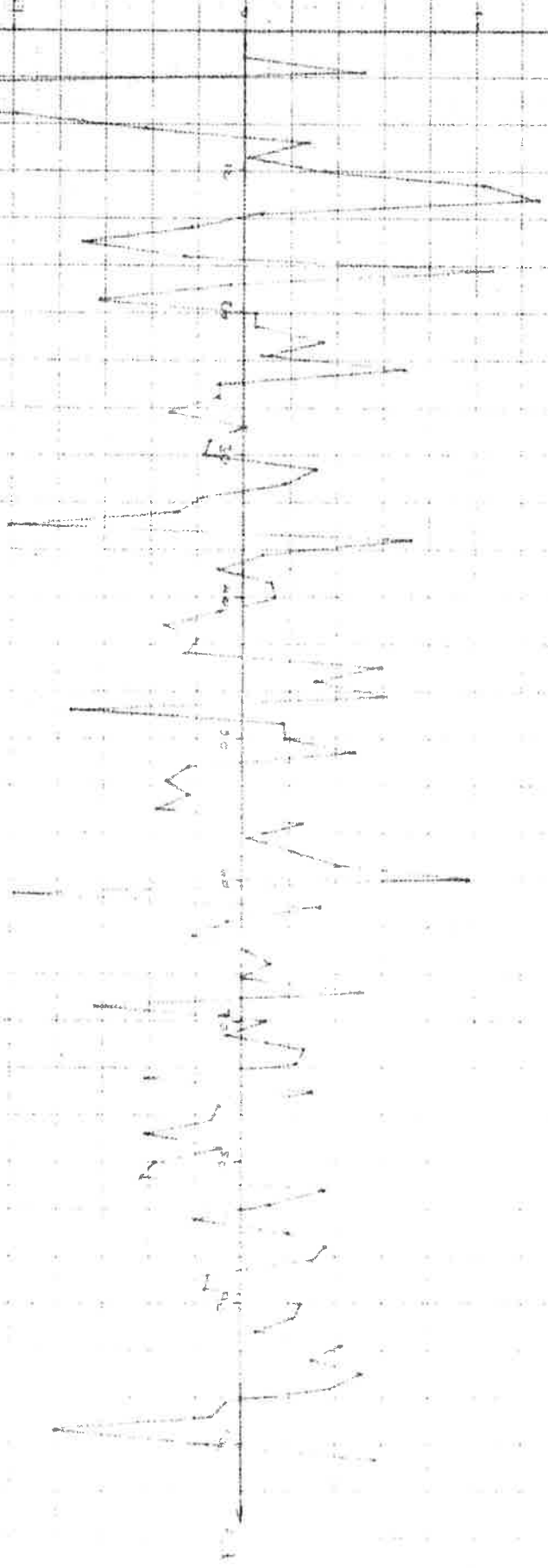
Figure 1



$y(t) = \sin(0.2t)$  identified  
som första ordningens system

Diagnostik 6.2  
Ex 6  
ASX/D11

(SIN-1)



$y(t) = e(t) \cdot 0.5 \sin(10^6 t)$  Identificator  
Som andre siffer i tallet

Diagnose G3  
Ex 6  
23/11.2



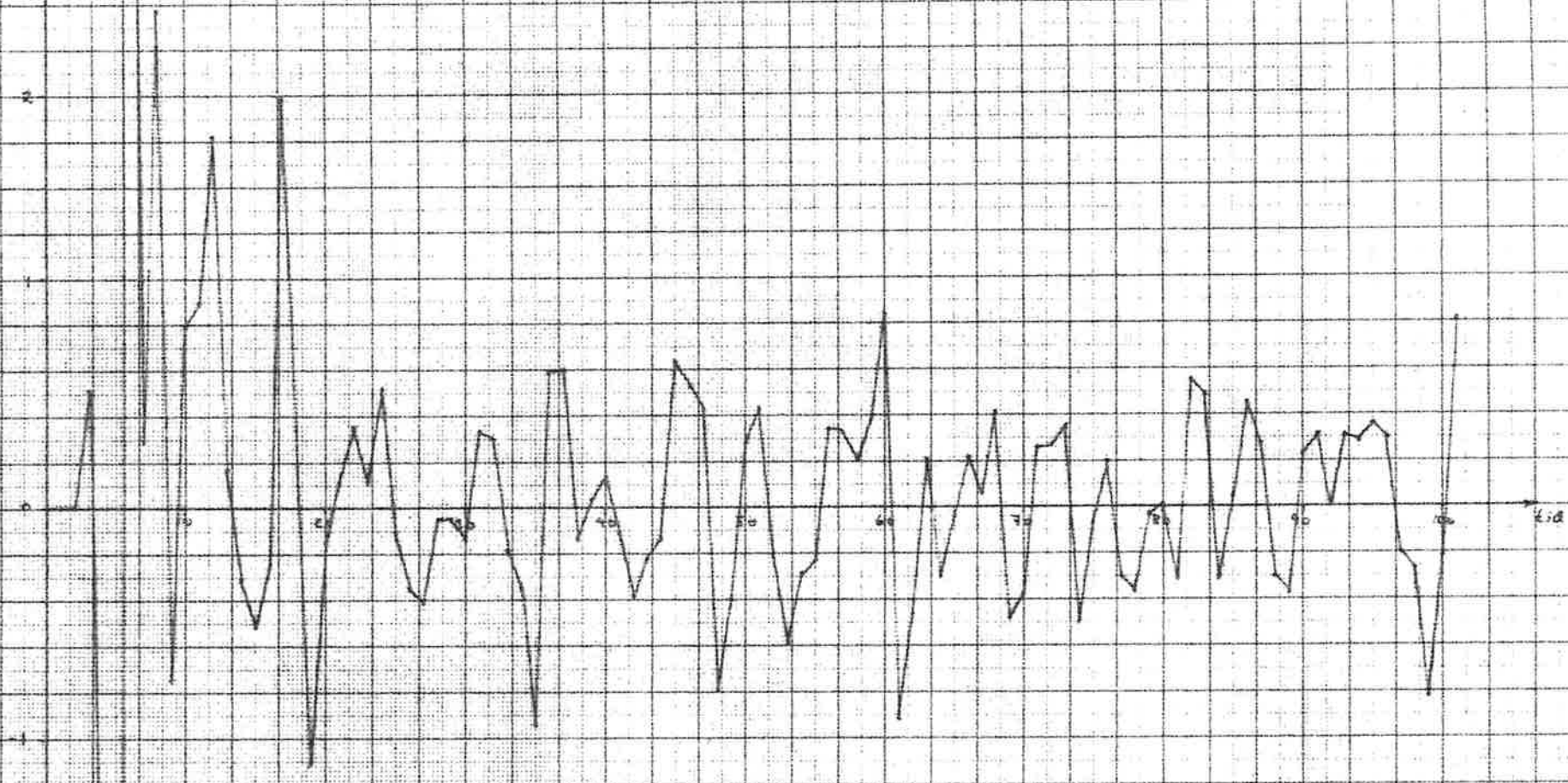
Unit 8 - 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000

