LISPID - User's Manual

Källstrom, Claes G

1978

Document Version:
Publisher's PDF, also known as Version of record

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

CODEN: LUTFD2/(TFRT-7147)/1-147/(1978)

LISPID - USER'S MANUAL

CLAES G. KÄLLSTRÖM

Department of Automatic Control
Lund Institute of Technology
August 1978

Dokumentutgivare
Lund Institute of Technology

Handläggare Dept of Automatic Control
Claes G. Källström

Dokumentnamn
REPORT

Utgivningsdatum
Aug 1978

Dokumentbeteckning
LUTFD2/(TFRT-7147)/1-147 (1978)

Ärendebeteckning
STU 77-5766

Författare
Claes G. Källström

10T4

Dokumenttitel och undertitel

LISPID - user´s manual

Referat (sammandrag)

LISPID is a flexible computer program for identification of linear multivariable stochastic systems. It is mainly written in FORTRAN and it has been developed on UNIVAC 1108. The prediction error method, the maximum likelihood method and the output error method are included. The mathematical model used in LISPID can be given in continuous or discrete time form and it can be time varying. Different descriptions of process and measurement noise are permitted and the sampling can be uniform or varying.

Referat skrivet av
Author

Förslag till ytterligare nyckelord
system identification, process identification, parameter estimation

Klassifikationssystem och -klass(er)
50T0

Indextermer (ange källa)
52T0

Omfång
147 pages

Övriga bibliografiska uppgifter
56T2

Språk
English

Sekretessuppgifter
60T0

ISSN
60T4

ISBN
60T6

Dokumentet kan erhållas från
Department of Automatic Control
Lund Institute of Technology
P O Box 725, S-220 07 LUND 7, Sweden

Mottagarens uppgifter
62T4

Pris
66T0

LISPID - USER´S MANUAL

Claes G. Källström

TABLE OF CONTENTS

1. INTRODUCTION

The computer program LISPID (LInear System Parameter IDentification) is designed for identification of linear, multivariable stochastic systems. It has been developed on UNIVAC 1108 (EXEC 8). It has also been implemented on IBM 360 and 370, but this manual only describes the UNIVAC version of LISPID.

The program LISPID which mainly is written in FORTRAN consists of 65 subroutines. Including comments the program size is approximately 10 000 statements. The program without any data storage requires a core of 55 k cells on the UNIVAC 1108, if no segmentation is used. Using segmentation and overlays the core required can be reduced to 30 k. Additional memory space is required to store data. LISPID is designed for batch processing, but the EXEC 8 system of UNIVAC 1108 permits in certain cases that the program is executed in demand mode from a terminal.

A general description of LISPID is given in Källström, Essebo and Åström (1976). The program is based on the maximum likelihood method for parameter estimation, since this method has many desirable properties. The one-step prediction errors are minimized in the ML procedure. It is, however, also possible in LISPID to minimize the prediction errors an arbitrary number of steps ahead. It is also possible to assume no process noise in the model, which means that an output error identification method is obtained.

The program LISPID is flexible and allows for a wide variety of model structures. The model can be given in continuous or discrete time form. Different descriptions of process and measurement noise are permitted. The model can also be time varying. The sampling can be uniform or varying and different types of measurements like instantaneous or integrating are permitted. The model may be parameterized in an arbitrary manner.

Calculation of the parameter estimates is basically an optimization problem. Two different algorithms which only use function evaluations are included in LISPID.

The program LISPID can print and plot the input signals, the measurements, the model outputs, the model errors, and the prediction errors. The latter ones have in practice proved to be extremely valuable to check measured data. By analysing the autocorrelation functions of prediction errors and the cross correlation functions between inputs and prediction errors, which are computed and plotted by LISPID, it can be judged if the model obtained is reasonable.

The loss function can be plotted close to the point proposed as optimal, when the estimated parameters are changed one at a time. A possibility of judging if a local optimum point actually has been reached is provided by these plots. They can also be used for an efficient scaling of the unknown parameters to facilitate the optimization. Estimated standard deviations of the parameters obtained can also be calculated.

The program LISPID has been applied to data from many different processes. A review is given in Källström, Essebo and Åström (1976). Examples of different applications are drum boiler dynamics, nuclear reactor dynamics, dynamics of a power generator, pharmacokinetics, plasma kinetics of insulin, thyronine metabolism, and macroeconomic dynamics. The program has been extensively used to determine the steering dynamics of ships (Åström, Källström, Norrbin and Byström, 1975; Åström and Källström, 1976; Byström and Källström, 1978).

2. MODEL STRUCTURES

The program LISPID admits many different model structures. The basic model is a process governed by the stochastic differential equation (see Åström, 1970):

$$dx = Ax\,dt + Bu\,dt + dw \tag{2.1}$$

where x is an n_x - dimensional state vector and u is an n_u - dimensional control vector. The process noise $\{w(t),\ t_o \leq t \leq \infty\}$ is an n_x - dimensional stochastic process with uncorrelated increments which have zero mean values and incremental covariance R_1 dt. The initial state $x(t_o)$ of (2.1) is assumed to have mean value m and covariance R_o. It is also assumed that the initial state is uncorrelated with the process noise $\{w(t)\}$.

The program LISPID admits two different models for the measuring process. It is assumed that the measurements are obtained at discrete times $t_o, t_1, \ldots, t_{N-1}$.

An integrating measurement device can be characterized by

$$\begin{cases} dz = Cx\,dt + Du\,dt + de \\ y(t_k) = \int_{t_k}^{t_{k+1}} dz \qquad k = 0,1,\ldots, N-1 \end{cases} \tag{2.2}$$

where y is an n_y - dimensional vector of measurements and $\{e(t),\ t_o \leq t \leq \infty\}$ are measurement errors which are assumed to be a stochastic process with uncorrelated increments having zero mean values and incremental covariance R_2 dt. The incremental cross covariance between w and e is denoted R_{12} dt. A discussion of the integrating measurement procedure is given in Åström (1970).

Instantaneous measurements characterized by

$$y(t_k) = \tilde{C}\,x(t_k) + \tilde{D}u(t_k) + \tilde{e}(t_k) \quad k = 0,1,\ldots, N-1 \tag{2.3}$$

are also admitted. The measurement errors $\{\tilde{e}(t_k)\}$ are assumed to be a stochastic process of second order with zero mean and covariance $\tilde{R}_2$. It is furthermore assumed that the measurement errors are independent of w and of the initial state.

Sampling of the model (2.1), (2,2) or (2.1), (2.3) is performed in LISPID. The following model is obtained in both cases if it is assumed that the input signals are constant during the sampling interval (see Åström, 1970):

$$\begin{cases} x(t_{k+1}) = \tilde{A}\, x(t_k) + \tilde{B}u(t_k) + \tilde{w}(t_k) \\ y(t_k) = \tilde{C}\, x(t_k) + \tilde{D}u(t_k) + \tilde{e}(t_k) \\ \qquad k = 0,1,\ldots, N-1 \end{cases} \tag{2.4}$$

The process noise $\{\tilde{w}(t_k)\}$ is a second order stochastic process with zero mean and covariance $\tilde{R}_1$. The cross covariance between $\tilde{w}$ and $\tilde{e}$ is denoted $\tilde{R}_{12}$. The initial state is uncorrelated with $\tilde{w}$. The sampling is performed using algorithms described in Mårtensson (1969) and Källström (1973). It is also possible to postulate the discrete time model (2.4) directly.

The innovations representation

$$\begin{cases} \hat{x}(t_{k+1}) = \tilde{A}\, \hat{x}(t_k) + \tilde{B}u(t_k) + K\, \varepsilon(t_k) \\ y(t_k) = \tilde{C}\, \hat{x}(t_k) + \tilde{D}u(t_k) + \varepsilon(t_k) \\ \qquad k = 0,1,\ldots, N-1 \end{cases} \tag{2.5}$$

is computed from (2.4). See Åström (1970). The model (2.5) is always used for the actual iterations performed in LISPID. Notice that it is possible to postulate the innovations representation (2.5) directly.

The following model structures are permitted in LISPID:

1: (2.1),(2.2) - $A,B,C,D,R_1,R_{12},R_2,R_o,m$

2: (2.1),(2.3) - $A,B,\tilde{C},\tilde{D},R_1,\tilde{R}_2,R_o,m$

3: (2.4) - $\tilde{A},\tilde{B},\tilde{C},\tilde{D},\tilde{R}_1,\tilde{R}_{12},\tilde{R}_2,R_o,m$ (2.6)

4: (2.1),(2.2),(2.5) - A,B,C,D,K,m

5: (2.1),(2.3),(2.5) - $A,B,\tilde{C},\tilde{D},K,m$

6: (2.5) - $\tilde{A},\tilde{B},\tilde{C},\tilde{D},K,m$

In each case parameters to be estimated can enter the indicated matrices and vectors in an arbitrary manner. The different models may also be time varying in LISPID.

It is possible to assume no process in the different model structures. The identification procedure is then equivalent to an output error method. In this case the covariance matrices R_1, R_{12}, R_2, $\tilde{R}_1$, $\tilde{R}_{12}$, $\tilde{R}_2$, R_o and the filter gain K are not used, i.e. they are excluded from (2.6).

3. CRITERIA

The program LISPID admits optimization of different criteria. The most general loss functions are

$$V_1(p_1,p_2) = \frac{1}{N-p_2} \det \left[\frac{1}{p_2-p_1+1} \sum_{k=0}^{N-p_2-1} \sum_{p=p_1}^{p_2} \varepsilon_p(t_{k+p})\varepsilon_p^T(t_{k+p}) \right] \qquad (3.1)$$

$$V_2(p_1,p_2) = \frac{1}{2(N-p_2)(p_2-p_1+1)} \sum_{k=0}^{N-p_2-1} \sum_{p=p_1}^{p_2} \left[\log \det R_p(t_{k+p}) + \right.$$

$$\left. + \varepsilon_p^T(t_{k+p}) R_p^{-1}(t_{k+p}) \varepsilon_p(t_{k+p}) \right] + \frac{1}{2} n_y \log 2\pi \qquad (3.2)$$

where an average loss over an interval (p_1, p_2) of prediction errors is minimized. A necessary condition is $p_2 \geq p_1 \geq 1$, but small modifications of (3.1) and (3.2) are minimized in the ML procedure ($p_1 = p_2 = 1$). See below. The p-step prediction error ε_p and the corresponding covariance matrix R_p are determined recursively through the Kalman-Bucy filtering theory. A detailed description of the recursions is given in Källström, Essebo and Åström (1976). The loss function V_1 can be regarded as a simplified version of V_2. It is possible to use V_1 instead of V_2 if the system and covariance matrices are time-invariant and if the sampling interval is constant (Åström and Eykhoff, 1971). The loss function selected in LISPID is determined from the given model structure. Notice, however, that V_1 always is calculated, even if V_2 is minimized.

The following modified versions of (3.1) and (3.2) are minimized in the ML procedure ($p_1 = p_2 = 1$):

$$V_1 = \frac{1}{N} \det \left[\sum_{k=0}^{N-1} \varepsilon(t_k)\ \varepsilon^T(t_k) \right] \tag{3.3}$$

$$V_2 = \frac{1}{2N} \sum_{k=0}^{N-1} \left[\log \det R(t_k) + \varepsilon^T(t_k)\ R^{-1}(t_k)\varepsilon(t_k) \right] + \frac{1}{2} n_y \log 2\pi \tag{3.4}$$

where ε is the one-step prediction error and R the corresponding covariance matrix. Notice that (3.3) and (3.4) differ from (3.1) and (3.2) only in the way ε and R at the initial time t_o are treated. The loss function (3.4) is related to the likelihood function L through:

$$V_2 = - \frac{1}{N} \log L \tag{3.5}$$

The output error method can be regarded as a special case of the ML method, where it is assumed that no process noise is present. The loss function (3.3) or (3.4) is thus minimized in the output error procedure. Due to its flexible structure it is also easy to incorporate an arbitrary user-defined loss function into LISPID.

4. INPUT PARAMETERS

The computations in LISPID are controlled by a number of parameters. The parameter values are given by the user and they are read from cards or card images using the subroutine IOLISP. A table of the input parameters is shown in Appendix A. The different parameters are explained by:

NPAR - Total number of parameters of the model, i.e. dimension of vector PAR ($2 \leq$ NPAR ≤ 40).

NTH - Number of parameters that are estimated, i.e. dimension of vectors ITH and SCAL (2 ≤ NTH ≤ NPAR).

PAR - Vector of dimension NPAR containing values of the model parameters. A value is considered as an initial guess, if the corresponding parameter is going to be estimated in LISPID, otherwise the value is assumed to be fixed.

ITH - Vector of dimension NTH containing pointers to the parameters of PAR that are going to be estimated. For example, if PAR (4) is the first parameter of PAR that is going to be estimated, then ITH (1) = 4. (1 ≤ ITH (I) ≤ NPAR, I = 1,2,..., NTH).

SCAL - Vector of dimension NTH containing scale factors for the parameters of PAR that are going to be estimated. The values PAR (ITH (I)) * SCAL (I), I = 1,2,..., NTH, are actually used in LISPID. The minimization algorithm NUFLET is very sensitive to the scaling, while the algorithm POWBRE is less sensitive. Initially the scale factors should be determined in such a way that the magnitude of the scaled variables is approximately 0.1. For example, if ITH (1) = 4 and the value of PAR (4) is known to be approximately 0.01, then SCAL (1) = 10 is appropriate. (SCAL (I) ≠ 0, I = 1,2,..., NTH).

IMIN - IMIN = 1: The minimization algorithm NUFLET is used.
IMIN = 2: The minimization algorithm POWBRE is used.
NUFLET is faster than POWBRE, but the minimization is often terminated too soon when NUFLET is used. It is suitable to start with NUFLET and continue with POWBRE as soon as appropriate scale factors have been determined. However, if NTH is less than approximately 5, then NUFLET can be used all the time.

LOOP - LOOP = -1: Print data, i.e. input signals and measurements, on line printer and/or plot data. Notice that it is necessary that NPRI (3) = 1 if the data are going to be printed and NPLOT = 1 or -1 if the data are going to be plotted.

LOOP = 0: The model given is simulated to the input signals using the parameter values of PAR, i.e. no parameter estimation is performed.

LOOP = 1: Parameter estimation is performed.

NPRI (1) - NPRI (1) = 1: System and covariance matrices are printed using the initial parameter values.

NPRI (1) = 0: No printing of the initial system.

NPRI (2) - NPRI (2) = 1: System and covariance matrices are printed using the final parameter values.

NPRI (2) = 0: No printing of the final system.

NPRI (3) - NPRI (3) = 1: Input signals, measurements, model outputs, model errors, and prediction errors are printed.

NPRI (3) = 0: No printing of data.

NPLOT - NPLOT = 1: Input signals, measurements, model outputs, model errors, and prediction errors are plotted. The model outputs are plotted as dashed lines and the measurements as continuous lines in the same diagrams. The inputs are shown in histogram plots.

NPLOT = -1: Input signals, measurements, model outputs, model errors, and prediction errors are plotted. The model outputs are plotted as continuous lines and the measurements as dots in the same diagrams. The inputs are shown in histogram plots.

NPLOT = 0: No plotting of data.

NP - Number of sampling events ($2 \leq NP \leq 2000$).

IT - IT = 1: The sampling times are given by the user in the vector TIM.

IT = 0: No sampling times are given in the vector TIM. It is suitable to use IT = 0 when the sampling interval is constant.

ISYS - ISYS > 0 means a continuous time model.

ISYS < 0 means a discrete time model.

ISYS = 1 or -1: No process noise is assumed, i.e. the identification procedure is equivalent to an output error method.

ISYS = 2 or -2: Prediction error method based on the innovations model (cf. (2.5)).

ISYS = 3 or -3: Prediction error method based on the general model (2.1) or (2.4).

MEAS - MEAS = 0: Instantaneous measurements (cf. (2.3)).

MEAS = 1: Integrating measurements (cf. (2.2)).

ISAMP - ISAMP = 1: Constant sampling interval.

ISAMP = 2: Constant or almost constant sampling interval, but some measurements are missing. It is necessary that IT = 1.

ISAMP = 3: Varying sampling interval. It is necessary that IT = 1 if ISYS > 0.

TSAMP - Sampling interval. If the sampling interval is calculated from the vector TIM or if the sampling interval is not required for the computations, then TSAMP = -1 must be used.

NPRED1 - Parameter p_1 of (3.1) and (3.2).

NPRED2 - Parameter p_2 of (3.1) and (3.2). When the maximum likelihood or output error method is applied, then NPRED1 = NPRED2 = 1 must be used. A necessary condition is 1 $\leq$ NPRED1 $\leq$ NPRED2.

NX - Number of states, i.e. n_x in Section 2 (1 $\leq$ NX $\leq$ 20).

NU - Number of inputs, i.e. n_u in Section 2. If a model without input signals is applied then NU = 0 must be used. (0 $\leq$ NU $\leq$ 20).

NY - Number of measurement signals, i.e. n_y in Section 2 (1 $\leq$ NY $\leq$ 20).