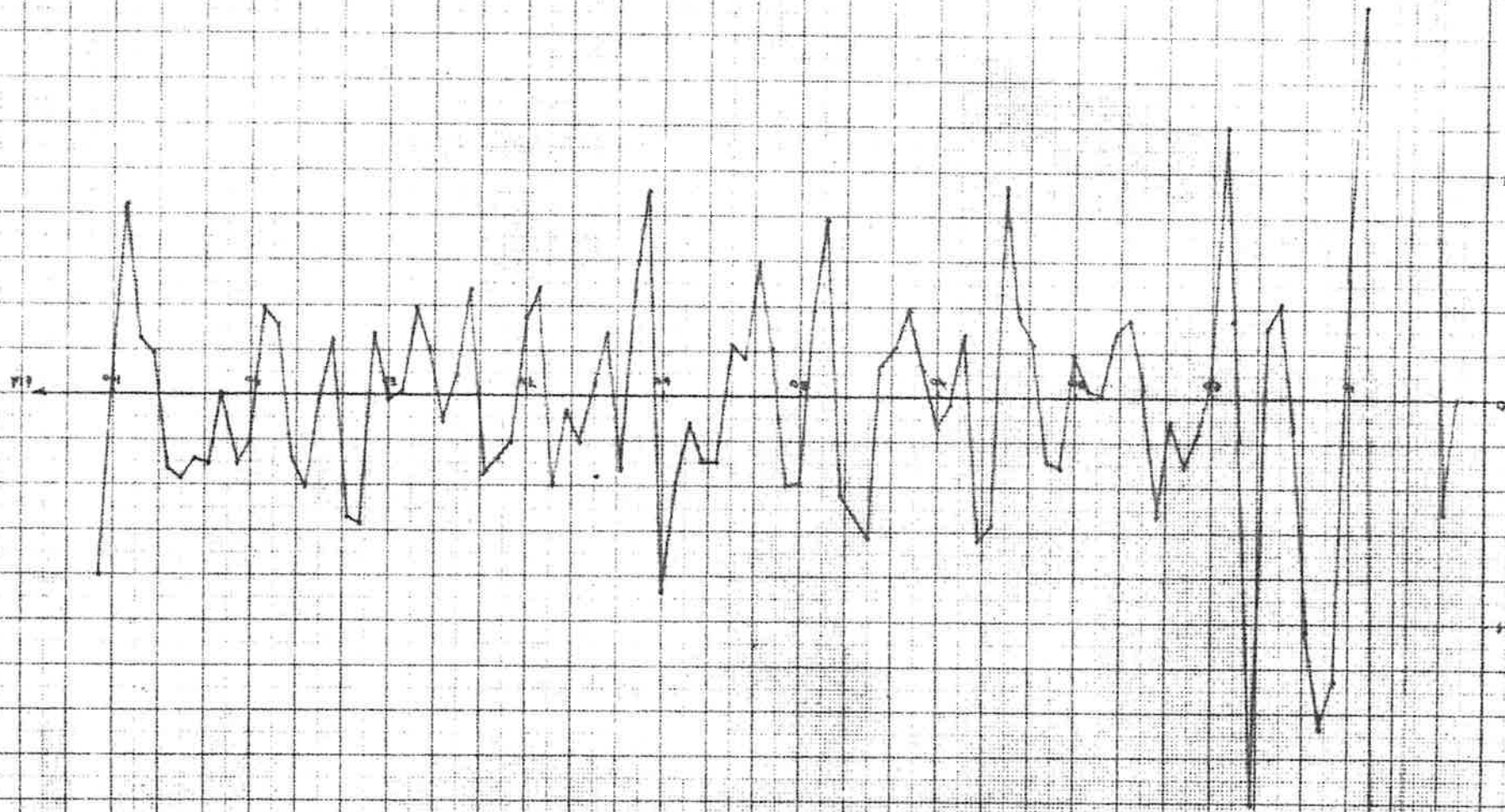
Diagram 6.1

Ex 6

24/01/2

$y(k-1)$





R47/D14  
E6  
Diagram 6.5

identifikation  
 $y(t) = e(t) + 0.2 e(t-1)$   
 som fjerde ordningens system

(147)14

Data genererade av processen

$$y(t) = z_1(t) + z_2(t)$$

$$z_1(t) = z_1(t-1) + e_1(t)$$

$$z_2(t) = 0.5 z_2(t-1) + e_2(t)$$

Diagram 7.1

Ex 7

R33/P5

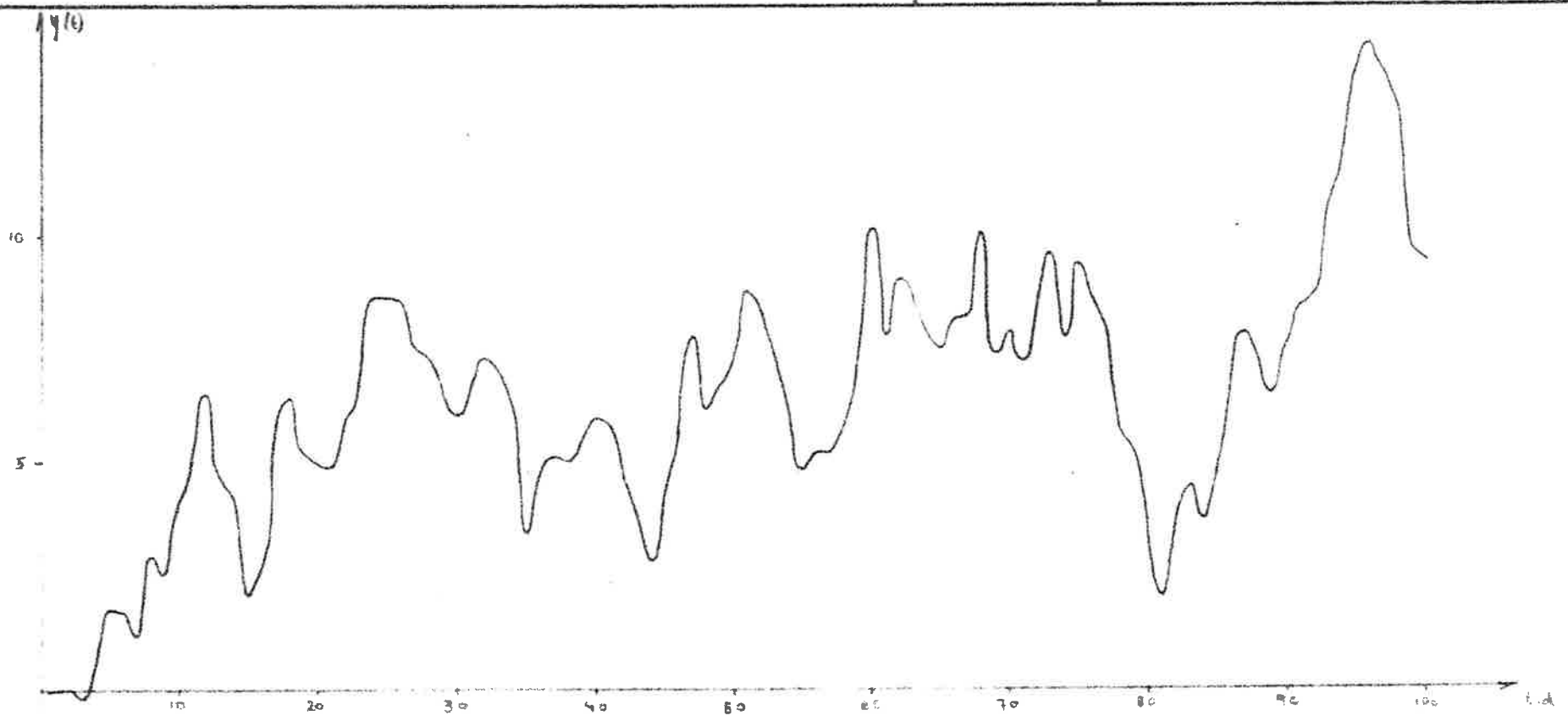


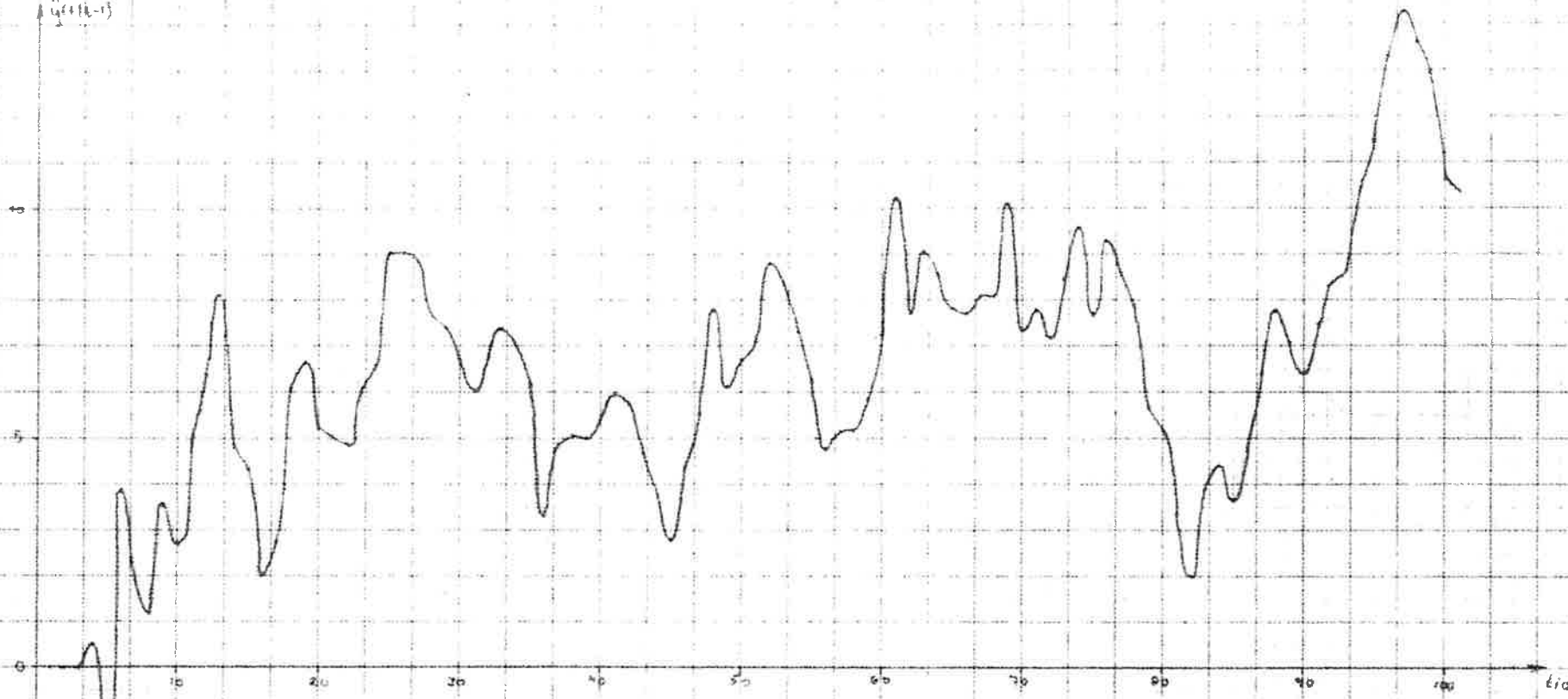
Diagram 7.2