NOMAT - Vector of dimension 6 describing the system and covariance matrices that are going to be used (cf. Section 2). If a matrix is zero then the corresponding value of NOMAT ( ) should be zero, otherwise NOMAT ( ) = 1. NOMAT (1) = 1 if B or $\tilde{B}$ nonzero.

NOMAT (2) = 1 if C or $\tilde{C}$ nonzero.
NOMAT (3) = 1 if D or $\tilde{D}$ nonzero.
NOMAT (4) = 1 if R_1 or $\tilde{R}_1$ nonzero.
NOMAT (5) = 1 if R_{12} or $\tilde{R}_{12}$ nonzero.
NOMAT (6) = 1 if R_2 or $\tilde{R}_2$ nonzero.

IACC - IACC = 1: Standard deviations of parameter estimates, gradient vector, second derivative matrix, inverse of second derivative matrix, eigenvalues and eigenvectors of second derivative matrix are calculated and printed. The point proposed as optimal is used as nominal point and the step lengths h and -h, where h = 10^{-3}, are used for the calculations.

IACC = 2: As IACC = 1 but the step lengths h, 2h, -h,-2h, where h = 10^{-3}, are used. IACC = 2 requires more computations than IACC = 1, but a more accurate result is obtained.

IACC = 0: The standard deviations and the related quantities are not calculated and not printed.

HH - This parameter is not used, which means that the value zero can, for example, be assigned to HH.

NPLOTC - NPLOTC = 1: Test quantities for the prediction errors are calculated and printed.

NPLOTC = 2: As NPLOTC = 1, but autocorrelation functions of prediction errors and cross correlation functions between inputs and prediction errors are also calculated and printed.

NPLOTC = 3: As NPLOTC = 1, but the correlation functions are also calculated and plotted.

NPLOTC = 4: As NPLOTC = 2, but the correlation functions are also plotted.

NPLOTC = 0: The test quantities and the correlation functions are not calculated, printed or plotted.

NOL - Maximum number of time lags used in the calculations of the correlation functions, when NPLOTC = 2, 3 or 4. The time lags 0, 1, 2,..., NOL are used for the autocorrelation functions and -NOL,..., -1, 0, 1,..., NOL are used for the cross correlation functions. ($0 \le$ NOL ≤ 50).

TS - Sampling interval used only if subroutine SGAIN is called from the model subroutine. Usually TS should be equal to TSAMP.

EPSI - Test quantity used only if subroutine SGAIN is called from the model subroutine. The parameter EPSI determines when the iterations of the Riccati equation are terminated. A suitable value is EPSI = 0.00001.

NKAL - Maximum number of iterations of the Riccati equation used only if subroutine SGAIN is called from the model subroutine. A suitable value is NKAL = 500. (NKAL $\ge$ 2).

ICHK - ICHK = 1: The loss function is plotted on line printer close to the point proposed as optimal, when the estimated parameters are changed one at a time.
ICHK = 0: The loss function is not plotted on line printer.

ICR - ICR = 1: More input parameters are read (see subroutines IOLISP and LISPID).
ICR = 0: No more input parameters are read. Input parameters which are not given by the user are determined in subroutine LISDAT.

Six different model structures were given in (2.6). The corresponding input parameter values are:

1:	ISYS = 3	MEAS = 1	
2:	ISYS = 3	MEAS = 0	
3:	ISYS = -3	MEAS = 0	(4.1)
4:	ISYS = 2	MEAS = 1	
5:	ISYS = 2	MEAS = 0	
6:	ISYS = -2	MEAS = 0	

When the output error method is applied the following 3 model structures are permitted:

7: (2.1), (2.2)	- A, B, C, D, m	ISYS = 1	MEAS = 1	
8: (2.1), (2.3)	- A, B, $\tilde{C}$, $\tilde{D}$, m	ISYS = 1	MEAS = 0	(4.2)
9: (2.4) or (2.5)	- $\tilde{A}$, $\tilde{B}$, $\tilde{C}$, $\tilde{D}$, m	ISYS = -1	MEAS = 0	

Notice that the vector NOMAT must be assigned appropriate values in each case.

Some suitable input parameters for 3 different cases are given below. The first case is printing and plotting of data, i.e. no identification or simulation is performed:

LOOP = -1
NPRI (3) = 1
NPLOT = 1 or -1

The second case is parameter estimation. To avoid unnecessary computations, printings and plottings during the identification mode it is suggested to use the following parameter values:

LOOP = 1
NPRI (1) = 1
NPRI (2) = 1
NPRI (3) = 0
NPLOT = 0
IACC = 0
NPLOTC = 0
ICHK = 1

The third case is simulation of a model that possibly has been obtained in the identification mode. By using the following parameter values as much information as possible is printed and plotted:

LOOP = 0
NPRI (1) = 1
NPRI (2) = 1
NPRI (3) = 1
NPLOT = 1 or -1
IACC = 2
NPLOTC = 4
ICHK = 1

Four complete test examples are discussed in Section 6. The input parameters for each example are given in Appendix F.

5. USER - DESIGNED PROGRAMS

It is necessary to incorporate a main program MAIN and a model subroutine SYST before LISPID can be used. MAIN and SYST should be written by the user in FORTRAN. It is, of course, possible to use other names than MAIN and SYST.

The main program MAIN should call subroutines IOLISP, DATEXP and GRDCHK. It is necessary also to call the administration subroutine LISPID from MAIN. The data required for the identification must be read and organized in a certain way. The organization of data is described in detail in the program head of subroutine LISPID. It is suitable to call a user-designed subroutine RDATA from MAIN to read the data.

The mathematical model used for the identification must be described in the model subroutine SYST. The argument list of SYST is predetermined and it is given in the program head of LISPID. The subroutine SINT should be called from SYST to evaluate the loss function. It is also suitable to call SGAIN and OVFLOW from SYST.

A schematic diagram of the program package LISPID is given in Fig. 5.1. Many auxiliary subroutines not shown in Fig. 5.1 are also included. It is, of course, also possible to call auxiliary subroutines from the user-designed programs MAIN, RDATA and SYST. Compiled elements of the subroutines required can be found on a special LISPID file. The file is normally stored on magnetic tape. The element table of the LISPID file is given in Appendix B where it is also described how the LISPID file can be transfered from magnetic tape to disc . The program heads of the most important subroutines are listed in Appendix C.

A standard main program MAIN and subroutine RDATA have been written to facilitate the use of LISPID. It is then assumed that the data are read from cards or card images in the following way:

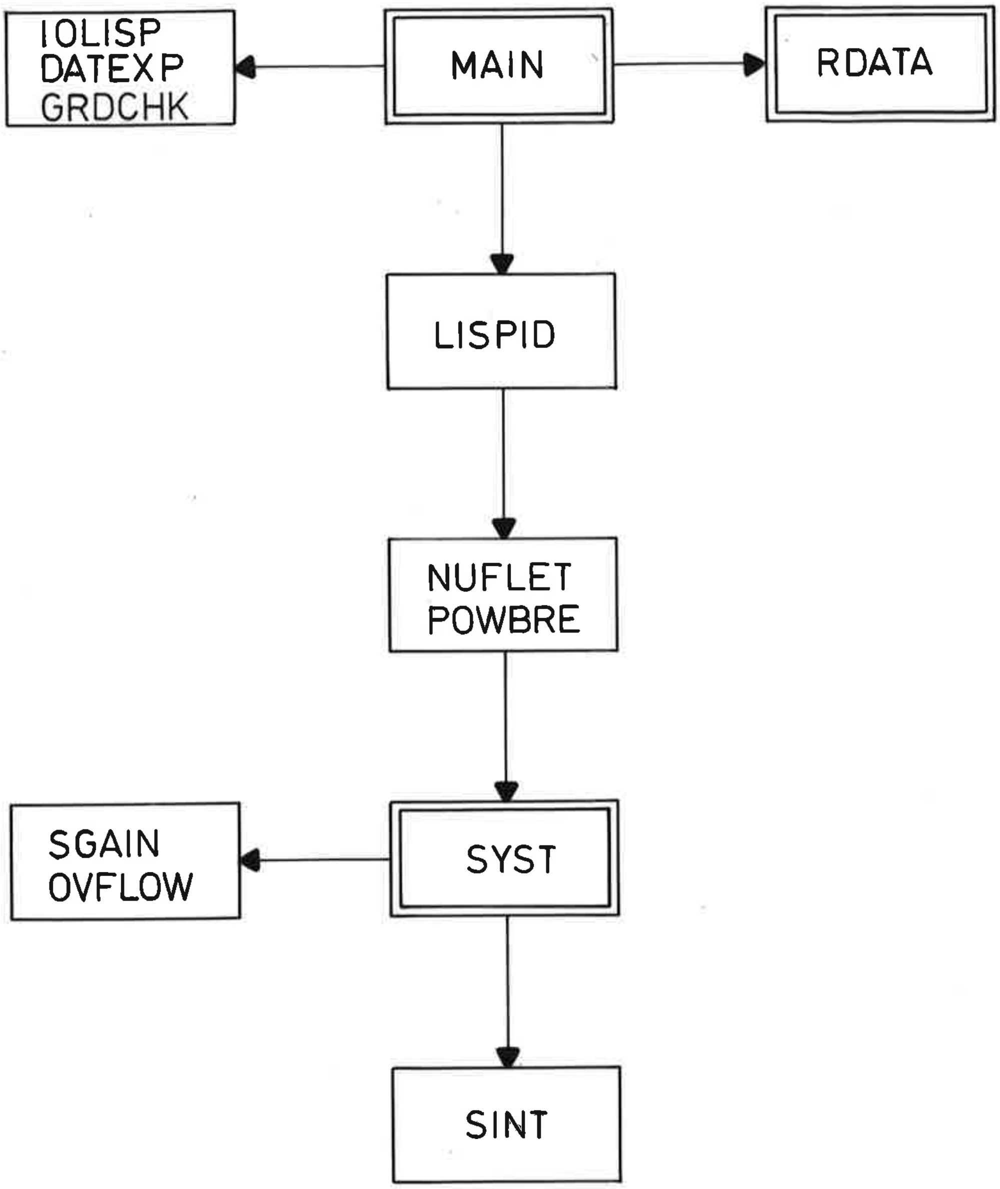


Fig. 5.1 - Schematic diagram of the program package LISPID. MAIN, RDATA and SYST are user-designed programs.

Card 1: $[t(1)]\ u_1(1)\ldots\ u_{NU}(1)\ y_1(1)\ldots\ y_{NY}(1)$

Card 2: $[t(2)]\ u_1(2)\ldots\ u_{NU}(2)\ y_1(2)\ldots\ y_{NY}(2)$

. (5.1)

.

.

Card NP: $[t(NP)]\ u_1(NP)\ldots\ u_{NU}(NP)\ y_1(NP)\ldots y_{NY}(NP)$

where t(1) is the first sampling time, $u_1(1)$ is the value of the first input signal at time t(1) and y_1 is the value of the first measurement signal at time t(1). The number of input signals are NU and the number of measurement signals are NY. Notice that it is possible to use NU=0. The sampling times t(1), t(2),..., t(NP) must be omitted when IT=0. The number of sampling events NP and the parameters NU, NY and IT used in RDATA are taken from the input parameters (see Section 4 and Appendix A). The data (5.1) are read in free format. Examples of permitted real numbers are given in Appendix A. The numbers must be separated by at least one blank. If the data for each sampling time need more space than one card, then it is possible to use two or more cards for each sampling event. It is also possible to punch data of two or more sampling events on the same card. The FORTRAN programs MAIN and RDATA and the corresponding compiled elements are stored on the LISPID file. Listings are given in Appendix D. Notice that it is necessary that the name of the model subroutine is SYST if MAIN is used without changes. If the data are organized in a way different from the description above, then the user must change subroutine RDATA.

A frame-work of a model subroutine SYST is stored on the LISPID file. A listing is given in Appendix D. It is necessary to incorporate a mathematical model defined by the user before SYST can be compiled and linked together with the program package LISPID.

Suitable MAP commands for generation of an absolute element which can be executed are stored on the LISPID file. The name of the element containing the MAP commands is LISMAP. The element is listed in Appendix E.

Examples of user-designed programs are given in Appendix F. The test examples are discussed in Section 6. The generation and execution of a complete test example is shown in Appendix G.

6. TEST EXAMPLES

Four complete test examples are discribed in this section. They are all applications of system identification to determination of ship steering dynamics (Åström, Källström, Norrbin and Byström, 1975; Åström and Källström, 1976; Byström and Källström, 1978).

Example 1

The following simple transfer function describes the rudder input - heading output relation for a ship:

$$G(s) = \frac{K}{s(1+sT)} \tag{6.1}$$

A state space representation of (6.1) is easily obtained:

$$\begin{bmatrix} \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} -\frac{1}{T} & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} r \\ \psi \end{bmatrix} + \begin{bmatrix} K/T \\ 0 \end{bmatrix} \delta \tag{6.2}$$

where δ is the rudder angle, r is the yaw rate and ψ is the heading angle.

It is now possible to determine K and T from a record of rudder and heading angles using LISPID. The model structure 2 of (2.6) is assumed, i.e. the general model (2.1) combined with instantaneous measurements (2.3). The following assignments are performed in the model subroutine SYST (see Appendix F):

$$A = \begin{bmatrix} -1/p_2 & 0 \\ 1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} p_1/p_2 \\ 0 \end{bmatrix}$$

$$\tilde{C} = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

$$R_1 = \begin{bmatrix} |p_3| & 0 \\ 0 & 0 \end{bmatrix}$$

$$\tilde{R}_2 = 0.01 \qquad (6.3)$$

$$R_o = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$m = \begin{bmatrix} 0 \\ 147.15 \end{bmatrix}$$

Notice that the corresponding variables used in SYST are A, B, C, R1, R2, P0 and X0. It is concluded that p_1 = K and p_2 = T.

By assuming that the data are recorded with a constant sampling interval it is possible to simplify the calculations in LISPID, since the mathematical model used is time-invariant. The simplification is obtained by calculating the filter gain K of (2.5) using subroutine SGAIN. The model structure 5 of (2.6) is thus actually used in LISPID, i.e. ISYS = 2 is required if prediction error identification is to be performed.

The standard programs MAIN and RDATA (see Appendix D) are used for this test example. Suitable input parameters for maximum likelihood identification are shown in Appendix F, where test data also are given (DAT1). Prediction error identification is easily obtained by changing the input parameters NPRED1 and NPRED2. Output error identification is obtained by changing the input parameters

NTH = 2
ISYS = 1
NOMAT (4) = 0
NOMAT (6) = 0

and by deleting ITH(3) and SCAL(3). The MAP commands used for this example are listed in Appendix F (LMAP1).

The generation and execution of test example 1 is described in detail in Appendix G. The prints and plots obtained when the input parameters given in Appendix F are used can be found in Appendix H. A discussion of the prints and plots are given in Section 7.

Example 2

An improved model of ship steering dynamics is given by the state space representation

$$\begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{V}{L} a_{11}' & V a_{12}' & 0 \\ \frac{V}{L^2} a_{21}' & \frac{V}{L} a_{22}' & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} v \\ r \\ \psi \end{bmatrix} + \begin{bmatrix} \alpha_1 \frac{V^2}{L} b_{11}' \\ \alpha_1 \frac{V^2}{L^2} b_{21}' \\ 0 \end{bmatrix} \delta(t-T_D) + \begin{bmatrix} f_1 \\ f_2 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} v_m(t_k) \\ \psi_m(t_k) \end{bmatrix} = \begin{bmatrix} \alpha_2 & L_1 \cdot \alpha_2 & 0 \\ 0 & 0 & 1/\alpha_1 \end{bmatrix} \begin{bmatrix} v(t_k) \\ r(t_k) \\ \psi(t_k) \end{bmatrix} + \begin{bmatrix} f_3 \\ 0 \end{bmatrix} \quad k=0,1,\ldots,N-1 \tag{6.4}$$

where the transversal velocity v at the origin also is introduced as a state variable. The parameters a_{11}', a_{12}', a_{21}', a_{22}', b_{11}' and b_{21}' are normalized using the ship speed V and the ship length L. It is assumed that V is constant. Instantaneous measurements of the transversal velocity v_m at a distance L_1 forward the origin and of the heading angle ψ_m are recorded. α_1 and α_2 are conversion factors from degrees to radians and from m/s to knots ($\alpha_1 = 0.01745$, $\alpha_2 = 1.944$).

The biases f_1, f_2 and f_3 can be estimated by introducing an artificial unit step input signal. The time delay T_D describes approximately the time constant of the steering engine. It is possible to incorporate the model (6.4) with a time delay in LISPID by introducing an extra input signal containing the rudder angles slided one step forwards in time. However, it is then necessary to sample the model (6.4) by calling COSDY and, if process noise is present, also TRANS from the model subroutine. If it is assumed that the sampling interval is constant (ISAMP = 1) or constant but some measurements are missing (ISAMP = 2), then the filter gain K of (2.5) can be computed using subroutine SGAIN, since the model (6.4) is time-invariant. Cf. test example 1. In this case the model structure 6 of (2.6) is actually used in LISPID, i.e. ISYS = -2 is required if prediction error identification is to be performed.