E = 7

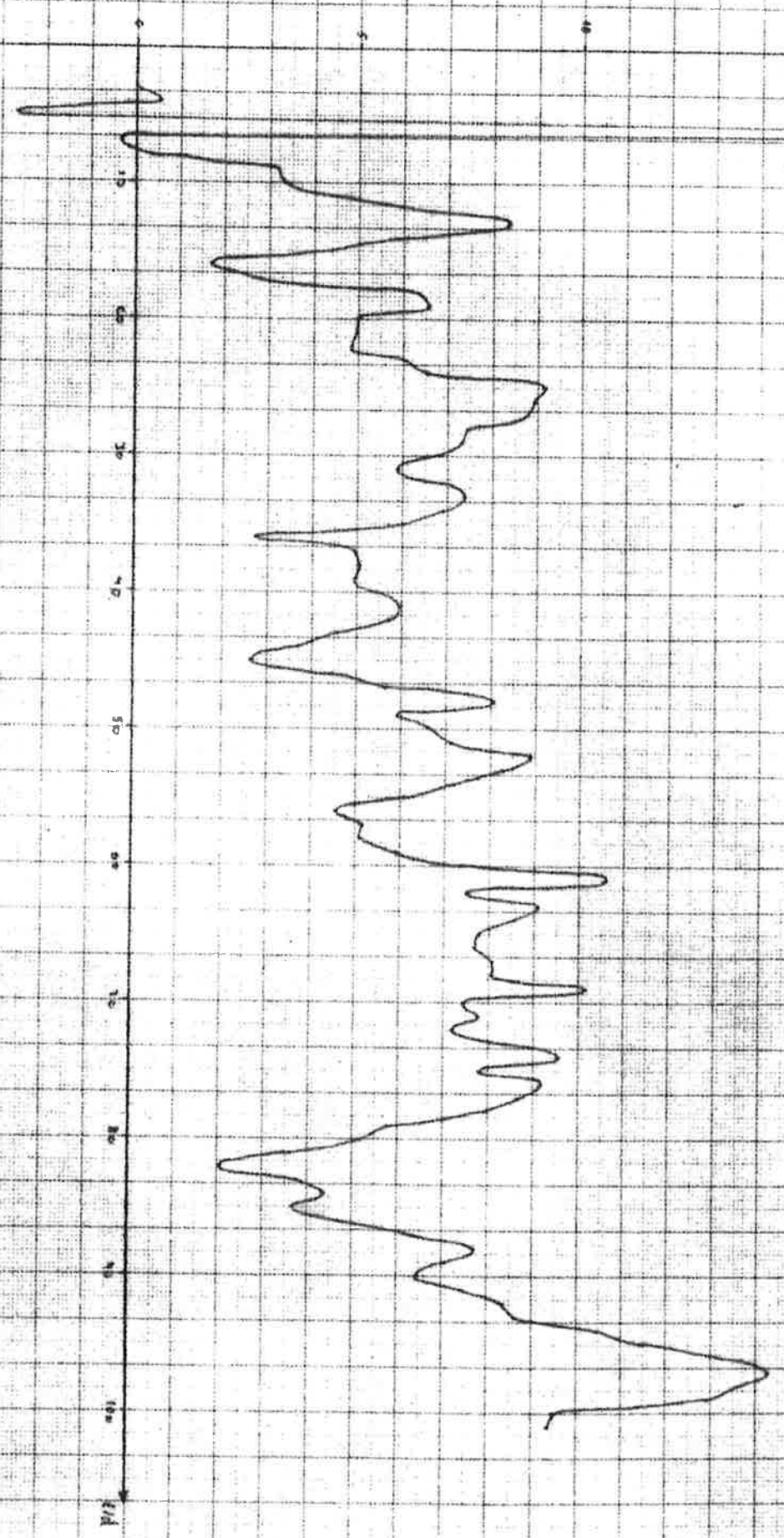
R49/P1.1

$y_1(t) = x_1(t) + x_2(t)$   
 $x_1(t) = x_1(t) \cdot \cos(\omega t)$   
 $x_2(t) = x_2(t) \cdot \cos(\omega t)$   
The two signals have the same frequency.