The following assignments are performed in the model subroutine SYST2 (see Appendix F):

$$A = \begin{bmatrix} \frac{p_1}{p_2} p_3 & p_1 p_4 & 0 \\ \frac{p_1}{p_2^2} p_5 & \frac{p_1}{p_2} p_6 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} \alpha_1 \frac{p_1^2}{p_2} p_7 & p_9 & 0 \\ \alpha_1 \frac{p_1^2}{p_2^2} p_8 & p_{10} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{C} = \begin{bmatrix} \alpha_2 & p_{12} \alpha_2 & 0 \\ 0 & 0 & 1/\alpha_1 \end{bmatrix} \qquad (6.5)$$

$$\tilde{D} = \begin{bmatrix} 0 & p_{11} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$R_1 = \begin{bmatrix} |p_{13}| & \sqrt{|p_{13}||p_{14}|} \sin p_{15} & 0 \\ \sqrt{|p_{13}||p_{14}|} \sin p_{15} & |p_{14}| & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{R}_2 = \begin{bmatrix} |p_{16}| & 0 \\ 0 & |p_{17}| \end{bmatrix}$$

$$R_o = \begin{bmatrix} |p_{18}| & 0 & 0 \\ 0 & |p_{19}| & 0 \\ 0 & 0 & |p_{20}| \end{bmatrix}$$

$$m = \begin{bmatrix} p_{21} \\ p_{22} \\ \alpha_1 p_{23} \end{bmatrix}$$

$$T_D = T_s \; |\sin p_{24}|$$

The parameters $p_3 = a_{11}'$, $p_4 = a_{12}'$, $p_5 = a_{21}'$, $p_6 = a_{22}'$, $p_9 = f_1$, $p_{10} = f_2$, $p_{11} = f_3$, p_{13}, p_{14} and p_{24} are estimated using the prediction error method (NPRED1 = NPRED2 = 5), while the other parameters are given fixed values dependent on the ship and the experimental conditions. The special form of the entries R_1 (1,2) and R_1(2,1) guarantees that the matrix R_1 is nonnegative definite. The special expression for T_D guarantees that $0 \leq T_D \leq T_s$, where T_s is the sampling interval.

A new main program MAIN2 and a new subroutine RDATA2 have been written, since it is necessary to generate extra input signals (see Appendix F). Notice that the extra inputs are excluded in the computations of the correlation functions by assigning INU(2) = INU(3) = 0 in MAIN2. It is convenient to write the parameters on a disc file, when many parameters are estimated. This is performed by calling PARFIL from MAIN2. Notice that it is necessary to assign a disc file "106", before PARFIL is called.

Suitable input parameters for prediction error identification are shown in Appendix F, where test data also are given (DAT2). Notice that ISAMP=2 is used. Output error identification is obtained by changing the input parameters

NTH = 8

ISYS = -1

NOMAT (4) = 0

NOMAT (6) = 0

and by deleting ITH(8) - ITH(9) and SCAL(8) - SCAL(9). The MAP commands used for example 2 are listed in Appendix F (LMAP2).

Example 3

This example is a small modification of example 2. The time delay T_D is assumed to be zero and the measurements are now recorded with a varying sampling interval (ISAMP = 3). Only one extra input signal is required in this case to estimate the biases f_1, f_2 and f_3 (cf.(6.4)). This means that the third columns of the matrices B and $\tilde{D}$ of (6.5) are omitted as well as the expression for T_D.

It is not possible to compute a stationary filter gain K using SGAIN since the sampling interval is varying. The model structure 2 of (2.6) is thus used, i.e. ISYS = 3. In this case the parameters of the R_o-matrix are also estimated. The parameters p_3, p_4, p_5, p_6, p_9, p_{10}, p_{11}, p_{13}, p_{14}, p_{18}, p_{19} and p_{20} of (6.5) are thus estimated using the prediction error method (NPRED1 = 3, NPRED2 = 5).

The programs MAIN3, RDATA3, SYST3, the MAP commands LMAP3 and the test data DAT3 are listed in Appendix F. LOOP is equal to zero in the test data, which means that simulations only are performed.

Example 4

A small modification of example 3 is considered. The ship speed V (cf. (6.4)) is varying in this example. The same parameters as in example 3 are estimated, but it is necessary to use NPRED1 = NPRED2 = 1 since the model is time varying. The measurements of the speed V are read using subroutine RDATA4 and they are stored in a vector VSP of the COMMON BLOCK /SPEED/. The speed measurements are printed in the main program MAIN4. The model subroutine SYST4 shows how to introduce the speed dependence. The programs are listed in Appendix F together with the testdata DAT4 and the map commands LMAP4.

7. PRINTS AND PLOTS

The output generated from test example 1 of Section 6 is given in Appendix H. Most parts are self-explained, but some points are explained in some more detail in this section.

The vector TH contains the scaled parameters of vector PAR that are going to be estimated. The names of the sampled system and covariance matrices are obtained by adding a D after the corresponding names of the continuous matrices. The matrix R1 is the continuous R_1-matrix of Section 2, while R1D is the descrete matrix $\tilde{R}_1$. P0 is equivalent to R_o of Section 2. The stationary solution of the Riccati equation is denoted P. The matrix C*P*CT + R2D is the covariance matrix of prediction errors and KT is the transposed filter gain K of (2.5).

The loss VLOSS is minimized in LISPID. VLOSS is equal to VLOS1 (V_1 of Section 3) if ISYS = 1, -1, 2 or -2 and equal to VLOS2 (V_2 of Section 3) if ISYS = 3 or -3. NVLOS1 and NVLOS2 are the number of contributions to each loss function, i.e. usually NP- NPRED2 ($N-p_2$ of Section 3) if NPRED2 $\geq$ 2 and NP if NPRED2 = 1. VLOSDT is the loss VLOS1 obtained without process noise. Thus VLOSS = VLOS1 = VLOSDT when output error identification is performed.

The print IEXIT = 0 obtained from NUFLET means that the minimum point has been reached according to the minimization algorithm. The corresponding print from POWBRE is: "THIS IS THE MINUMUM:".

The accuracy of the estimated parameter vector $\hat{\theta}$= TH is approximately given by

$$\text{cov}\ (\hat{\theta}) = 2 * \text{VLOS1}\ (\hat{\theta}) * \text{V2}^{-1}\ (\hat{\theta})\ /\ \text{NVLOS1} \qquad (7.1)$$

if ISYS = 1, -1, 2 or -2 and by

$$\text{cov}\ (\hat{\theta}) = \text{V2}^{-1}\ (\hat{\theta})\ /\ \text{NVLOS2} \qquad (7.2)$$

if ISYS = 3 or -3, provided that NPRED1 = NPRED2 = 1. V2 is the second derivative matrix. The estimated standard deviations STDEV are then obtained as the square root of the diagonal elements of cov. An element of STDEV is assigned the value -1 if the corresponding diagonal value of cov is negative. Remember that STDEV has meaning only if output error or maximum likelihood identifications are performed (NPRED1 = NPRED2 = 1).

The input signals are denoted U1, U2,... and the measurement signals Y1, Y2,... The outputs of the deterministic model, i.e. without process noise, are denoted YMOD1, YMOD2,... The model errors ERRMOD1 are equal to Y1-YMOD1 and EPS1 are the corresponding prediction errors. If IEPS is not equal to zero, then the corresponding prediction error has not contributed to the loss function. The different values of IEPS are explained in the program head of LISPID (see Appendix C). When IEPS ≠ 0 then YMOD1, YMOD2,..., ERRMOD1, ERRMOD2,..., EPS1, EPS2,... are usually assigned the value -1.

The line printer plots of the loss function obtained when ICHK = 1 are explained in the program head of GRDCHK (see Appendix C).

8. ACKNOWLEDGEMENTS

The design and development of the program package LISPID have been supported by the Swedish Board for Technical Development under Contracts 71-50/U33, 72-202/U137, 73 35 46, 73 41 28 U and 75-4053. This report has been written under Contract 77-5766.

The author would like to express his gratitude to Prof. Karl Johan Åström, who contributed with the basic ideas of LISPID, and to Mr. T. Essebo for many valuable improvements.

The manuscript has been expertely typed by Mrs. L. Andersson and the figures are drawn by Miss B.-M. Carlsson.

9. REFERENCES

Åström, K J (1970): Introduction to Stochastic Control Theory. Academic Press, New York.

Åström, K J, and P Eykhoff (1971): System identification - a survey. Automatica, 7, 123.

Åström, K J, C G Källström, N H Norrbin, and L Byström (1975): The identification of linear ship steering dynamics using maximum likelihood parameter estimation. Publ. No. 75, Swedish State Shipbuilding Experimental Tank, Gothenburg, Sweden.

Åström, K J, and C G Källström (1976): Identification of ship steering dynamics. Automatica, 12, 9.

Byström, L, and C G Källström (1978): System identification of linear and non-linear ship steering dynamics. Paper submitted to the 5th Ship Control Systems Symp., Annapolis, Maryland, USA.

Källström, C (1973): Computing exp (A) and $\int$ exp (As) ds. TFRT - 3053, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Källström, C G, T Essebo, and K J Åström (1976): A computer program for maximum likelihood identification of linear, multivariable stochastic systems. Proc. 4th IFAC Symp. on Identification and System Parameter Estimation, Vol. 2, pp 508-521, Tbilisi, USSR.

Mårtensson, K (1969): Linear quadratic control package. Part II - the discrete problem. TFRT - 3010, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

APPENDIX A - INPUT PARAMETERS

A table of the input parameters for LISPID is shown on the next page. The parameters which begin with I, L, M or N are integer variables, while the others are real variables. The parameter values are read from cards or card images in free format. Examples of admitted integer numbers are:

```
 0
+10
-15
```

Examples of admitted real numbers are:

```
 0.
-0.5
+10.
 5.E2
-1.E+10
+1.5E-10
```

Notice that it is possible to include an arbitrary comment after the number by use of the character "   (ASCII code).
For example:

```
10 "NPAR
1. "PAR(1) - COEFFICIENT A1
```

INPUT PARAMETERS FOR LISPID

Only NPAR values of PAR and only NTH values of ITH and SCAL must be given.

NPAR	PAR(30)	ITH(21)	SCAL(12)	NPRI(1)
NTH	PAR(31)	ITH(22)	SCAL(13)	NPRI(2)
PAR(1)	PAR(32)	ITH(23)	SCAL(14)	NPRI(3)
PAR(2)	PAR(33)	ITH(24)	SCAL(15)	NPLOT
PAR(3)	PAR(34)	ITH(25)	SCAL(16)	NP
PAR(4)	PAR(35)	ITH(26)	SCAL(17)	IT
PAR(5)	PAR(36)	ITH(27)	SCAL(18)	ISYS
PAR(6)	PAR(37)	ITH(28)	SCAL(19)	MEAS
PAR(7)	PAR(38)	ITH(29)	SCAL(20)	ISAMP
PAR(8)	PAR(39)	ITH(30)	SCAL(21)	TSAMP
PAR(9)	PAR(40)	ITH(31)	SCAL(22)	NPRED1
PAR(10)	ITH(1)	ITH(32)	SCAL(23)	NPRED2
PAR(11)	ITH(2)	ITH(33)	SCAL(24)	NX
PAR(12)	ITH(3)	ITH(34)	SCAL(25)	NU
PAR(13)	ITH(4)	ITH(35)	SCAL(26)	NY
PAR(14)	ITH(5)	ITH(36)	SCAL(27)	NOMAT(1)
PAR(15)	ITH(6)	ITH(37)	SCAL(28)	NOMAT(2)
PAR(16)	ITH(7)	ITH(38)	SCAL(29)	NOMAT(3)
PAR(17)	ITH(8)	ITH(39)	SCAL(30)	NOMAT(4)
PAR(18)	ITH(9)	ITH(40)	SCAL(31)	NOMAT(5)
PAR(19)	ITH(10)	SCAL(1)	SCAL(32)	NOMAT(6)
PAR(20)	ITH(11)	SCAL(2)	SCAL(33)	IACC
PAR(21)	ITH(12)	SCAL(3)	SCAL(34)	HH
PAR(22)	ITH(13)	SCAL(4)	SCAL(35)	NPLOTC
PAR(23)	ITH(14)	SCAL(5)	SCAL(36)	NOL
PAR(24)	ITH(15)	SCAL(6)	SCAL(37)	TS
PAR(25)	ITH(16)	SCAL(7)	SCAL(38)	EPSI
PAR(26)	ITH(17)	SCAL(8)	SCAL(39)	NKAL
PAR(27)	ITH(18)	SCAL(9)	SCAL(40)	ICHK
PAR(28)	ITH(19)	SCAL(10)	IMIN	ICR
PAR(29)	ITH(20)	SCAL(11)	LOOP	

APPENDIX B - THE LISPID FILE

This appendix shows the commands used to transfer the LISPID file from magnetic tape to disc and the element table of the LISPID file. The underlined commands are those typed by the user. It is important to delete the LISPID disc file as soon as the file is not required:

```
@DELETE  LIS*LISPID.
```

```
 ENTER USERID/PASSWORD:
>U/P*
*UNIVAC 1100 OPERATING SYSTEM  VER. 32.R2BX-D399(RSI)*
>@RUN CK,208876,CK
DUP ID, NEW ID IS CKB
NOW IS 17:42 MONDAY 24 JUL 78
>@CAT,P LIS*LISPID.,F2
READY
>@ASG,AX LIS*LISPID.
READY
>@REGISTER LIS*LISPID./60
LDC FILE REGISTRATION PROCESSOR. EXECUTION STARTED 78-07-24 AT 17:42:51

THE FILE LIS*LISPID IN PROJECT CK
HAS BEEN REGISTERED ON ACCOUNT 208876.
IT WILL BE DELETED WHEN IT HAS NOT BEEN REFERENCED FOR 60 DAYS.

END OF EXECUTION.
>@MSG,W UQLRT2,208876,3039,READ
3039/R

>@ASG,T A.,U9H,3039
READY
>@COPY,G A.,LIS*LISPID.
FURPUR 27R1 RL72-8 07/24/78 17:45:19
LIS    *LISPID(1) COPIED ON 07/24/78 AT 17:02:53
    27 BLOCKS COPIED.
EOF ENCOUNTERED ON INPUT TAPE
>@FIN




RUNID CK      ACCT 208876         PROJ CK              PRIO M&M
    0-CKB*MSG: UQLRT2,208876,3039,READ
    0 3039/R
       LOAD 3039 2/4 A -1 CKB
       SERVICE 2/4 3039   CKB
START 17:42:03 JUL 24,1978        FIN 17:48:55 JUL 24,1978
  PRIS KR    TIME: TOTAL:00:00:17.009  CPU:00:00:00.210  MEM:00:00:02.581
    10.15        CC/ER:00:00:12.066  I/O:00:00:04.733 WAIT:00:02:54.379
*TERMINAL INACTIVE*
>
```

```
"PRT,T LIS*LISPID.
FURPUR 27R1 RL72-8 07/18/78 12:06:53

LIS*LISPID(1)
FOR    MAIN(0)
REL    MAIN
FOR    RDATA(0)
REL    RDATA
FOR    SYST(0)
ELT    LISMAP(0)
REL    AXEL
REL    COSA
REL    COSDY
REL    CSGFT
REL    DATEXP
REL    DECOM
REL    DESYM
REL    DIGITS
REL    DOTLIN
REL    EIGS
REL    EXPAN
REL    FINO
REL    FONED
REL    GRASD
REL    GRDCHK
REL    IFAC/LDC
REL    IMACON/LDC
REL    IOLISP
REL    IRTTFF
REL    KALMAN
REL    LENGTH/LDC
REL    LIPLOT
REL    LISDAT
REL    LISPID
REL    LISY
REL    LODSEG
REL    LOS
REL    MCORE
REL    MIN
REL    MINFIT
REL    MPRI
REL    NORM
REL    NOUT$E/LDC
REL    NUFLET
REL    OVFLOW
REL    PARFIL
REL    POWBRE
REL    QUAD1
REL    RABC
REL    RBUFF
REL    REK
REL    RESLIS
REL    RESPLT
REL    REST
REL    RFP
REL    RIFF
REL    RITA
REL    RMACON/LDC
REL    RTTFF
REL    SAMP
REL    SCL
REL    SGAIN
REL    SIGN1
REL    SINT
```

```
REL    SOLVB
REL    SOLVS
REL    SORT
REL    STRING/LDC
REL    SYMIN
REL    SYSPRI
REL    TRANS
REL    XOPT/LDC
```

APPENDIX C - PROGRAM HEADS

The program heads of the most important subroutines are listed in this appendix:

COSA
COSDY
* DATEXP
DECOM
DESYM
EIGS
* GRDCHK
IOLISP
IRTTFF
KALMAN
* LISDAT
LISPID
MPRI
NORM
NUFLET
* OVFLOW
* PARFIL
POWBRE
RTTFF
SGAIN
SINT
SOLVB
SOLVS
SYMIN
TRANS

A star (*) means that the whole subroutine is listed. Compiled elements of the subroutines are stored on the LISPID file.

```
      SUBROUTINE COSA(A,B,NX,NU,T,IA)
C
C     COMPUTES FI AND GAMMA OF THE DISCRETE SYSTEM FROM A,B AND T,
C     WHERE A AND B ARE THE CONTINUOUS SYSTEM MATRICES AND T
C     THE SAMPLING INTERVAL.
C     THE NUMBER OF TERMS USED IN THE SERIES EXPANSION IS 8.
C     REFERENCE, C.KALLSTROM : COMPUTING EXP(A) AND
C     INTEGRAL(EXP(AS)DS) ,REPORT 7309.
C     AUTHOR, C.KALLSTROM 1972-11-16.
C     REVISED, C. KALLSTROM 1973-11-13.
C
C     A-        MATRIX OF ORDER NX*NX,
C               AT INPUT CONTAINING A OR A*T
C               AT OUTPUT CONTAINING FI.
C     B-        MATRIX OF ORDER NX*NU,
C               AT INPUT CONTAINING B OR B*T
C               AT OUTPUT CONTAINING GAMMA.
C               NOT REFERENCED IN COSA IF NU=0.
C     NX-       NUMBER OF STATES (MAX 20, MIN 1).
C     NU-       NUMBER OF CONTROL VARIABLES (MAX 20, MIN 0).
C               PUT NU=0 IF NO B.
C     T-        SAMPLING INTERVAL.
C               PUT T=-1. IF A AND B ALREADY CONTAIN A*T AND B*T, ELSE PUT
C               T EQUAL TO THE ACTUAL SAMPLING INTERVAL.
C     IA-       DIMENSION PARAMETER OF A AND B.
C
C     THE COMMON BLOCK /EX/, WHICH IS SHARED WITH SUBROUTINE EXPAN,
C     CONTAINS WORK ARRAYS./COSWK/ CONTAINS WORK ARRAYS
C
C     SUBROUTINE REQUIRED
C               NORM
C               EXPAN
C
      DIMENSION A(IA,1),B(IA,1)
C
      COMMON/EX/ KDIV,S2(20,20),S3(20,20)
C
      COMMON/COSWK/S1(20,20)
C
```

```
      SUBROUTINE COSDY(A,B,FI,GAM1,GAM2,NX,NU,TSAMP,TAU,IA)
C
C     SUBROUTINE TO TRANSFORM A CONTINUOUS LINEAR SYSTEM DX/DT=A*X(T) +
C     B*U(T)  TO A DISCRETE SYSTEM X(T+TSAMP)=FI*X(T) + GAM1*U(T) +
C     GAM2*U(T+TAU) , WHERE U IS A PIECEWISE CONSTANT CONTROL VARIABLE
C     DISCONTINUOUS AT THE TIMES K*TSAMP+TAU ,K=0,1,2,...
C
C     AUTHOR, C. KALLSTROM 1974-03-01.
C
C     A-     CONTINUOUS SYSTEM MATRIX OF ORDER NX*NX ,NOT DESTROYED.
C     B-     CONTINUOUS SYSTEM MATRIX OF ORDER NX*NU ,NOT DESTROYED.
C            NOT REFERENCED IF NU=0.
C     FI-    COMPUTED DISCRETE SYSTEM MATRIX OF ORDER NX*NX.
C     GAM1-  COMPUTED DISCRETE SYSTEM MATRIX OF ORDER NX*NU
C            NOT USED IF NU=0.
C     GAM2-  COMPUTED DISCRETE SYSTEM MATRIX OF ORDER NX*NU
C            NOT USED IF NU=0.
C     NX-    NUMBER OF STATES (MAX 20 ,MIN 1).
C     NU-    NUMBER OF INPUTS (MAX 20 ,MIN 0).
C            PUT NU=0 IF NO B.
C     TSAMP-SAMPLING INTERVAL (NO MAX ,MIN 0.).
C     TAU-   TIME LAG BETWEEN THE SAMPLING EVENT AND NEXT DISCONTINUITY
C            OF U (NO MAX ,NO MIN). NOT USED IF NU=0.
C     IA-    DIMENSION PARAMETER OF A,B,FI,GAM1 AND GAM2.
C
C     NOTE: IF TAU .GE. TSAMP ,GAM1 CONTAINS THE ORDINARY
C            DISCRETE SYSTEM MATRIX GAMMA AND GAM2 CONTAINS ZEROS,
C            IF TAU .LE. 0. ,GAM1 CONTAINS ZEROS AND GAM2 CONTAINS
C            THE ORDINARY DISCRETE SYSTEM MATRIX GAMMA.
C
C     SUBROUTINE REQUIRED
C            COSA
C               NORM
C               EXPAN
C
      DIMENSION A(IA,1),B(IA,1),FI(IA,1),GAM1(IA,1),GAM2(IA,1)
C
```

```
      SUBROUTINE DATEXP
C
C     SUBROUTINE FOR LISPID TO COMPUTE PARAMETERS FOR /INDEX/ AND
C     EXPAND PROGRAM SIZE FOR /DATA/ .
C
C     AUTHOR, T. ESSEBO 1974-04-17.
C
C     NOTE: DATEXP SHOULD BE CALLED IN THE MAIN PROGRAM AS SOON AS
C     NP,IT,NX,NU,NY,NTH IN /SYSPAR/ HAVE BEEN ASSIGNED PROPER
C     VALUES (AND BEFORE ANYTHING IS STORED IN /DATA/ ).
C
C     SUBROUTINE REQUIRED
C         MCORE
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXPO,IXXO,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      DATA MAXCOR/38000/
C
C     COMPUTE PARAMETERS FOR /INDEX/ FROM /SYSPAR/ .
C
C     NSIZE=NP*(NU+IT+4*NY+1) + 7*NX*NX + 2*NX*NU + 6*NX*NY +
C           2*NY*NU + 6*NY*NY + 2*NX + 2*NTH*NTH
C
      NX2=NX*NX
      NY2=NY*NY
      NXNY=NX*NY
      NXNU=NX*NU
      NYNU=NY*NU
C
      IXYMS=NP*NU
      IXTIM=IXYMS+NP*NY
      IXYMOD=IXTIM+NP*IT
      IXERMD=IXYMOD+NP*NY
      IXEPS=IXERMD+NP*NY
      IXIEPS=IXEPS+NP*NY
      IXA=IXIEPS+NP
      IXB=IXA+NX2
      IXC=IXB+NXNU
      IXD=IXC+NXNY
      IXR1=IXD+NYNU
      IXR12=IXR1+NX2
      IXR2=IXR12+NXNY
      IXAK=IXR2+NY2
      IXPO=IXAK+NXNY
      IXXO=IXPO+NX2
      IXAD=IXXO+NX
      IXBD=IXAD+NX2
      IXCD=IXBD+NXNU
      IXDD=IXCD+NXNY
      IXR1D=IXDD+NYNU
      IXR12D=IXR1D+NX2
      IXR2D=IXR12D+NXNY
      IXAKD=IXR2D+NY2
      IXP=IXAKD+NXNY
      IXX=IXP+NX2
```

```
      IXRR=IXX+NX
      IXPPD=IXRR+NY2
      IXDLTR=IXPPD+NX2
      IXRPD=IXDLTR+NY2
      IXRNY=IXRPD+NY2
      IXV1=IXRNY+NY2
      IXV2=IXV1+NTH*NTH
C
      NSIZE=IXV2+NTH*NTH
      NSIZ1=NSIZE-1
C
C     CHECK THAT PROGRAM EXPANSION IS POSSIBLE.
C
      IF(NSIZE-MAXCOR)10,5,5
    5 WRITE(6,100)NSIZE
  100 FORMAT(1H0,52(1H*)/24H *  COMMONBLOCK /DATA/ =,I8,
     *21H WHICH IS TOO LARGE */1X,52(1H*)/)
C
      STOP
C
   10 CALL MCORE(NSIZ1)
C
      WRITE(6,110)NSIZE
  110 FORMAT(17H0SIZE OF /DATA/ : ,I8/)
C
C     PUT ZEROS IN /DATA/
C
      DO 30 I=1,NSIZE
   30 V(I)=0.
C
      RETURN
      END
```

```
      SUBROUTINE DECOM(A,NN,IA,EPS,ISING)
C
C     COMPUTES TRIANGULAR MATRICES L AND U AND PERMUTATION MATRIX SO
C     THAT L*U=P*A, USING GAUSS ELIMINATION WITH PARTIAL PIVOTING. STORES
C     IN COMMON/IPUL30/.
C     REFERENCE, FORSYTHE-MOLER.
C     AUTHOR, PER HAGANDER 1968-09-05.
C     REVISED, CLAES KALLSTROM 1971-03-20.
C
C     A-MATRIX OF ORDER NNXNN, NOT DESTROYED.
C     NN- ORDER OF THE MATRIX (MAX 30, MIN 1).
C     IA- DIMENSION PARAMETER.
C     EPS- PIVOT TEST QUANTATY.  1E-7 SEEMS REASONABLE.
C     ISING-IS RETURNED 1 IF ANY OF THE ROWS OF A ARE ZERO.
C                       2  IF ANY PIVOT IS SMALL.
C                       0 OTHERWISE.
C     ATTENTION.  EPS=1E-7  PREVENTS SOME HORRIBLE RESULTS BUT DOES NOT
C     GUARANTUE CORRECT RESULT FOR ILLCONDITIONED MATRICES.
C
C     SUBROUTINE REQUIRED
C            NONE
C
      DIMENSION A(IA,IA)
      DIMENSION SCALES(30)
C
      COMMON/IPUL30/IPS(30),UL(30,30)
C
```

```
      SUBROUTINE DESYM(A,G,N,EPS,IRANK,IA)
C
C     DECOMPOSES POSITIVE (SEMI) DEFINITE SYMMETRIC MATRIX A INTO
C     G*GTRANSPOSE  WHERE G IS A LOWER TRIANGULAR MATRIX.  THE RANK OF A
C     IS COMPUTED AND  (N-IRANK) COLUMNS OF A MIGHT BE ZERO.
C     REFERENCE, FORSYTHE MOLER  P114.
C     AUTHOR, IVAR GUSTAVSSON 1969-01-27.
C     REVISED, CLAES KALLSTROM 1971-03-20.
C
C     A      MATRIX OF ORDER  NXN              NOT DESTROYED
C     G      MATRIX OF ORDER  NXN              OUTPUT
C     N      ACTUAL ORDER OF A AND G (NO MAX, MIN 1).
C     EPS    TEST QUANTITY.
C     IRANK  RANK OF A (AND G),  -1  IF DECOMPOSITION IMPOSSIBLE.
C     IA     DIMENSION PARAMETER.
C
C     SUBROUTINE REQUIRED
C             NONE
C
      DIMENSION A(IA,IA),G(IA,IA)
C
```

```
      SUBROUTINE EIGS(A,R,EV,N,IA,MV)
C
C     COMPUTES EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC MATRIX
C     USING THRESHOLD JACOBI METHOD.
C     REFERENCE, RALSTON AND WILF, MATHEMATICAL METHODS FOR DIGITAL
C     COMPUTERS, CHAPTER 7.
C     AUTHOR, C.KALLSTROM 1970-07-16.
C
C     A -ORIGINAL MATRIX (SYMMETRIC), DESTROYED IN COMPUTATION.
C        RESULTANT EIGENVALUES ARE DEVELOPED IN DIAGONAL OF MATRIX IN
C        DESCENDING ORDER.
C     R -RESULTANT MATRIX OF EIGENVECTORS (STORED COLUMNWISE, IN SAME
C        SEQUENCE AS EIGENVALUES).
C     EV-VECTOR CONTAINING THE EIGENVALUES IN DESCENDING ORDER.
C     N -ORDER OF MATRICES A AND R.
C     IA-DIMENSION PARAMETER.
C     MV-INPUT CODE
C        0  COMPUTE EIGENVALUES AND EIGENVECTORS.
C        1  COMPUTE EIGENVALUES ONLY (R MUST STILL APPEAR IN CALLING
C           SEQUENCE).
C     THE OFF-DIAGONAL ELEMENTS IN A ARE SET EQUAL TO 0 BEFORE RETURN.
C     THERE ARE NO MAXIMUM ORDER OF THE MATRICES A AND R.
C
C     SUBROUTINE REQUIRED
C           NONE
C
      DIMENSION A(IA,IA),R(IA,IA),EV(1)
C
```

```
      SUBROUTINE GRDCHK(FUNCT,X,N,XM,HH)
C
C     SUBROUTINE FOR CHECKING GRADIENT COMPUTATION IN NUFLET
C     AND PLOTTING (ON LINE PRINTER) THE APPEARANCE OF THE
C     FUNCTION F(X1,X2,...XN) NEAR THE POINT X(X1,X2,...XN)
C
C     AUTHOR, T. ESSEBO 1974-04-02.
C     REVISED, T.ESSEBO 1977-02-18.
C
C     FUNCT,X,N,XM,HH : THE SAME ARGUMENTS AS IN NUFLET.
C     N : MIN 1,NO MAX
C     X IS NOT CHANGED BY GRDCHK
C
C     GRDCHK WILL PRODUCE A LINE PRINTER PLOT AND ASSOCIATED
C     TEXT FOR EVERY COMPONENT OF X. (8 PLOTS PER PAGE)
C     FUNCT IS CALLED 20*N+1 TIMES
C
C     THE VERTICAL AXIS IS THE X-AXIS ,RANGING FROM -2*HH*XM(I)
C     WITH STEP 0.2*HH*XM(I) TO 2*HH*XM(I)
C     TO THE RIGHT OF THE PLOT THE SCALED FUNCTION VALUES
C     FS ARE PRINTED (RANGE: 0.0 TO 25.0)
C
C     FS=(SCALE FACTOR)*(FV(X)-MIN. VALUE)
C
C     G1,G2,G4: COMPUTED NUMERICAL GRADIENTS ( H=HH*XM(I) )
C     G1: X , X+H  USED
C     G2: X-H , X+H  USED
C     G4:  X-2H , X-H , X+H , X+2H  USED
C
      DIMENSION X(1),XM(1),LC(21,4),FV(21,4),LA(26,4),ZP(4),SCL(4),
     *ST(4),G1(4),G2(4),G4(4),NRX(4),IF1(6),IF2(5),IF3(9),IF4(8),
     *IF5(7),IF6(6),IF7(8),IF8(6),SCL1(4)
C
      DATA EPS/1.E-20/,LA/104*1H /,IBL/1H /,IAST/1H*/,IPL/1H+/,MZR/1H0/
     *,LTP/5H" FS /,LBT/5H"    /,LP/6/,
     *IF1/'(','4','(3H "*,5(5H++++*),A5))'/,
     *IF2/'(','4','(1X,28A1,F4.1))'/,
     *IF3/'(/','4','(3H FV,I2,1H=,E9.3,2H +,E9.3,3H*FS,4X))'/,
     *IF4/'(','4','(18H X-STEP=0.2*HH*XM=,E9.4,6X))'/,
     *IF5/'(','4','(2H X,I2,2H= ,E14.8,13X))'/,
     *IF6/'(','4','(6H F(X)=,E14.8,13X))'/,
     *IF7/'(','4','(4H G1=,E11.5,4H G2=,E11.5,3X))'/,
     *IF8/'(','4','(4H G4=,E11.5,18X))'/
C
      CALL FUNCT(X,N,F,D1,D2,ID,0,IER)
      IF(IER.EQ.-1)GO TO 95
C
      WRITE(LP,100)
  100 FORMAT(1H1)
      NN=0
C
    5 IF(N-NN)99,99,7
    7 NC=MIN0(4,N-NN)
C
      DO 25 I=1,NC
      NNI=NN+I
      XS=X(NNI)
      ST(I)=0.20*HH*XM(NNI)
      FV(11,I)=F
      XS1=XS
      XS2=XS
      DO 12 J=1,10
      XS1=XS1+ST(I)
```

```
      XS2=XS2-ST(I)
      X(NNI)=XS1
C
      CALL FUNCT(X,N,FV1,D1,D2,ID,0,IER)
      IF(IER.EQ.-1)GO TO 94
C
      X(NNI)=XS2
C
      CALL FUNCT(X,N,FV2,D1,D2,ID,0,IER)
      IF(IER.EQ.-1)GO TO 94
C
      FV(11+J,I)=FV1
   12 FV(11-J,I)=FV2
      X(NNI)=XS
      VMN=F
      VMX=F
      DO 15 J=1,21
      VMN=AMIN1(VMN,FV(J,I))
   15 VMX=AMAX1(VMX,FV(J,I))
      ZP(I)=VMN
      VS=VMX-VMN
      IF(VS-EPS)17,17,16
   16 VS=25./VS
      SCL1(I)=1./VS
      GO TO 18
   17 VS=0.
      SCL1(I)=0.
   18 SCL(I)=VS
      DELX=HH*XM(NNI)
      G1(I)=(FV(16,I)-FV(11,I))/DELX
      G2(I)=(FV(16,I)-FV(6,I))/(2.*DELX)
      G4(I)=(8.*(FV(16,I)-FV(6,I))-(FV(21,I)-FV(1,I)))/(12.*DELX)
      DO 20 J=1,21
      FV(J,I)=SCL(I)*(FV(J,I)-ZP(I))
   20 LC(J,I)=INT(FV(J,I)+1.50)
   25 NRX(I)=NNI
C
      ENCODE(130,LW)NC
  130 FORMAT(I1)
      IF1(2)=LW
      IF2(2)=LW
      IF3(2)=LW
      IF4(2)=LW
      IF5(2)=LW
      IF6(2)=LW
      IF7(2)=LW
      IF8(2)=LW
      WRITE(LP,IF1)(LTP,I=1,NC)
C
      DO 40 I=1,21
      MLP=IPL
      IF(MOD(I-1,5).EQ.0)MLP=IAST
      IF(I.EQ.11)MLP=MZR
      DO 32 J=1,NC
      MP=LC(I,J)
   32 LA(MP,J)=IAST