$y_1(t) = x_1(t) + x_2(t)$



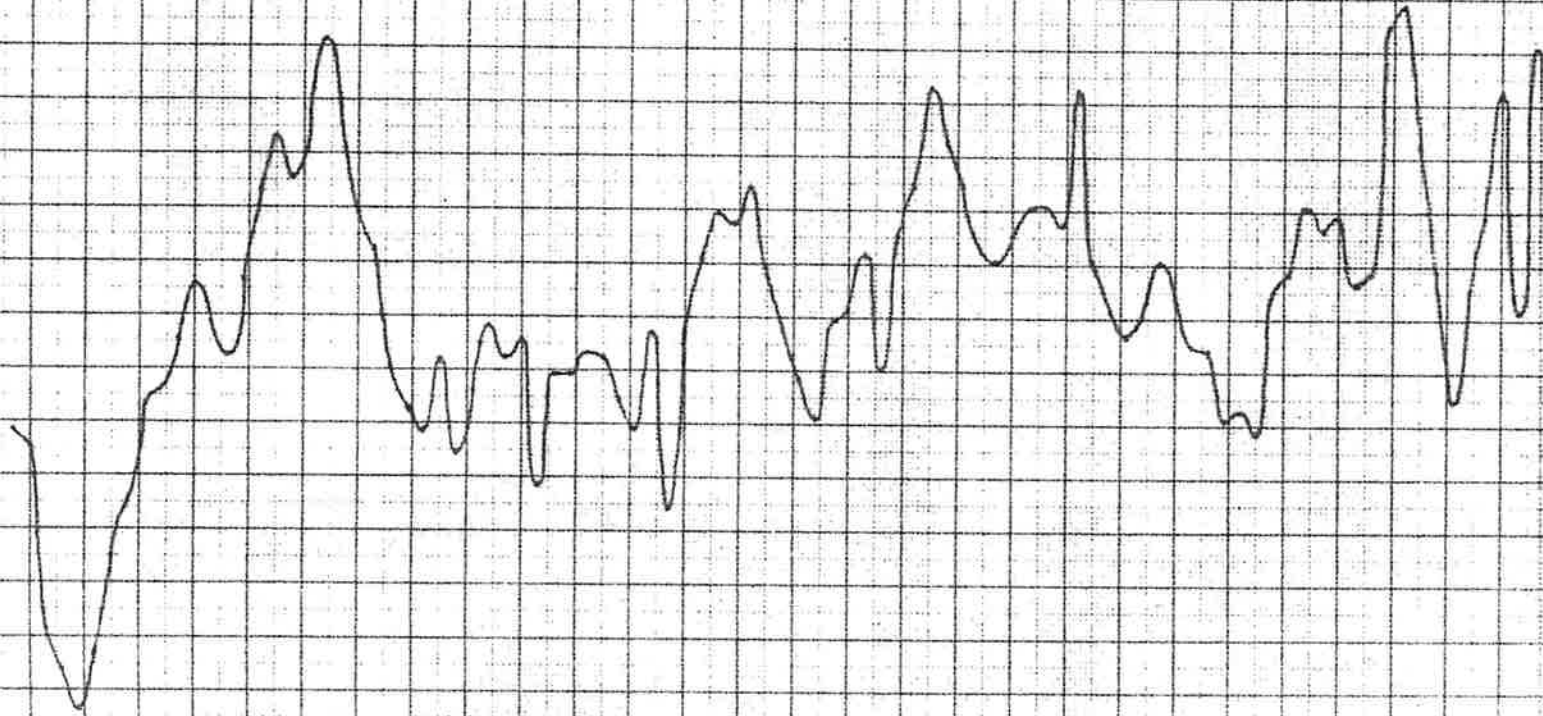
1971-72



$y(t) = 1(t) * z(t)$   
 $x_1(t) = x_1(t) * e^{(t)}$   
 $x_2(t) = 0.8z_1(t) * e^{(t)}$   
Identifying some order parameters  
system

Diagram 7.3  
Ex 7  
R50/P112

10 20 30 40 50 60 70 80 90 100 End



$y(t) = \dots$

$$y(t) = x_1(t) + x_2(t)$$

$$x_1(t) = x(t-1) + e(t)$$

$$x_2(t) = 0.8x_1(t-1) + e_2(t)$$

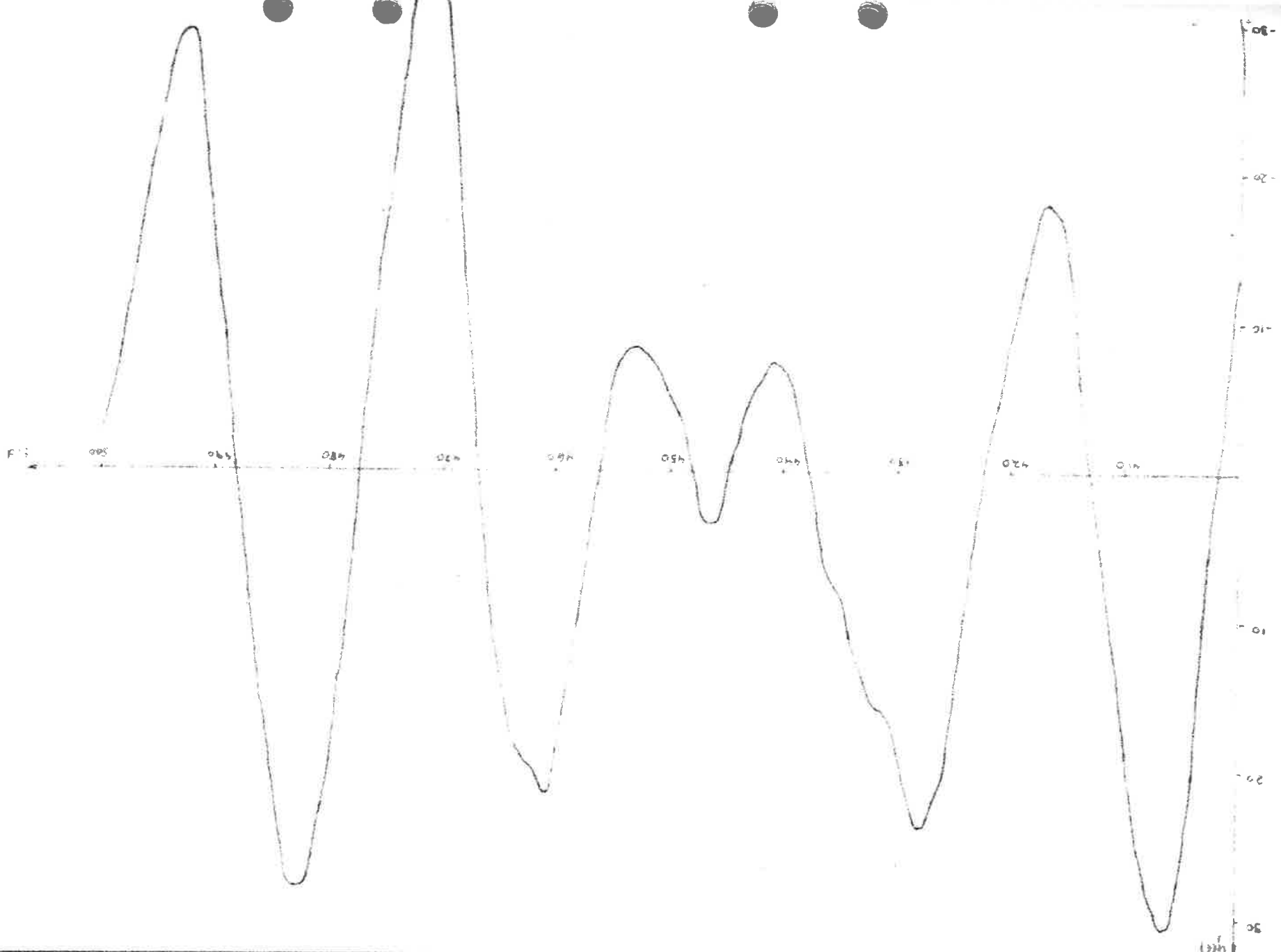
Identifikation von Feedbacksystemen

System

Diagram 4.1

Fx

RSS/D1:3



Data generated by Procsen

$$x_1(t) = 1.90 x_1(t-1) - 0.0425 x_1(t-2) + e_1(t)$$

$$x_2(t) = 1.50 x_2(t-1) - 0.1 x_2(t-2) + e_2(t)$$

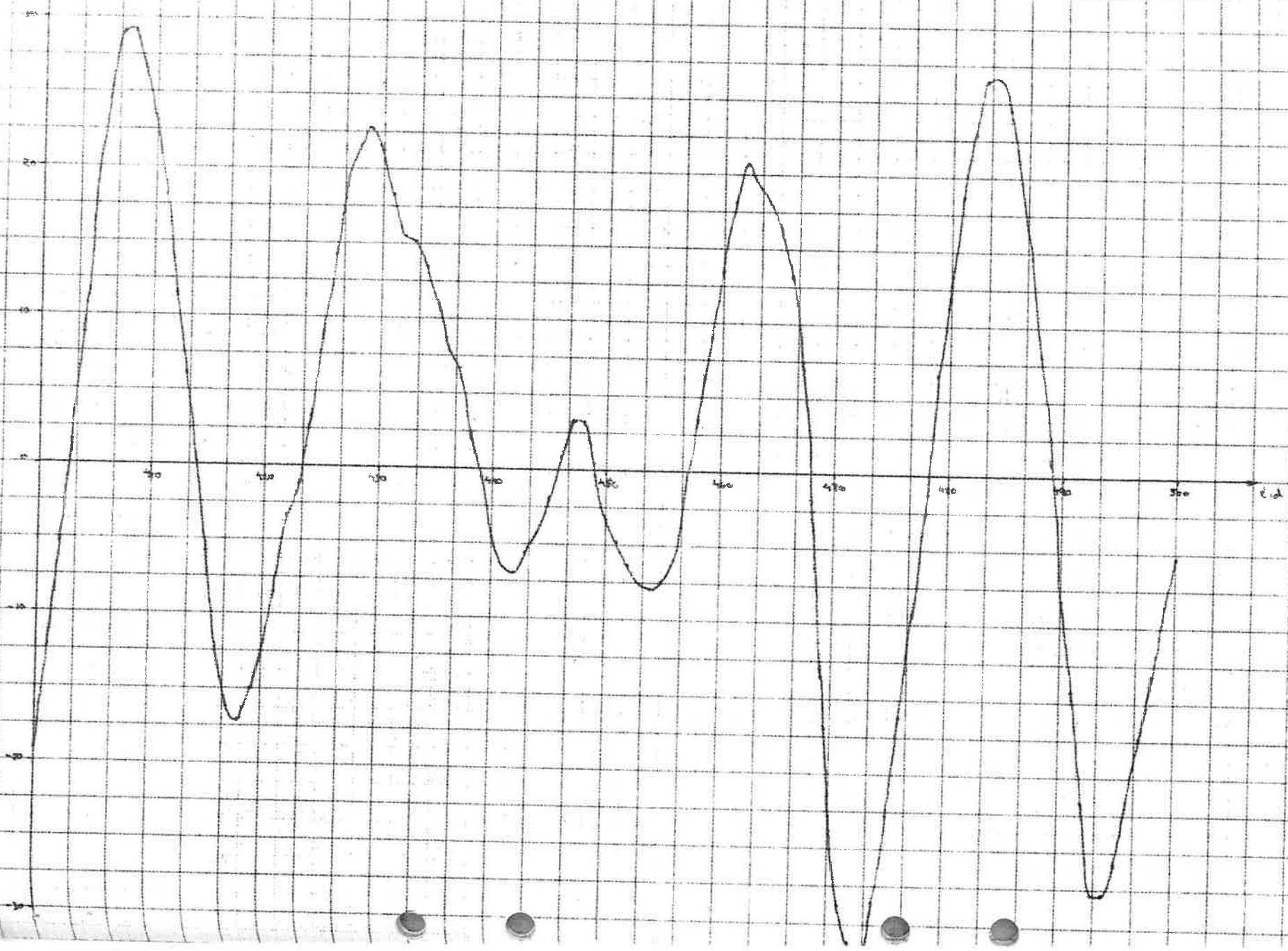
$$y(t) = x_1(t) + x_2(t) \quad t = 400 - 800$$

Diagram 8.1  
E 8  
8.17/16

$y(t) = x_1(t) + x_2(t)$   
 $x_1(t) = \cos(x_1(t)) \rightarrow \max(x_1(t) - 2) < 0, t >$   
 $x_2(t) = \cos(x_2(t)) - 3, t > 0, t > 0, t >$   
 identifikasi awal data dan sistem system  
 t=0:0:500

Diagram 8.2  
 C + 8  
 R53/PL:1

$y(t) = x_1(t) + x_2(t)$





$y(t) = x(t) * h(t)$   
 $x(t) = 100e^{-t}u(t) - 0.99x(t-2) + e^{-t}u(t)$   
 $x(t) = 100e^{-t}u(t) - 0.99x(t-2) + e^{-t}u(t)$   
 Identify filter form and/or determining system