      WRITE(LP,IF2)(MLP,(LA(K,L),K=1,26),MLP,FV(I,L),L=1,NC)
      DO 35 J=1,NC
      MP=LC(I,J)
   35 LA(MP,J)=IBL
   40 CONTINUE
C
      WRITE(LP,IF1)(LBT,I=1,NC)
      WRITE(LP,IF3)(NRX(I),ZP(I),SCL1(I),I=1,NC)
```

```
      WRITE(LP,IF4)(ST(I),I=1,NC)
      WRITE(LP,IF5)(NRX(I),X(NN+I),I=1,NC)
      WRITE(LP,IF6)(F,I=1,NC)
      WRITE(LP,IF7)(G1(I),G2(I),I=1,NC)
      WRITE(LP,IF8)(G4(I),I=1,NC)
      WRITE(LP,145)
  145 FORMAT(/)
      NN=NN+4
      GO TO 5
C
   94 X(NNI)=XS
   95 WRITE(LP,195)
  195 FORMAT(15HOERROR IN FUNCT/)
   99 RETURN
      END
```

```
      SUBROUTINE IOLISP(TH,IR,IP)
C
C     SUBROUTINE TO READ DATA FROM CARD READER FOR LINEAR SYSTEM
C     PARAMETER IDENTIFICATION AND/OR PRINT DATA ON LINE PRINTER.
C
C     AUTHOR, C. KALLSTROM 1974-02-15.
C     REVISED, C. KALLSTROM 1974-03-18.
C     REVISED, T. ESSEBO 1974-04-17.
C
C     TH-    PARAMETER VECTOR OF DIMENSION NTH FOR LISPID.
C     IR-    PUT IR=0 IF NO READING.
C            PUT IR=1 TO READ DATA FROM CARD READER. IF IR=1 ZEROS ARE
C            MOVED TO A,B,C,D,R1,R12,R2,AK,XO AND PO OF COMMON BLOCK
C            /SYS/ AND TO PAR OF COMMON BLOCK /COMSY/ BEFORE THE READING.
C            PARAMETER VALUES ARE ALSO MOVED FROM PAR TO TH AND VOLD IS
C            PUT EQUAL TO 1.E10.
C     IP-    PUT IP=0 IF NO PRINTING.
C            PUT IP=1 TO PRINT DATA.
C            PUT IP=2 TO PRINT THE SAME DATA AS FOR IP=1 ,BUT IF ISYS=2
C            OR -2 AND SUBROUTINE SGAIN IS CALLED IN SYST TO COMPUTE THE
C            DISCRETE ,STATIONARY FILTER GAIN AK OR IF ISYS=3,-3,4 OR -4
C            SOME MORE DATA IS PRINTED.
C
C     DATA TO BE READ IF IR=1 (IN FREE FORMAT):
C     NPAR
C     NTH
C     PAR(1)
C       .
C     PAR(NPAR)
C     ITH(1)
C       .
C     ITH(NTH)
C     SCAL(1)
C       .
C     SCAL(NTH)
C     IMIN
C     IF IMIN<0 2 PARAMETERS WILL BE READ:
C     IMIN=-1 (NUFLET)
C     HZ
C     EPZ
C     IMIN=-2 (POWBRE)
C     DIST
C     TEPS
C     LOOP
C     NPRI(1)
C     NPRI(2)
C     NPRI(3)
C     NPLOT
C     NP
C     IT
C     ISYS
C     MEAS
C     ISAMP
C     TSAMP
C     NPRED1
C     NPRED2
C     NX
C     NU
C     NY
C     NOMAT(1)
C       .
C     NOMAT(6)
C     IACC
```

```
C       HH
C       NPLOTC
C       NOL
C       TS
C       EPSI
C       NKAL
C       ICHK
C       ICR
C
C       IF ICR=0 NO MORE DATA WILL BE READ.
C       IF ICR=1 ,FOLLOWING DATA WILL ALSO BE READ:
C       NMAX
C       EPST
C       EPSK
C       CHISQ
C       IPLOT
C       ITEXT
C       IIY
C       SX
C       SY
C
C       ATTENTION: IT IS ONLY SENSIBLE TO CALL IOLISP WITH IP=2 ,IF
C       LISPID EARLIER IS CALLED AT LEAST ONCE.
C
C       SUBROUTINE REQUIRED
C            MPRI
C            RTTFF
C            LISDAT
C
        DIMENSION TH(1)
C
        COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
       *NX,NU,NY,NTH,NOMAT(6)
C
        COMMON/DATA/V(1)
C
        COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
       *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
       *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
       *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
        COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
       *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
       *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
       *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
        COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
       *IERRSY
C
        COMMON/KAL1/DUMM1(60),RCOV(20,20),AKT(20,20),DUMM2(2)
C
```

```
      INTEGER FUNCTION IRTTFF(IC)
C
C     READS ONE INTEGER NUMBER FROM STANDARD INPUT IN FREE FORMAT.
C     IC .NE. 0 FORCES READING OF A NEW LINE AND IC IS SET TO 0
C     IC .EQ. 0 - THE PREVIOUSLY READ LINE IS SCANNED AND IF EMPTY
C                 A NEW LINE IS READ
C     IF EOF ENCOUNTERED OF INPUT IC IS SET TO -1
C
C     AUTHOR, T. ESSEBO 1975-02-19
C     REVISED T. ESSEBO 1977-03-06
C
C     SUBROUTINE REQUIRED
C          RBUFF
C          IFAC
C          RIFF
C
      LOGICAL EOF
      DIMENSION RRES(2)
      COMMON/RTFCOM/IP,BUFF(20)
      DATA MAXCH/8/,LUN/5/
C
```

```
      SUBROUTINE KALMAN(A,B,C,D,R1,R12,R2,P,X,U,Y,NX,NU,NY,NOMAT,
     *   ICONT,NOXTT,EPS,CHISQ,NTEST,IA,IC)
C
C
C     THIS SUBROUTINE ITERATES THE DISCRETE KALMAN BUCY FILTERING
C     EQUATIONS
C
C     YHAT=C*X(T/T-1)+D*U
C     E=Y-YHAT
C     R=C*P(T/T-1)*CT+R2
C     AK=(A*P(T/T-1)*CT+R12)*R(-1)
C     X(T/T)=X(T/T-1)+P(T/T-1)*CT*R(-1)*E   (OPTIONALLY)
C     X(T+1/T)=A*X(T/T-1)+B*U+AK*E
C     P(T+1/T)=A*P(T/T-1)*AT+R1-AK*(C*P(T/T-1)*AT+R21)
C
C
C     IF THE MEASUREMENT IS REJECTED OR THE COVARIANCE MATRIX OF
C     RESIDUALS R IS SINGULAR, AK IS NOT COMPUTED AND THE STATES
C     X AND P ARE UPDATED AS
C
C     X(T/T)=X(T/T-1)   (OPTIONALLY)
C     X(T+1/T)=A*X(T/T-1)+B*U
C     P(T+1/T)=A*P(T/T-1)*AT+R1
C
C
C     IF R IS NONSINGULAR, THE TEST QUANTITY
C             TEST=ET*R(-1)*E
C     CAN BE COMPUTED AND COMPARED TO CHISQ, AND THE CONTRIBUTION TO
C     THE LOGLIKELIHOOD FUNCTION
C             DELTAV=ET*R(-1)*E+LOG(DET R)
C     CAN BE COMPUTED.
C
C
C     THE COMPUTATIONS ARE CONTROLLED BY THE PARAMETER ICONT;
C
C     ICONT=1 : STRAIGHT FORWARD COMPUTATION, YHAT,E,R AND, IF R IS
C               NONSINGULAR, AK,TEST AND DELTAV ARE COMPUTED.
C     ICONT=2 : MEASUREMENT REJECTED, YHAT,E,R AND, IF R IS
C               NONSINGULAR, TEST AND DELTAV ARE COMPUTED.
C     ICONT=3 : MEASUREMENT REJECTED, YHAT,E,R,AK,TEST AND DELTAV ARE
C               NOT COMPUTED.
C     ICONT=4 : MEASUREMENT REJECTED IF TEST GREATER THAN CHISQ OR
C               R SINGULAR, ELSE MEASUREMENT ACCEPTED. YHAT,E,R AND,
C               IF R IS NONSINGULAR, TEST AND DELTAV ARE COMPUTED.
C               IF MEASUREMENT IS ACCEPTED, AK IS COMPUTED, TOO.
C
C
C     REFERENCE, ASTROM: INTRODUCTION TO STOCHASTIC CONTROL THEORY,
C               ACADEMIC PRESS, 1970.
C     AUTHOR, C.KALLSTROM 1973-03-15.
C
C     A-        DYNAMICS MATRIX OF ORDER NX*NX.
C     B-        INPUT MATRIX OF ORDER NX*NU. NOT REFERENCED IN KALMAN
C               IF NU=0 OR NOMAT(1)=0.
C     C-        OUTPUT MATRIX OF ORDER NY*NX. NOT REFERENCED IN KALMAN
C               IF NOMAT(2)=0.
C     D-        MATRIX OF ORDER NY*NU. NOT REFERENCED IN KALMAN
C               IF NU=0 OR NOMAT(3)=0.
C     R1-       STATE NOISE COVARIANCE MATRIX OF ORDER NX*NX.
C               NOT REFERENCED IN KALMAN IF NOMAT(4)=0.
C     R12-      STATE NOISE / MEASUREMENT ERROR COVARIANCE MATRIX OF
C               ORDER NX*NY. NOT REFERENCED IN KALMAN IF NOMAT(5)=0.
C     R2-       MEASUREMENT ERROR COVARIANCE MATRIX OF ORDER NY*NY.
```

```
C               NOT REFERENCED IN KALMAN IF NOMAT(6)=0.
C       P-      COVARIANCE MATRIX OF ONE STEP AHEAD PREDICTION
C               ERROR P(T/T-1) OF ORDER NX*NX, RETURNED CONTAINING
C               P(T+1/T). P IS SYMMETRIZED BEFORE RETURN.
C       X-      ONE STEP AHEAD STATE ESTIMATE VECTOR X(T/T-1) OF
C               DIMENSION NX, RETURNED CONTAINING X(T+1/T).
C       U-      CONTROL VECTOR OF DIMENSION NU, NOT CHANGED.
C               NOT REFERENCED IN KALMAN IF NU=0 OR IF NOMAT(1)=0 AND
C               NOMAT(3)=0.
C       Y-      MEASUREMENT VECTOR OF DIMENSION NY, NOT CHANGED.
C               NOT REFERENCED IN KALMAN IF ICONT=3.
C       NX-     NUMBER OF STATES (MAX 20, MIN 1).
C       NU-     NUMBER OF CONTROL VARIABLES (NO MAX, MIN 0).
C               PUT NU=0 IF NO CONTROL SIGNAL IS APPLIED.
C       NY-     NUMBER OF MEASUREMENTS (MAX 20, MIN 1).
C       NOMAT-  INPUT VECTOR OF DIMENSION 6.
C               PUT NOMAT(1)=0 IF NO B, ELSE PUT NOMAT(1)=1
C               PUT NOMAT(2)=0 IF NO C, ELSE PUT NOMAT(2)=1
C               PUT NOMAT(3)=0 IF NO D, ELSE PUT NOMAT(3)=1
C               PUT NOMAT(4)=0 IF NO R1, ELSE PUT NOMAT(4)=1
C               PUT NOMAT(5)=0 IF NO R12, ELSE PUT NOMAT(5)=1
C               PUT NOMAT(6)=0 IF NO R2, ELSE PUT NOMAT(6)=1
C               NOMAT(1) AND NOMAT(3) ARE NOT USED IF NU=0.
C       ICONT-  CONTROL PARAMETER (SEE ABOVE), (MAX 4, MIN 1).
C       NOXTT-  PUT NOXTT=0 IF NO COMPUTATION OF X(T/T), ELSE PUT NOXTT=1.
C       EPS-    THE TEST QUANTITY EPS*NORM(R) IS USED TO DECIDE IF
C               R IS SINGULAR.
C       CHISQ-  TEST QUANTITY TO DECIDE IF MEASUREMENT VECTOR IS
C               SUFFICIENTLY CLOSE TO EXPECTED VALUE. NOT REFERENCED
C               IN KALMAN IF ICONT=3.
C       NTEST-  INDICATOR RETURNED
C               0 IF NO TEST OF RESIDUALS IS PERFORMED
C               1 IF RESIDUALS SUFFICIENTLY SMALL (TEST LESS THAN
C                 OR EQUAL TO CHISQ)
C               2 IF RESIDUALS TOO LARGE (TEST GREATER THAN CHISQ)
C               3 IF R IS SINGULAR
C       IA-     DIMENSION PARAMETER OF A,B,R1,R12 AND P.
C       IC-     DIMENSION PARAMETER OF C,D AND R2.
C
C       NOTE 1: IT IS NOT NECESSARY TO DIMENSION A MATRIX OR A VECTOR,
C       WHICH IS NOT REFERENCED IN KALMAN, IN THE CALLING PROGRAM.
C
C       NOTE 2: THE SYMMETRIC MATRICES OF THE CALL (R1,R2 AND P) MUST
C       BE COMPLETE.
C
C       NOTE 3: THE COMMON BLOCK /KAL1/ CONTAINS VARIABLES SOMETIMES
C       COMPUTED ACCORDING TO THE DESCRIPTION ABOVE. XTT CONTAINS X(T/T)
C       AND AKT CONTAINS AK(TRANSPOSED), IF THEY ARE COMPUTED.
C
C       NOTE 4: THE COMMON BLOCK /KAL2/ CONTAINS WORK ARRAYS.
C
C       SUBROUTINE RQUIRED
C               NORM
C               DESYM
C               SOLVS
C
        DIMENSION A(IA,1),B(IA,1),C(IC,1),D(IC,1),R1(IA,1),R12(IA,1),
       *   R2(IC,1),P(IA,1),X(1),U(1),Y(1),NOMAT(1)
C
        COMMON /KAL1/ YHAT(20),E(20),XTT(20),R(20,20),AKT(20,20),TEST,
       *   DELTAV
C
        COMMON /KAL2/ S1(20,20),S2(20,20),RIE(20)
C
```

```
      SUBROUTINE LISDAT
C
C     SUBROUTINE TO SET STANDARD VALUES IN /LISCON/ .
C
C     AUTHOR, T. ESSEBO 1974-05-07.
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
      LOOP=1
      NPRI(1)=1
      NPRI(2)=1
      NPRI(3)=1
      NPLOT=0
      ITRAN=0
      NMAX=500
      EPST=1.E-6
      EPSK=1.E-9
      CHISQ=6.
      IMIN=1
      DFN=-0.2
      HZ=1.E-3
      EPZ=1.E-5
      MODE=1
      MAXFN=500
      IPRINT=1
      DIST=1.
      SCALX=1.
      TEPS=1.E-6
      NSTP=1
      ILLC=1
      IPRIN=2
      IND=8
      IF=0
      IPLOT=1
      ITEXT=1
      IIY=1
      SX=16.
      SY=6.
      IDH=0
      HH=0.01
      IACC=0
      NPLOTC=2
      NOL=10
      ICHK=0
      DO 5 I=1,20
      INU(I)=1
    5 INY(I)=1
      DO 7 I=1,50
    7 XM(I)=1.
      RETURN
      END
```

```
      SUBROUTINE LISPID(SYST,TH,IERR)
C
C     ADMINISTRATION SUBROUTINE FOR LINEAR SYSTEM PARAMETER
C     IDENTIFICATION.
C     AUTHOR, C. KALLSTROM 1973-09-12.
C     REVISED, C. KALLSTROM 1977-02-18.
C
C SYST- A SUBROUTINE XXX(TH,NTH,VLOSS,DUM1,DUM2,IDUM,ICONT,IER),
C       WHICH COMPUTES THE LOSS VLOSS FOR THE PARAMETER VECTOR TH OF
C       DIMENSION NTH.THIS SUBROUTINE THE USER MUST SUPPLY. FROM
C       LISPID XXX IS ALWAYS CALLED WITH ICONT=0,WITH ONE EXCEPTION:
C       IMMEDIATELY AFTER THE MINIMIZING XXX IS CALLED ONCE WITH
C       ICONT=-1. THIS COULD BE USED TO SAVE THE FINAL PARAMETER