Dec 20 8.3  
 Ex 8  
 R54/P12

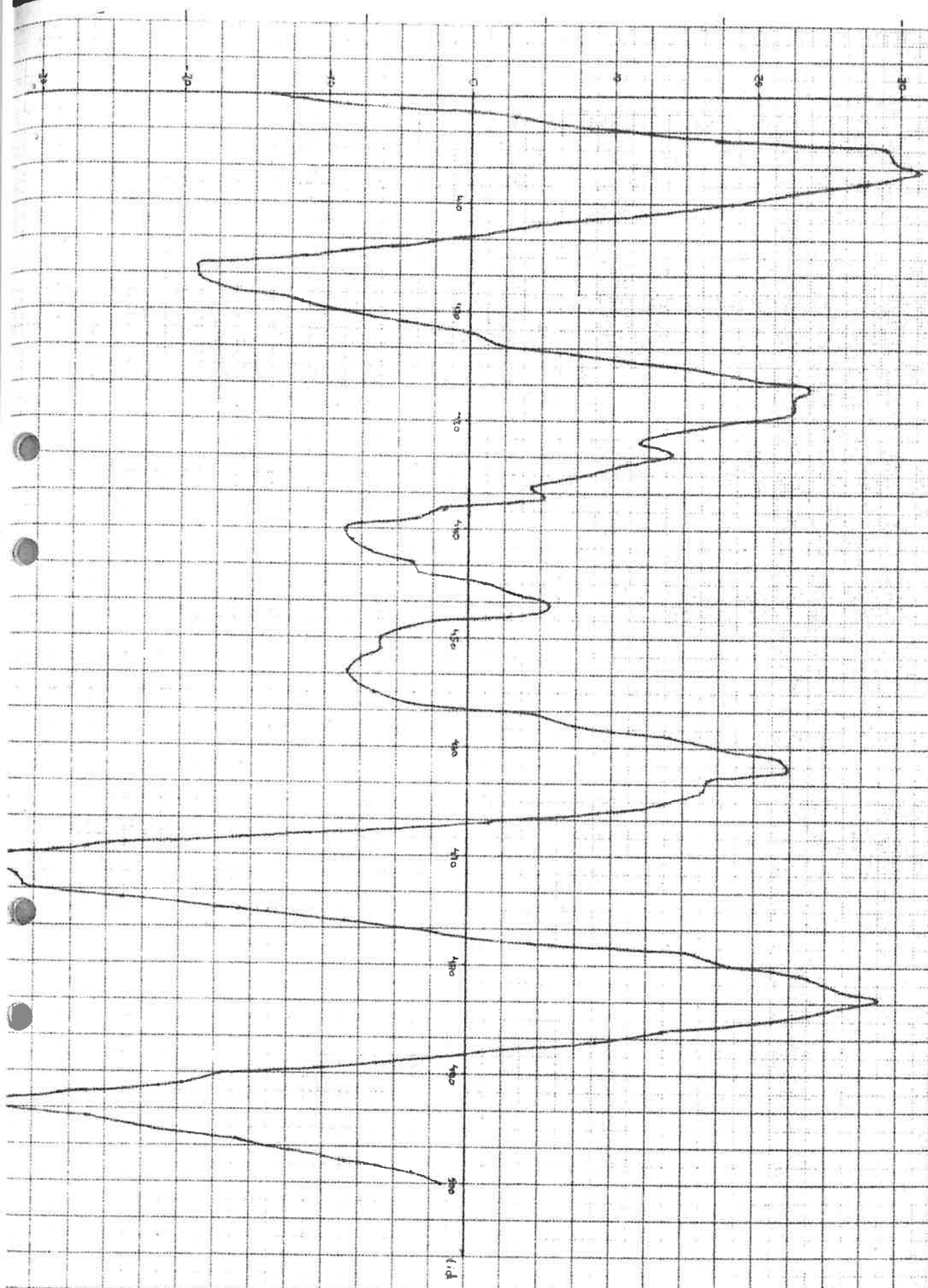
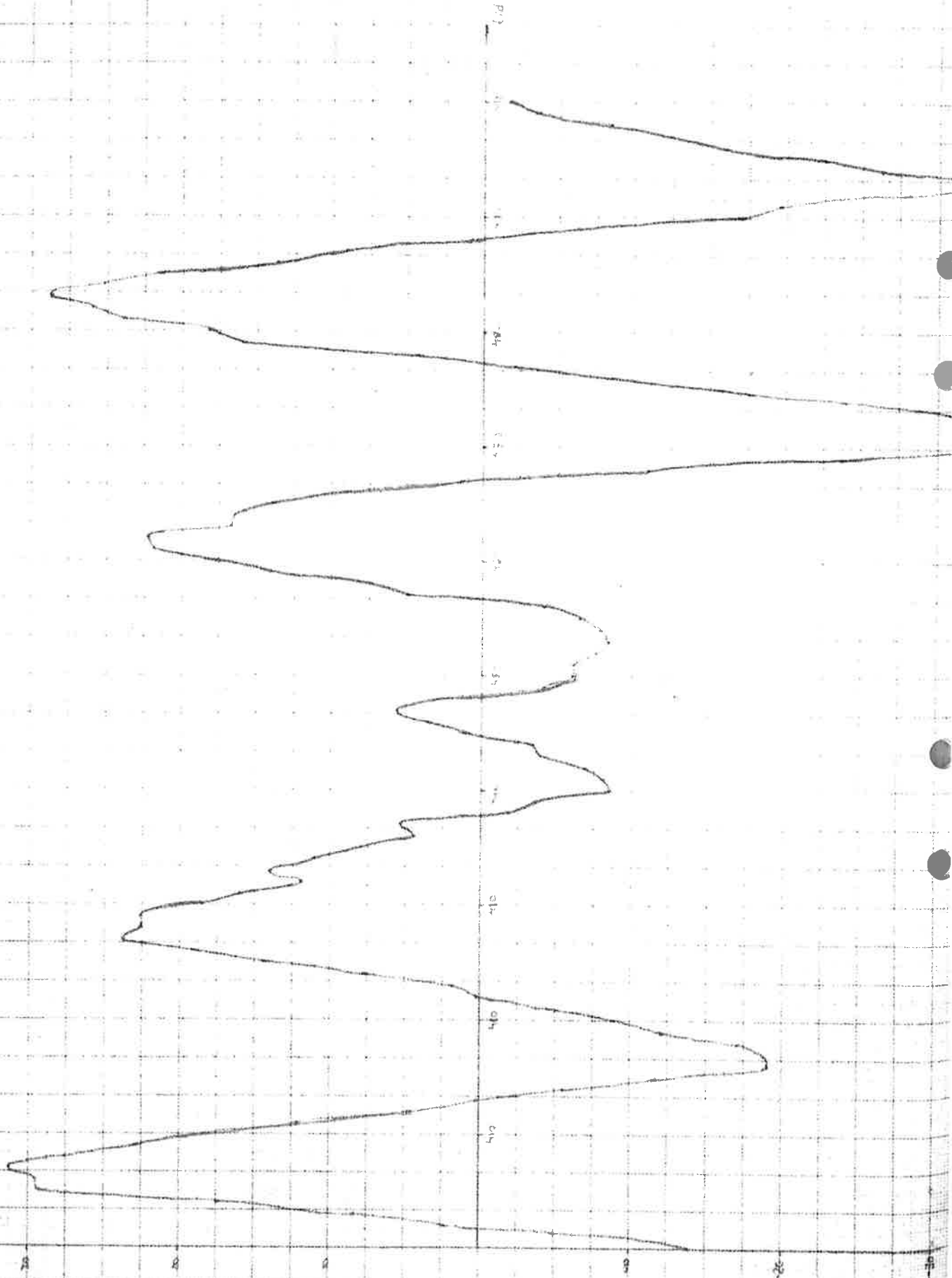