C       VECTOR.  PUT IER=0 IN XXX IF
C       THE COMPUTATIONS ARE OK, PUT IER=-1 TO TERMINATE THE
C       MINIMIZING.
C TH-   PARAMETER VECTOR OF DIMENSION NTH, AT INPUT CONTAINING
C       ESTIMATED MINIMUM POINT. AT RETURN CONTAINING COMPUTED
C       MINIMUM POINT ,IF LOOP .GT. 0 AND IF IERR IS RETURNED EQUAL
C       TO 0.
C IERR- ERROR PARAMETER,ALWAYS PRINTED IF .NE. 0 :
C       IERR=0 IF OK.
C       IERR=1 IF NTH AND/OR LOOP HAS AN ILLEGAL VALUE.
C       IERR=2 IF ONE OR MORE VARIABLES OF THE COMMON BLOCKS /DATA/
C       AND /SYSPAR/ HAVE ILLEGAL VALUES.
C       IERR=3 IF TROUBLES TO PLOT CURVES (NPLOT MUST BE EQUAL TO 1).
C       NO PLOTTING IS INITIATED.
C       IERR=4 IF THE SAME TROUBLES AS FOR IERR=3,BUT THE PLOTTING
C       IS STARTED.
C       IERR=5 IF IER IN THE CALL OF XXX IS RETURNED -1.
C       IERR=6 IF THE COMPUTATIONS OF GRADIENT G AND SECOND DERIVATIVE
C       MATRIX V2 HAVE FAILED, OR IF V2 IS SINGULAR.
C       IERR=7 IF BOTH IERR=5 AND IERR=6.
C
C     DESCRIPTION OF THE COMMON BLOCKS:
C
C     /SYSPAR/
C     MUST BE ASSIGNED VALUES BY THE USER.
C NP-   NUMBER OF SAMPLE EVENTS (MAX 2000,MIN 2).
C IT-   PUT IT=1 IF THE TIMES FOR THE SAMPLE EVENTS ARE SUPPLIED IN
C       /DATA/  ELSE PUT IT=0.
C ISYS- DESCRIBES THE SYSTEM MODEL TO BE USED.
C       ISYS .GT. 0 MEANS A CONTINUOUS MODEL,ISYS .LT. 0 MEANS A
C       DISCRETE MODEL.
C       ISYS=1,-1 : ONLY MEASUREMENT NOISE.
C       ISYS=2,-2 : MEASUREMENT NOISE,AND STATE NOISE MODELLED BY
C       FIX GAIN AK.
C       ISYS=3,-3 : MEASUREMENT AND STATE NOISE MODELLED BY COVARIANCE
C       MATRICES R1,(R12) AND R2.
C       ISYS=4,-4 : AS ISYS=3,-3 BUT THE MODEL IS ONLY SIMULATED AND
C       THE RESIDUALS ARE TESTED AGAINST CHISQ.
C MEAS- PUT MEAS=0 IF INSTANTANEOUS MEASUREMENTS,PUT MEAS=1 IF
C       INTEGRATING MEASUREMENTS.
C       IF MEAS=1 ,THE VALUE YMS(1,*) IN /DATA/ MUST BE THE
C       INTEGRATED VALUE FROM TIME1 TO TIME2,AND SO ON.
C ISAMP-DESCRIBES THE SAMPLE INTERVAL.
C       ISAMP=1 : CONSTANT SAMPLE INTERVAL.
C       ISAMP=2 : CONSTANT SAMPLE INTERVAL, BUT SOME MEASUREMENTS
C       ARE MISSING. IT MUST BE EQUAL TO 1.
C       ISAMP=3 : VARIABLE SAMPLE INTERVAL. THE PARAMETER IT MUST BE
C       EQUAL TO 1 IF ISYS .GT. 0.
C TSAMP-SAMPLE INTERVAL TO BE USED IF IT=0 .PUT TSAMP=-1. TO INDICATE
C       THAT THE SAMPLE INTERVAL MUST BE COMPUTED FROM TIM IN /DATA/ .
```

```
C NPRED1-
C NPRED2-THE LOSS FUNCTION VALUE IS OBTAINED AS A MEAN VALUE OF LOSSES
C        WHEN PREDICTING NPRED1,NPRED1+1, ... ,NPRED2 STEPS AHEAD.
C        NPRED1:(MAX NPRED2,MIN 1) .NPRED2: (NO MAX,MIN NPRED1).
C        PUT NPRED1=NPRED2=1 TO OBTAIN MAXIMUM LIKELIHOOD ESTIMATES.
C NX-    NUMBER OF STATES (MAX 20,MIN 1).
C NU-    NUMBER OF INPUT SIGNALS (MAX 20,MIN 0).
C        PUT NU=0 IF NO INPUT SIGNAL IS APPLIED.
C NY-    NUMBER OF MEASUREMENT SIGNALS (MAX 20,MIN 1)
C NTH-   NUMBER OF PARAMETERS IN THE OPTIMIZATION.
C NOMAT-VECTOR OF DIMENSION 6 DESCRIBING THE SYSTEM AND COVARIANCE
C        MATRICES. PUT NOMAT(1)=0 IF NO B,ELSE PUT NOMAT(1)=1.
C        PUT NOMAT(2)=0 IF NO C, ELSE PUT NOMAT(2)=1.
C        PUT NOMAT(3)=0 IF NO D, ELSE PUT NOMAT(3)=1.
C        PUT NOMAT(4)=0 IF NO R1, ELSE PUT NOMAT(4)=1
C        PUT NOMAT(5)=0 IF NO R12, ELSE PUT NOMAT(5)=1.
C        PUT NOMAT(6)=0 IF NO R2, ELSE PUT NOMAT(6)=1.
C
C      /DATA/ AND /INDEX/
C      /DATA/ CONSISTS OF VECTOR V IN WHICH THE MEASUREMENTS
C      ,SYSTEM MATRICES AND RESULTING OUTPUTS ARE STORED IN
C      CONSECUTIVE ORDER. THE RELATIVE ADDRESSES OF THE DIFFERENT
C      VECTORS AND MATRICES OF /DATA/ ARE STORED IN /INDEX/.
C      SUBROUTINE DATEXP WILL COMPUTE THE PARAMETERS OF /INDEX/
C      FROM /SYSPAR/ AND CREATE THE NECESSARY AREA NEEDED IN
C      /DATA/. A CALL OF DATEXP MUST BE MADE (PREFERABLY IN THE
C      MAIN PROGRAM) BEFORE ANYTHING CAN BE STORED IN /DATA/.
C      THE ORGANIZATION OF /DATA/ :
C      UIN(NP,NU)      -INPUT SIGNALS
C      YMS(NP,NY)      -MEASUREMENTS
C      TIM(NP)         -TIMES FOR SAMPLE EVENTS (IF IT=1)
C      YMOD(NP,NY)     -OUTPUT FROM DETERMINISTIC MODEL
C      ERRMOD(NP,NY)   -MODEL ERRORS
C      EPS(NP,NY)      -RESIDUALS
C      IEPS(NP)        -INTEGER VECTOR DESCRIBING THE RESIDUALS
C        IEPS(1)=0 IF OK,I.E. THE RESIDUAL AT TIME1 HAS CONTRIBUTED TO
C        THE LOSS FUNCTION.
C        IEPS(1)=1 IF THE RESIDUAL AT TIME1 HAS NOT CONTRIBUTED TO THE
C        LOSS FUNCTION.
C        IEPS(1)=2 IF RESIDUAL TOO LARGE WHEN TESTING AGAINST CHISQ.
C        (ONLY USED WHEN ISYS=4,-4).
C        IEPS(1)=3 IF THE COVARIANCE MATRIX OF RESIDUALS IS SINGULAR
C        WHEN COMPUTING THE CONTRIBUTION TO THE LOSS FUNCTION IN
C        SUBROUTINE KALMAN.
C      A(NX,NX),B(NX,NU),C(NY,NX),D(NY,NU),R1(NX,NX),R12(NX,NY),
C      R2(NY,NY),AK(NX,NY),P0(NX,NX),X0(NX) - SYSTEM AND
C      COVARIANCE MATRICES ,INITIAL ERROR COVARIANCE MATRIX
C      AND INITIAL STATE
C      AD(NX,NX),BD(NX,NU),....,AKD(NX,NY),P(NX,NX),X(NX) -USED
C      AS INTERNAL STORAGE (MAINLY IN SINT)
C      RR(NY,NY),PPD(NX,NX),DELTAR(NY,NY),RPD(NY,NY),RNY(NY,NY),
C      V1(NTH,NTH),V2(NTH,NTH) -USED AS STORAGE AND WORK
C      ARRAYS IN DIFFERENT PARTS OF THE PROGRAM.
C
C      THE USER MUST SUPPLY THE VALUES OF UIN,YMS ,(TIM).
C      THE SYSTEM AND COVARIANCE MATRICES A,B,..AK AND P0,X0
C      MUST ALSO BE ASSIGNED VALUES THAT MAY DEPEND ON THE
C      PARAMETER VECTOR TH. THE DEPENDENCE MUST BE SUPPLIED
C      IN SUBROUTINE XXX (OR IN A ROUTINE CALLED BY XXX).
C
C      THE RELATIVE ADDRESS OF A MATRIX IN /DATA/ IS GIVEN BY
C      A POINTER IN /INDEX/ WITH THE SAME NAME AS THE MATRIX
C      BUT PREFIXED WITH IX. THIS POINTER POINTS TO THE CELL IN
C      V BEFORE THE FIRST CELL OF THE MATRIX (OR VECTOR).
```

```
C      EX.  COVARIANCE MATRIX R12 HAS ADDRESS IXR12+1 IN V.
C      MATRICES ARE STORED COLUMN-WISE IN V.
C
C      BY USING THE UNIVAC FORTRAN V FACILITY  DEFINE PROCEDURES
C      IT IS POSSIBLE TO REFER TO MATRICES AND VECTORS OF
C      /DATA/ AS IF THEY WERE DECLARED IN A DIMENSION STATEMENT.
C      A DEFINE STATEMENT MUST APPEAR BEFORE THE FIRST EXECUTABLE
C      STATEMENT OF A PROGRAM.
C      EX.  MATRIX R12
C      DEFINE R12(I,J)=V(IXR12+NX*J-NX+I)
C      .
C      .
C      R12(I,3)=FX(I)
C      .
C      XP=R12(3,3)
C
C      EXCEPTION: DEFINE PROCEDURE NAMES MUST NOT APPEAR AS
C      FORMAL ARGUMENTS IN SUBROUTINE CALLS ,THUS:
C      CALL SUB(V(IXR12+1),NX,NY)
C
C      FOR FURTHER INFORMATION SEE: UNIVAC 1100 FORTRAN V
C      MANUAL ,P. 4-2
C
C      /LISP/
C      CONTAINS RESULTS FROM THE COMPUTATIONS.
C VLOS1-VLOS1=DET(SUM(EPS*EPST))/NVLOS1, ALWAYS COMPUTED. USED AS
C         STANDARD LOSS IF ISYS=1,-1,2,-2.
C NVLOS1-NUMBER OF CONTRIBUTIONS TO VLOS1.
C VLOS2-VLOS2=-LOG(L(TH,R))/NVLOS2 ,ONLY COMPUTED AND USED AS STANDARD
C         LOSS IF ISYS=3,-3,4,-4.
C NVLOS2-NUMBER OF CONTRIBUTIONS TO VLOS2.
C VLOSS-OBTAINED MINIMUM LOSS.
C VLOSDT-OBTAINED LOSS WHEN SIMULATING THE DETERMINISTIC MODEL.
C G-      GRADIENT VECTOR OF DIMENSION NTH.
C STDEV-VECTOR OF DIMENSION NTH CONTAINING ESTIMATED
C         STANDARD DEVIATIONS.
C      IABSIS,...,NVLDT1- INTERNAL VARIABLES.
C
C      /LISCON/
C      THE VARIABLES HAVE STANDARD VALUES ASSIGNED BY A CALL OF
C      SUBROUTINE LISDAT ,BUT THESE VALUES CAN BE OVERROLLED BY THE USER.
C LOOP- NUMBER OF CALLS TO THE MINIMIZING ALGORITHM (NO MAX,MIN -1).
C         PUT LOOP=0 FOR SIMULATON ONLY. PUT LOOP=-1 TO PRINT AND/OR
C         PLOT DATA IN /DATA/. IN THIS CASE ONLY THE
C         FOLLOWING INPUT ARGUMENTS MUST HAVE VALUES:LOOP,NPRI(3),NPLOT,
C         NP,IT,ISAMP,NU,NY AND TIM,UIN,YMS IN /DATA/. STANDARD:1
C NPRI- VECTOR OF DIMENSION 3. PUT NPRI(1)=1 TO PRINT SYSTEM AND
C         COVARIANCE MATRICES FOR THE INITIAL PARAMETER VECTOR,ELSE PUT
C         NPRI(1)=0. PUT NPRI(2)=1 TO PRINT SYSTEM AND COVARIANCE
C         MATRICES FOR THE FINAL PARAMETER VECTOR, ELSE PUT NPRI(2)=0.
C         PUT NPRI(3)=1 TO PRINT INPUT SIGNALS,MEASUREMENTS,MODEL
C         OUTPUTS ,MODEL ERRORS AND RESIDUALS ,ELSE PUT NPRI(3)=0.
C      STANDARD: 1,1,1
C NPLOT-PUT NPLOT=1 TO PLOT CURVES ON PLOTTER,ELSE PUT NPLOT=0.
C      PUT NPLOT=-1 IF THE MEASUREMENTS ARE TO BE PLOTTED AS DOTS.
C      STANDARD:0
C ITRAN-ONLY USED IF ISYS=1,2 AND MEAS=0. PUT ITRAN=1 TO SAMPLE THE
C         MATRICES A AND B BY SUBROUTINE TRANS INSTEAD OF SUBROUTINE
C         COSA. STANDARD: 0.
C NMAX- MAXIMUM NUMBER OF TERMS USED IN SUBROUTINE TRANS.STANDARD: 500.
C EPST- TEST QUANTITY USED IN SUBROUTINE TRANS. STANDARD: 1.E-6.
C EPSK- TEST QUANTITY USED IN SUBROUTINE KALMAN.STANDARD: 1.E-9.
C CHISQ-ONLY USED IF ISYS=4,-4 .TEST QUANTITY USED IN SUBROUTINE KALMAN
C         TO TEST IF RESIDUAL TOO LARGE.STANDARD: 6.0.
```

```
C IMIN- DECIDES WHICH MINIMIZING ALGORITHM TO BE USED.
C       IMIN=1 NUFLET (STANDARD)
C       IMIN=2 POWBRE
C DFN,HZ, ... ,XM(50) - PARAMETER VALUES FOR NUFLET
C DIST, .. ,IPRIN -PARAMETERS FOR POWBRE
C IND-  NUMBER OF COLUMNS PER PAPER WIDTH WHEN PRINTING VECTORS AND
C       MATRICES BY SUBROUTINE MPRI. STANDARD: 8.
C IF-   IF=0 MEANS FORMAT G16.8 WHEN PRINTING VECTORS AND MATRICES BY
C       SUBROUTINE MPRI, IF=1  MEANS FORMAT E16.8.STANDARD: 0.
C IPLOT-IPLOT=0 MEANS ORDINARY PLOT WHEN PLOTTING THE INPUT SIGNALS BY
C       SUBROUTINE RITA, IPLOT=1 MEANS HISTOGRAM PLOT.STANDARD: 1.
C ITEXT-ITEXT=0 MEANS NO TEXT WHEN PLOTTING CURVES ,ITEXT=1 MEANS
C       STANDARD TEXT. STANDARD: 1.
C IIY-  IIY=1 MEANS THAT BOTH MEASUREMENTS AND MODEL OUTPUTS ARE
C       PLOTTED IN THE SAME DIAGRAM, IIY=0 MEANS THAT ONLY MODEL
C       OUTPUTS ARE PLOTTED. STANDARD: 1.
C SX-   LENGTH OF X-AXIS IN CM WHEN PLOTTING CURVES (NO MAX,MIN 2.)
C       STANDARD: 16.
C SY-   LENGTH OF Y-AXIS IN CM WHEN PLOTTING CURVES (MIN 2.). TOTAL
C       WIDTH OF THE PLOT IN Y-DIRECTION IS ALWAYS (3*SY + 4) CM.
C       STANDARD: 6.
C IDH-  IDH=0 MEANS THAT THE STEP LENGTHS WHEN COMPUTING G AND V2 BY
C       SUBROUTINE GRASD ARE CHOSEN AS HH*TH ,IDH=1 MEANS THAT THE
C       STEP LENGTHS MUST BE SUPPLIED IN VECTOR DH. STANDARD: 0.
C HH-   SEE ABOVE. STANDARD: 0.01.
C DH-   SEE ABOVE. IF IDH=0 ,DH CONTAINS THE COMPUTED STEP LENGTHS.
C IACC- IACC=1 MEANS THAT G AND V2 ARE COMPUTED WITH ACCURACY
C       ORDO(DH**2), IACC=2 MEANS ACCURACY ORDO(DH**4).
C       IACC=0 MEANS NO COMPUTATION OF G AND V2 .STANDARD:0.
C NPLOTC:0 NOTHING PRINTED OR PLOTTED IN RESLIS
C      :1 TEST QUANTITIES PRINTED
C      :2 1+CORRELATIONS PRINTED
C      :3 1+CORRELATIONS PLOTTED
C      :4 1+2+3
C     STANDARD: 2
C NOL-  NUMBER OF TIME LAGS (MAX 50,MIN 0).STANDARD:10
C INU-  INTEGER VECTOR OF DIMENSION 20. PUT INU(I)=1 IF INPUT
C       I IS TO BE USED WHEN COMPUTING CROSS CORRELATIONS,ELSE PUT
C       INU(I)=0.
C INY-  INTEGER VECTOR OF DIMENSION 20 . PUT INY(K)=1 IF OUTPUT
C       I IS TO BE USED IN RESLIS ,ELSE PUT INY(I)=0.
C     STANDARD VALUES OF INU AND INY:1,1,...1,1
C
C     NOTE 1: THE PLOTTING CAN ALSO BE CONTROLLED BY THE COMMON BLOCK
C     /RITFIG/ DESCRIBED IN SUBROUTINE RITA.
C
C     NOTE 2: THE EXTERNAL SYMBOLS PLCOM,PLTSEG AND RESSEG ARE
C     NAMES OF PROGRAM SEGMENTS THAT ARE LOADED INTO MAIN STORAGE
C     BY REQUEST OF LISPID ,USING THE ROUTINE LODSEG .
C
C     SUBROUTINES REQUIRED
C     (SYST)
C        SINT
C           SAMP
C              COSA
C                 NORM
C                 EXPAN
C           TRANS
C              NORM
C           LISY
C           KALMAN
C              NORM
C              DESYM
C              SOLVS
```

```
C           LOS
C              NORM
C              DESYM
C     MPRI
C     SYSPRI
C        MPRI
C     NUFLET
C        (SYST)
C     POWBRE (WITH SUBROUTINES)
C        (SYST)
C     LIPLOT
C        SCL
C        RITA
C           AXEL
C           DOTLIN
C     GRASD
C        (SYST)
C     NORM
C     SYMIN
C     EIGS
C     RESLIS
C        RESPLT
C        REST
C           CSGFT
C              FINO
C           DESYM
C           SOLVS
C     LODSEG
C
      DIMENSION TH(1),NOLD(9),EVAL(40),INTV(1)
C
      LOGICAL ILLCO
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/LISP/VLOS1,NVLOS1,VLOS2,NVLOS2,VLOSS,VLOSDT,G(40),
     *STDEV(40),IABSIS,IM2,ICOS,ML,NOM(6),NOTRAN(6),NRR,NVLDT1
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
      DEFINE IEPS(I)=INTV(IXIEPS+I)
      DEFINE TIM(I)=V(IXTIM+I)
      DEFINE YMS(I,J)=V(IXYMS+NP*J-NP+I)
      DEFINE YMOD(I,J)=V(IXYMOD+NP*J-NP+I)
      DEFINE EPS(I,J)=V(IXEPS+NP*J-NP+I)
      DEFINE V1(I,J)=V(IXV1+NTH*J-NTH+I)
      DEFINE COV(I,J)=V(IXDLTR+NY*J-NY+I)
      DEFINE RNY(I,J)=V(IXRNY+NY*J-NY+I)
      DEFINE RPD(I,J)=V(IXRPD+NY*J-NY+I)
      DEFINE RR(I,J)=V(IXRR+NY*J-NY+I)
C
      EQUIVALENCE (INTV,V)
C
```

```
      SUBROUTINE MPRI (A,M,N,IA,IND,IFORM,IERR)
C
C     SUBROUTINE TO PRINT MATRICES COLUMNWISE ON LINE PRINTER.
C     NO EMPTY LINE IS PROVIDED ABOVE THE FIRST ROW OF THE MATRIX.
C     AUTHOR, C.KALLSTROM 1972-03-21.
C
C     A-       MATRIX OF ORDER M*N TO BE PRINTED, NOT DESTROYED.
C     M-       NUMBER OF ROWS IN A (NO MAX,MIN 1).
C     N-       NUMBER OF COLUMNS IN A (NO MAX,MIN 1).
C     IA-      DIMENSION PARAMETER OF A.
C     IND-     ABS(IND) IS SUPPOSED TO BE THE NUMBER OF COLUMNS
C              PER PAPER WIDTH. PUT IND POSITIVE IF NORMAL PRINTING.
C              PUT IND NEGATIVE IF THE ROWS AND COLUMNS OF A ARE TO BE
C              NUMBERED WHEN PRINTING A.
C              IND: (MAX 8, MIN -6), IND .NE. 0.
C     IFORM-   PUT IFORM=0 IF FORMAT G16.8 IS TO BE USED WHEN PRINTING A.
C              PUT IFORM=1 IF FORMAT E16.8 IS TO BE USED WHEN PRINTING A.
C     IERR-    OUTPUT PARAMETER:
C              IERR=0 IF NO TROUBLE.
C              IERR=1 IF SUBROUTINE MPRI IS CALLED INCORRECTLY.
C              A DIAGNOSTIC MESSAGE IS PRINTED INSTEAD OF THE MATRIX.
C
C     NOTE: IT IS POSSIBLE TO USE SUBROUTINE MPRI WHEN A IS
C     DIMENSIONED AS A VECTOR A(*) IN THE CALLING PROGRAM BY PUTTING
C     N=1 (A IS PRINTED AS A COLUMN VECTOR) OR BY PUTTING
C     M=1 AND IA=1 (A IS PRINTED AS A ROW VECTOR).
C
C     TO FIT THE PRINTED MATRIX TO AN A4-PAGE, PUT IND=5 OR IND=-3 (UP-
C     RIGHT PAGE) OR PUT IND=7 OR IND=-5 (LYING PAGE).
C
C     SUBROUTINE REQUIRED
C              NONE
C
      DIMENSION A(IA,1)
C
```

```
      SUBROUTINE NORM(A,N,IA,S)
C
C     THE SUBROUTINE COMPUTES THE MINIMAXNORM OF A WHERE
C     A=NXN-MATRIX
C     AUTHOR,K.MORTENSSON 31/07-67
C
C     A IS MATRIX OF ORDER NXN
C     N IS THE ORDER OF THE MATRIX
C     S IS THE RESULTING NORM
C     IA IS THE DIMENSION PARAMETER
C
C     SUBROUTINE REQUIRED
C             NONE
C
      DIMENSION A(IA,IA)
C
```

```
      SUBROUTINE NUFLET(FUNCT,X,N,F,G,H,W,DFN,XM,HH,EPS,MODE,MAXFN,
     *IPRINT,IEXIT)
C
C     ROUTINE FOR FINDING THE MINIMUM OF A FUNCTION F(X), WHEN
C     ONLY FUNCTION VALUES ARE AVAILABLE
C     AUTHOR, T. GLAD 1973-07-01.
C     REVISED, C. KALLSTROM 1973-08-10.
C     REFERENCE:   R. FLETCHER,  FORTRAN SUBROUTINES FOR MINIMIZATION
C     BY QUASI-NEWTON METHODS,  REPORT AERE-R7125, HARWELL
C
C     FUNCT   A SUBROUTINE XXX(X,N,F,DUM1,DUM2,IDUM,ICONT,IERR),
C             WHICH COMPUTES THE FUNCTION VALUE F AT THE POINT X OF
C             DIMENSION N. FROM NUFLET SUBROUTINE XXX IS ALWAYS
C             CALLED WITH ICONT=0. PUT IERR=-1 IF XXX IS CALLED
C             ILLEGALLY OR IF THE MINIMIZING IS TO BE TERMINATED,
C             OTHERWISE PUT IERR=0.
C     X       A REAL ARRAY OF N ELEMENTS IN WHICH THE CURRENT ESTIMATE
C             OF THE SOLUTION IS STORED. AN INITIAL APPROXIMATION MUST
C             BE SET IN X ON ENTRY TO NUFLET AND THE BEST ESTIMATE
C             OBTAINED WILL BE RETURNED ON EXIT
C     N       THE NUMBER OF VARIABLES (MIN 2, NO MAX)
C     F       A REAL NUMBER IN WHICH THE BEST VALUE OF F(X) CORRESPONDING
C             TO X ABOVE WILL BE RETURNED
C     G       A REAL ARRAY OF N ELEMENTS USED TO STORE AN ESTIMATE OF
C             THE GRADIENT VECTOR. NOT TO BE SET ON ENTRY
C     H       A REAL ARRAY OF N*(N+1)/2 ELEMENTS IN WHICH AN ESTIMATE OF
C             HESSIAN MATRIX IS STORED. THE MATRIX IS REPRESENTED IN
C             THE PRODUCT FORM (LDL)T WHERE L IS A LOWER TRIANGULAR
C             MATRIX WITH UNIT DIAGONALS AND D IS A DIAGONAL MATRIX.
C             THE LOWER TRIANGLE OF L IS STORED BY COLUMNS IN H EXCEPTING
C             THAT THE UNIT DIAGONAL ELEMENTS ARE REPLACED BY THE
C             CORRESPONDING ELEMENTS OF D. THE SETTING OF H ON ENTRY IS
C             CONTROLLED BY THE PARAMETER MODE.
C     W       A REAL ARRAY OF 4*N ELEMENTS USED AS WORKING SPACE
C     DFN     A REAL NUMBER WHICH MUST BE SET SO AS TO GIVE NUFLET AN
C             ESTIMATE OF THE LIKELY REDUCTION TO BE OBTAINED IN F(X).
C             DFN IS USED ONLY ON THE FIRST ITERATION SO AN ORDER OF
C             MAGNITUDE ESTIMATE WILL SUFFICE.
C              DFN>0 THE SETTING OF DFN ITSELF WILL BE TAKEN AS THE
C              LIKELY REDUCTION TO BE OBTAINED IN F(X)
C              DFN=0 IT WILL BE ASSUMED THAT AN ESTIMATE OF THE MINIMUM
C              VALUE OF F(X) HAS BEEN SET IN ARGUMENT F, AND THE LIKELY
C              REDUCTION IN F(X) WILL BE COMPUTED ACCORDING TO THE
C              INITIAL FUNCTION VALUE
C              DFN<0 A MULTIPLE ABS(DFN) OF THE MODULUS OF THE INITIAL
C              FUNCTION VALUE WILL BE TAKEN AS AN ESTIMATE OF THE LIKELY
C              REDUCTION.
C     XM      A REAL ARRAY OF N ELEMENTS USED IN SCALING,SEE BELOW
C     HH      A REAL NUMBER. THE STEP LENGTH USED WHEN CALCULATING G(I)
C             BY DIFFERENCES IS HH*XM(I). HINT GIVEN BY FLETCHER: SET HH
C             EQUAL TO 2**(-T/2) WHERE T IS THE NUMBER OF SIGNIFICANT
C             BINARY DIGITS IN THE CALCULATION OF F.
C     EPS     A REAL NUMBER. THE ACCURACY REQUIRED IN X(I) IS EPS*XM(I).
C     MODE    AN INTEGER WHICH CONTROLS THE SETTING OF THE INITIAL
C             ESTIMATE OF THE HESSIAN MATRIX IN THE PARAMETER H.
C              MODE=1 AN ESTIMATE CORRESPONDING TO THE UNIT MATRIX IS SET
C              IN H BY NUFLET
C              MODE=2 NUFLET ASSUMES THAT THE HESSIAN MATRIX ITSELF HAS
C              BEEN SET IN H BY COLUMNS OF ITS LOWER TRIANGLE, AND THE
C              CONVERSION TO (LDL)T FORM IS CARRIED OUT BY NUFLET. THE
C              HESSIAN MATRIX MUST BE POSITIVE DEFINITE
C              MODE=3 NUFLET ASSUMES THAT THE HESSIAN MATRIX HAS BEEN SET
C              IN H IN PRODUCT FORM.
```

```
C      MAXFN  AN INTEGER SET TO THE MAXIMUM NUMBER OF CALLS OF FUNCT
C             PERMITTED
C      IPRINT AN INTEGER CONTROLLING PRINTING. PRINTING OCCURS EVERY
C             ABS(IPRINT) ITERATIONS AND ALSO ON EXIT, IN THE FORM
C
C               ITERATION NO.
C               NO. OF CALLS OF FUNCT
C               FUNCTION VALUE
C               X(1),X(2),...X(N)      8 TO A LINE
C               G(1),G(2),...G(N)      8 TO A LINE
C               IEXIT=   (ON EXIT ONLY)
C
C             THE VALUES OF X AND G CAN BE SUPPRESSED ON INTERMEDIATE
C             ITERATIONS BY SETTING IPRINT<0. ALL INTERMEDIATE PRINTING
C             CAN BE SUPPRESSED BY SETTING IPRINT=MAXFN+1. ALL PRINTING
C             CAN BE SUPPRESSED BY SETTING IPRINT=0.
C      IEXIT  AN INTEGER GIVING THE REASON FOR EXIT FROM NUFLET.THIS WILL
C             BE SET BY NUFLET AS FOLLOWS
C              IEXIT=0 THE NORMAL EXIT IN WHICH ABS(DX(I))<EPS*XM(I) FOR
C              I=1,2,..N WHERE DX(I) IS THE CHANGE IN X ON AN ITERATION
C              IEXIT=1 (MODE=2 ONLY) HESSIAN MATRIX IS NOT POS DEFINITE.
C              IEXIT=2 GT*DX .GE. 0 EITHER DUE TO ROUNDING ERRORS
C              BECAUSE EPS IS SET TOO SMALL FOR THE COMPUTER WORD LENGTH
C              OR THE TRUNCATION ERROR IN THE FINITE DIFFERENCE FORMULA
C              FOR G BEING DOMINANT.
C              IEXIT=3 FUNCT CALLED MAXFN TIMES.
C              IEXIT=4 MINIMIZING TERMINATED FROM FUNCT.
C
C      SUBROUTINE REQUIRED
C              (FUNCT)
C
      DIMENSION X(1),G(1),H(1),W(1),XM(1)
C
```

```
LOGICAL FUNCTION OVFLOW(IDUM)
LOGICAL OVFLAG
COMMON/OVFCOM/ OVFLAG
DATA NERR/0/
N=NINTRG(3)
OVFLOW=N.GT.NERR .AND. OVFLAG
NERR=N
RETURN
END
```

```
      SUBROUTINE PARFIL(Q,N)
C
C     WRITES PARAMETER VECTOR Q ON LUN 106
C
      DIMENSION Q(1)
C
      WRITE(106,100)(Q(I),I=1,N)
 100  FORMAT(1X,E16.9)
      RETURN
      END
```

```
      SUBROUTINE POWBRE(FUNC,X,N,FMIN,EIG,V,IV,DIST,SCALE,TOL,MODE,
     *MAXFN,NSTOP,ILLCO,IPRIN,IEXIT)
C
C     ROUTINE FOR FINDING THE MINIMUM OF A FUNCTION F(X), WHEN ONLY
C     FUNCTION VALUES ARE AVAILABLE
C
C     AUTHOR T. GLAD  1974-04-25
C     REVISED, C.KALLSTROM 1977-02-18.
C
C     FUNC-A SUBROUTINE XXX(X,N,F,DUM1,DUM2,IDUM,ICON,IERR) WHICH
C      COMPUTES THE FUNCTION VALUE F AT THE POINT X OF DIMENSION N
C      PUT IERR=-1 IF XXX IS CALLED ILLEGIALLY OR IF THE
C      MINIMIZATION IS TO BE TERMINATED, OTHERWISE PUT IERR=0
C     X AN N-VECTOR CONTAINING AN INITIAL GUESS OF THE MINIMUM,
C      RETURNED AS THE MINIMUM POINT
C     N- NUMBER OF VARIABLES (MIN 2,MAX 40)
C     FMIN-RETURNED AS THE VALUE OF F(X) AT THE MINIMUM
C     EIG-VECTOR OF DIMENSION N CONTAINING THE ESTIMATED EIGENVALUES
C      OF THE SECOND DERIVATIVE; TO BE SET ON ENTRY ONLY IF MODE=2
C     V-MATRIX OF DIMENSION N*N CONTAINING AS COLUMN VECTORS THE
C      SEARCH DIRECTIONS USED; ON RETURN THESE ARE APPROXIMATIONS OF
C      THE EIGENVECTORS OF THE SECOND DERIVATIVE; TO BE SET ON ENTRY
C      ONLY IF MODE=2
C     IV-DIMENSION PARAMETER OF V
C     DIST-ESTIMATED DISTANCE FROM INITIAL APPROXIMATION TO MINIMUM;
C      LIMITS THE STEP-SIZE IN THE LINEAR MINIMIZATION
C     SCALE-LIMIT OF RESCALING OF THE AXES; IF SCALE=1. NO RESCALING
C      IS DONE - RECOMMENDED ON WELL-CONDITIONED PROBLEMS; IF THE
C      PROBLEM IS ILL-CONDITIONED SCALE=10. IS RECOMMENDED
C     TOL-TOLERANCE; THE ALGORITHM TRIES TO MAKE THE ERROR IN X LESS
C      THAN (EPSMAC*ABS(X)+T) WHERE EPSMAX IS THE RELATIVE MACHINE
C      PRECISION
C     MODE-IF MODE=1 THE ALGORITHM IS STARTED WITH THE COORDINATE
C      AXES AS SEARCH DIRECTIONS
C      IF MODE=2 THE SEARCH DIRECTIONS ARE GIVEN IN V ON ENTRY
C     MAXFN-MAX. NUMBER OF FUNCTION EVALUATIONS TO BE MADE