Diagram 8.4

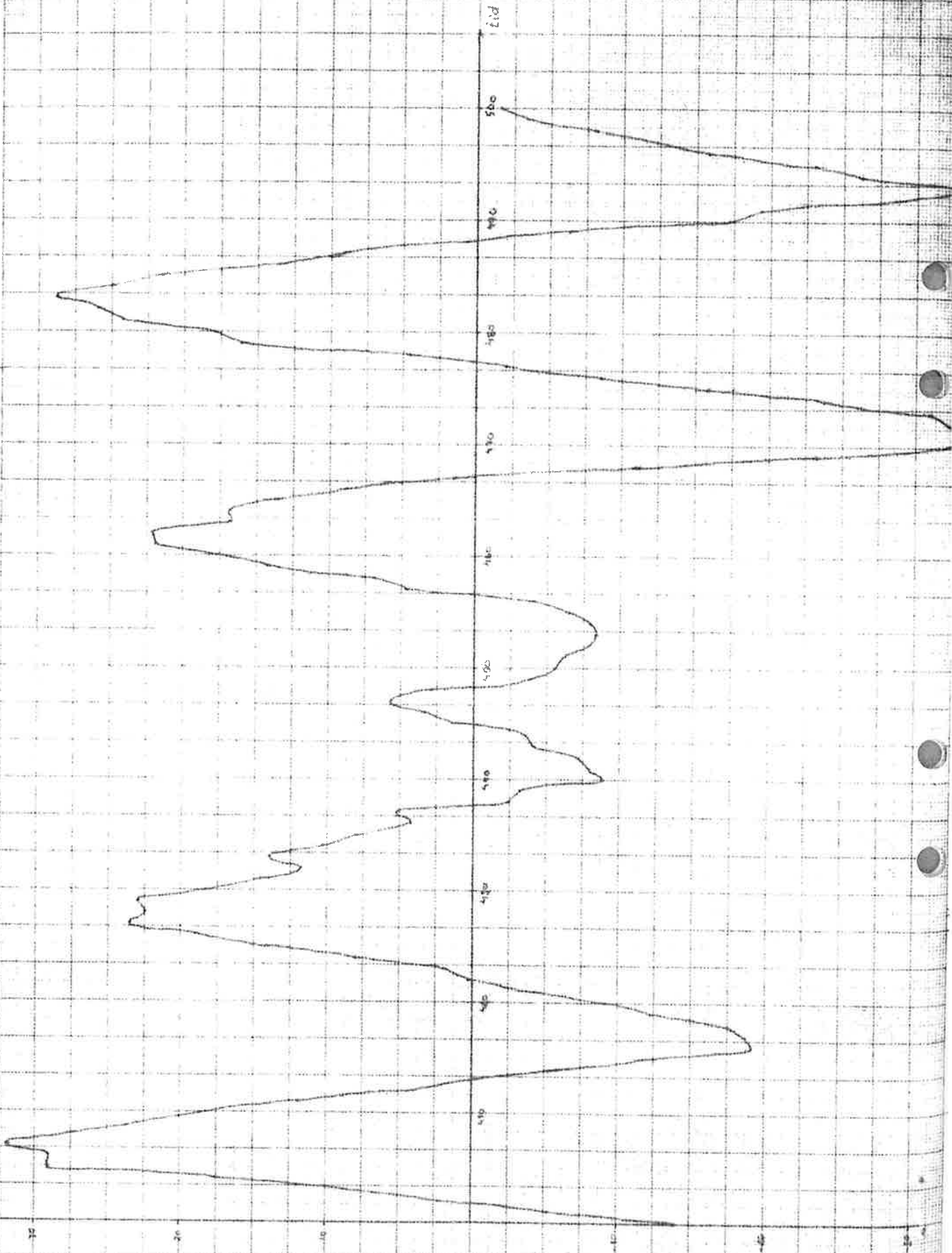
Ex B

236/D1.3

System 8.4  
 $x(t) = 100 \cos(2\pi t) + 0.5 \cos(4\pi t) + 0.1 \cos(6\pi t)$   
 $y(t) = 100 \cos(2\pi t) + 0.5 \cos(4\pi t) + 0.1 \cos(6\pi t)$   
Plot of  $x(t)$  and  $y(t)$  for  $t = 0$  to  $500$



$x_1(t) = x_1(t) + x_2(t)$   
 $x_1(t) = 1.50 \times 10^{-4} - 0.9025 \times 10^{-2} = -0.8875 \times 10^{-2}$   
 $x_2(t) = 1.50 \times 10^{-4} - 0.73 \times 10^{-2} = -0.715 \times 10^{-2}$   
 The circuit is a 2nd order system  
 system  $\tau = 0.005$  s.



DATE: 10/10/50  
PAGE: 3  
NO. 1015

10/10/50  
10/10/50  
10/10/50  
10/10/50