C     NSTOP-NUMBER OF ITERATIONS WITHOUT PROGRESS BEFORE TERMINATION
C      (NORMAL VALUE=1)
C     ILLCO-LOGICAL VARIABLE; IF THE PROBLEM IS SUPPOSED TO BE ILL-
C      CONDITIONED PUT ILLCO=.TRUE., OTHERWIZE .FALSE.
C     IPRIN-CONTROLS PRINTOUT
C      IF IPRIN=0 THERE IS NO PRINTOUT
C      IF IPRIN=1 X AND THE FUNCTION VALUE ARE PRINTED
C      IF IPRIN=2 EIGENVALUES AND SCALE FACTORS ARE ALSO PRINTED
C      IF IPRIN=4 EIGENVECTORS ARE ALSO PRINTED
C     IEXIT-IF ON EXIT, IEXIT=0 THE MINIMUM HAS BEEN FOUND
C      IF IEXIT=1 THE ALGORITHM IS TERMINATED ON MAX. NUMBER OF
C      FUNCTION EVALUATIONS
C      IF IEXIT=4 EXECUTION HAS BEEN TERMINATED FROM FUNC
C
C     SUBROUTINE REQUIRED
C        (FUNC)
C        MIN
C           FONED
C              (FUNC)
C        REK
C        QUAD1
C           MIN
C              FONED
C                 (FUNC)
C        MINFIT
C        SORT
C
```

```
      DIMENSION X(1),EIG(1),V(IV,1)
C
      LOGICAL ILLCO,ILLC
      REAL TOL,T,EPSMAC,DIST,X,FMIN,SCALE,MACHEP,H,RD1,
     *DUM1,DUM2,FF
      REAL S,SL,DN,DMIN,FX,F1,LDS,LDT,SF,DF,QF1,QD0,QD1,
     *QA,QB,QC,M2,M4,SMALL,VSMALL,LARGE,VLARGE,SCBD,LDFAC,T2,
     *D(40),Y(40),Z(40),Q0(40),Q1(40),V,EIG,XX(40)
      COMMON/INT/ NL,NF,KL,KT,KTM
      COMMON/SUBS/MACHEP,M2,M4,T,SMALL,DMIN,LDT,H,QA,QB,QC,QD0,QD1,QF1,
     *Q0,Q1
C
      DATA EPSMAC/.74505806E-08/
C
```

```
      REAL FUNCTION RTTFF(IC)
C
C     READS ONE REAL NUMBER FROM STANDARD INPUT IN FREE FORMAT.
C     IC .NE. 0 FORCES READING OF A NEW LINE AND IC IS SET TO 0
C     IC .EQ. 0 - THE PREVIOUSLY READ LINE IS SCANNED AND IF EMPTY
C                 A NEW LINE IS READ
C     IF EOF ENCOUNTERED OF INPUT IC IS SET TO -1
C
C     AUTHOR, T. ESSEBO 1975-02-19
C     REVISED T. ESSEBO 1977-03-06
C
C     SUBROUTINE REQUIRED
C         RBUFF
C         IFAC
C         RIFF
C
      LOGICAL EOF
      DIMENSION RRES(2)
      COMMON/RTFCOM/IP,BUFF(20)
      DATA MAXCH/8/,LUN/5/
C
```

```
      SUBROUTINE SGAIN(TS,EPSI,NKAL,IERROR)
C
C     SUBROUTINE TO BE USED FOR LINEAR SYSTEM PARAMETER IDENTIFICATION
C     TO SAMPLE SYSTEM AND COVARIANCE MATRICES (IF ISYS .GT. 0) AND
C     TO COMPUTE THE DISCRETE, STATIONARY FILTER GAIN AK.
C     AUTHOR, C. KALLSTROM 1974-02-15.
C     REVISED, T. ESSEBO 1974-04-16.
C
C     TS - SAMPLING INTERVAL (NOT USED IF ISYS .LT. 0).
C     EPSI - TEST QUANTITY. THE ITERATIONS ARE TERMINATED WHEN
C            NORM(P(T)-P(T-1))/NORM(P(T)) .LE. EPSI.
C     NKAL - MAXIMUM NUMBER OF ITERATIONS TO BE DONE (NO MAX,MIN 2).
C     IERROR - ERROR PARAMETER.
C              IERROR=0 IF OK.
C              IERROR=1 IF TS IS NONPOSITIVE.
C              IERROR=2 IF NO CONVERGENCE IN TRANS.
C              IERROR=3 IF NO CONVERGENCE WHEN THE KALMAN BUCY
C              FILTERING EQUATIONS ARE ITERATED.
C              IERROR=4  IF OVERFLOW
C
C     NOTE 1: FOLLOWING VARIABLES ARE USED (BUT NOT CHANGED) IN THE
C     SUBROUTINE
C     COMMON BLOCK /SYSPAR/:ISYS, MEAS,NX,NY,NOMAT
C     COMMON BLOCK /DATA/: A,C,R1,R12,R2,P0(ACCORDING TO VECTOR NOMAT)
C     COMMON BLOCK /LISCON/: NMAX,EPST,EPSK
C
C     NOTE 2: FOLLOWING MATRICES OF THE COMMON BLOCK /DATA/ ARE USED
C     AS STORAGE IN THE SUBROUTINE: AD,CD,R1D,R12D,R2D,AKD,P
C
C     NOTE 3: IF IERROR=0 ,THE COMPUTED STATIONARY FILTER GAIN IS STORED
C    IN AK  OF THE COMMON BLOCK /DATA/. IF IERROR=3 ,THE LAST COMPUTED
C     (NOT STATIONARY) FILTER GAIN IS STORED IN AK.
C
C     NOTE 4: ADDITIONAL INFORMATION CAN BE OBTAINED FROM THE COMMON
C     BLOCK /SGA/
C
C     NOTE 5: COMMON BLOCK /TRANS2/ ,WHICH IS SHARED WITH
C     SUBROUTINE TRANS CONTAINS WORK ARRAYS.
C
C     SUBROUTINES REQUIRED
C        TRANS
C           NORM
C        KALMAN
C           NORM
C           DESYM
C           SOLVS
C        NORM
C
      LOGICAL OVFLOW
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
```

```
C
      COMMON/TRANS2/DUM1(1600),S1(400)
C
      COMMON/KAL1/DUMM1(460),AKT(20,20),DUMM2(2)
C
      COMMON/SGA/NNTR(6),NOKAL(6),XKAL(20),YKAL(20),EPSIL,IKC,IK
C
      DEFINE AK(I,J)=V(IXAK+NX*J-NX+I)
C
```

```
      SUBROUTINE SINT(NTIME,NSTOP,INIT,IV,VLOSS,IERR)
C
C     SUBROUTINE TO BE USED FOR LINEAR SYSTEM PARAMETER
C     IDENTIFICATION TOGETHER WITH SUBROUTINE LISPID.
C     THE SUBROUTINE CAN PERFORM ONE OR MORE OF FOLLOWING TASKS:
C     1.        TRANSFORMATION OF CONTINUOUS SYSTEM AND COVARIANCE
C           MATRICES TO DISCRETE MATRICES
C     2.        ITERATION OF THE DISCRETE
C           SYSTEM EQUATIONS
C     3.        ITERATION OF THE DISCRETE KALMAN BUCY FILTERING
C           EQUATIONS
C     4.        PREDICTION
C     5.        COMPUTATION OF LOSS FUNCTION
C
C     AUTHOR, C. KALLSTROM 1973-06-26.
C     REVISED, C.KALLSTROM 1977-02-18.
C
C     NTIME-  COUNTER FOR MEASUREMENT EVENTS (MAX NP, MIN 1)
C             RETURNED CONTAINING THE VALUE OF NSTOP (IF IERR=0).
C     NSTOP-  (NO MAX, MIN NTIME)
C             THE EQUATIONS ARE ITERATED FROM NTIME TO MIN(NSTOP,NP).
C             NSTOP IS NOT CHANGED IN SINT
C     INIT-   PUT INIT=1 AT THE FIRST CALL OF SINT TO INITIALIZE,
C             THEN PUT INIT=0.
C     IV-     CONTROLS THE REJECTING OF MEASUREMENTS AND THE UPDATING OF
C             LOSS FUNCTIONS (MAX 4, MIN 0).
C             PUT IV=0 IF NO REJECTING OF MEASUREMENTS.
C             PUT IV=1 TO REJECT THE MEASUREMENTS NTIME TO STOP.
C             PUT IV=2 TO REJECT MEASUREMENTS REACHED BY PREDICTION
C             FROM NTIME TO NSTOP.
C             PUT IV=3 IF BOTH IV=1 AND IV=2.
C             PUT IV=4 TO COMPUTE THE FINAL LOSS,
C             RETURNED IN VLOSS,ONLY THE ARGUMENTS VLOSS AND IERR ARE
C             USED.
C     VLOSS-  RETURNED LOSS WHEN IV=4.
C             IF ISYS=1, -1, 2, -2,   VLOSS=(DET(SUM(EPS*EPST)))/N,
C             WHERE N IS THE NUMBER OF CONTRIBUTIONS.
C             IF ISYS=3, -3, 4, -4,   VLOSS=-LOG(L(TH,R))/N.
C     IERR- ERROR PARAMETER.
C             IERR=-1 IF THE COVARIANCE MATRIX IS SINGULAR WHEN
C                     COMPUTING VLOSS. ONLY USED WHEN IV=4 AND ISYS=
C                     1,-1,2 OR -2. VLOSS CONTAINS A ROUGH ESTIMATE
C                     OF THE REAL LOSS.
C             IERR=0 IF OK.
C             IERR=1 IF IV HAS AN ILLEGAL VALUE.
C             IERR=2 IF NTIME AND/OR NSTOP HAVE ILLEGAL VALUES.
C             IERR=3 IF A SAMPLING INTERVAL IS NONPOSITIVE.
C             IERR=4 IF NO CONVERGENCE IN TRANS
C                    (GIVE NMAX A LARGER VALUE OR INCREASE EPST).
C             IERR=5 ITERATION IS TERMINATED DUE TO REFERENCE OF
C                    MEASUREMENTS BEYOND INDEX NP
C         IERR=6  OVERFLOW IN COMPUTATIONS
C
C    NOTE 1: THE SYSTEM MATRICES A,B,..,R2,AK IN /DATA/ ARE
C    NEVER CHANGED BUT THEY ARE TRANSFERRED TO MATRICES AD,BD,..
C    ,AKD ACCORDING TO PARAMETER ISAMP ,AND IT IS THESE
C    MATRICES THAT ARE USED FOR THE ITERATIONS AND PREDICTION.
C    THE INITIAL STATES X0 AND P0 ARE ONLY TRANSFERED TO X AND P
C    WHEN INIT=1.
C
C    NOTE 2: THE  COMMON BLOCKS ARE DESCRIBED IN SUBROUTINE
C    LISPID (THE COMMON BLOCK /KAL1/ IS DESCRIBED IN KALMAN).
C
```

```
C
C     SUBROUTINES REQUIRED
C           SAMP
C              COSA
C                 NORM
C                 EXPAN
C              TRANS
C                 NORM
C           LISY
C           KALMAN
C              NORM
C              DESYM
C              SOLVS
C           LOS
C              NORM
C              DESYM
C
C
      LOGICAL OVFLOW
      DIMENSION U(20),Y(20),XPD(20),UPD(20),YPD(20),INTV(1)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/LISP/VLOS1,NVLOS1,VLOS2,NVLOS2,VLOSX,VLOSDT,G(40),
     *STDEV(40),IABSIS,IM2,ICOS,ML,NOM(6),NOTRAN(6),NRR,NVLDT1
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
      COMMON/KAL1/ YMODD(20),E(20),DUM1(20),R(20,20),AKT(20,20),
     *TEST,DELTAV
C
      EQUIVALENCE (INTV,V)
C
      DEFINE IEPS(I)=INTV(IXIEPS+I)
      DEFINE TIM(I)=V(IXTIM+I)
      DEFINE UIN(I,J)=V(NP*J-NP+I)
      DEFINE YMS(I,J)=V(IXYMS+NP*J-NP+I)
      DEFINE EPS(I,J)=V(IXEPS+NP*J-NP+I)
      DEFINE X(I)=V(IXX+I)
      DEFINE XO(I)=V(IXXO+I)
      DEFINE RR(I,J)=V(IXRR+NY*J-NY+I)
      DEFINE RPD(I,J)=V(IXRPD+NY*J-NY+I)
C
```

```
      SUBROUTINE SOLVB(B,X,NN,NNB,IA)
C
C     SOLVES AX=B  USING UL AND IPS IN COMMON FROM DECOM.
C     REFERENCE, FORSYTHE MOLER.
C     AUTHOR, PER HAGANDER 1968-09-05.
C     REVISED, CLAES KALLSTROM 1971-03-20.
C
C     B-MATRIX OF ORDER NNXNNB, CONTAINING R-H-S VECTORS, NOT DESTROYED.
C     X- MATRIX OF ORDER NNXNNB,RETURNED CONTAINING THE SOLUTIÕN VECTORS
C     NN- NUMBER OF EQUATIONS (MAX 30, MIN 1).
C     NNB- NUMBER OF RIGHT HAND SIDE VECTORS (NO MAX, MIN 1).
C     IA- DIMENSION PARAMETER.
C     NOTE, IF NNB=1, B AND X CAN BE DIMENSIONED AND TREATED AS VECTORS
C     IN THE CALLING PROGRAM.
C
C     SUBROUTINE REQUIRED
C            NONE
C
      DIMENSION B(IA,1),X(IA,1)
C
      COMMON/IPUL30/IPS(30),UL(30,30)
C
```

```
      SUBROUTINE SOLVS(G,B,X,NN,NNB,IA)
C
C     SOLVES AX=B, WHERE A IS A SYMMETRIC , POSITIVE DEFINITE MATRIX,
C     DECOMPOSED IN THE FORM A=G*G(TRANSPOSED) IN SUBROUTINE DESYM.
C     REFERENCE, FORSYTHE MOLER P 114.
C     AUTHOR,C.KALLSTROM 1970-08-26.
C     REVISED, CLAES KALLSTROM 1971-03-20.
C
C     G-  LOWER TRIANGULAR MATRIX OF ORDER NN*NN, NOT DESTROYED.
C         (THE UPPER TRIANGULAR PART IS NOT USED).
C     B-  MATRIX OF ORDER NN*NNB, CONTAINING RIGHT HAND SIDE VECTORS,
C         NOT DESTROYED.
C     X-  MATRIX OF ORDER NN*NNB, RETURNED CONTAINING THE SOLUTION
C         VECTORS.
C     NN- NUMBER OF EQUATIONS (NO MAX, MIN 1).
C     NNB-NUMBER OF RIGHT HAND SIDE VECTORS (NO MAX, MIN 1).
C     IA- DIMENSION PARAMETER (NO MAX).
C     NOTE. IF NNB=1, B AND X CAN BE DIMENSIONED AND TREATED AS VECTORS
C     IN THE CALLING PROGRAM.
C
C     SUBROUTINE REQUIRED
C           NONE
C
      DIMENSION G(IA,IA),B(IA,1),X(IA,1)
C
```

```
      SUBROUTINE SYMIN(N,IA,IFAIL,A,EPS)
C
C     SUBROUTINE FOR INVERSION OF SYMMETRIC MATRICES.
C     REFERENCE,RUTISHAUSER,CACM,ALG.NR.150.
C     AUTHOR,K.MORTENSSON 04/04-68.
C     REVISED, T. ESSEBO 1973-12-05.
C
C     A-MATRIX TO BE INVERTED.UPON RETURN A CONTAINS A-1 IF THE
C     INVERSION HAS SUCCEEDED.
C     N-ORDER OF A.
C     IFAIL-RETURNED 0 IF THE SUBROUTINE HAS EXECUTED CORRECTLY,
C     1 IF NOT.
C     IA-DIMENSION PARAMETER.
C     EPS= TEST QUANTITY (SUITABLE VALUE: 1.E-7)
C     CAUTION.NEAR-SINGULAR MATRICES MAY GIVE MISLEADING RESULTS.
C     MAXIMUM ORDER OF A=40.
C
C     SUBROUTINE REQUIRED
C             NONE
C
      DIMENSION A(IA,IA),P(40),Q(40),IR(40)
C
```

```
      SUBROUTINE TRANS(A,B,C,D,QR1,QR12,QR2,NX,NU,NY,NOMAT,IPROB,TSAMP,
     *   EPS,NMAX,NOCONV,IA,IC)
C
C     TRANSFORMS A CONTINUOUS CONTROL PROBLEM TO A DISCRETE CONTROL
C     PROBLEM OR A CONTINUOUS ESTIMATION PROBLEM TO A DISCRETE
C     ESTIMATION PROBLEM.
C     REFERENCE, ASTROM: ON THE CHOICE OF SAMPLING RATES IN OPTIMAL
C             LINEAR SYSTEMS, IBM RESEARCH REPORT RJ-243, 1963.
C             MORTENSSON: LINEAR QUADRATIC CONTROL PACKAGE, PART II -
C             THE DISCRETE PROBLEM, REPORT 6904.
C             ASTROM: INTRODUCTION TO STOCHASTIC CONTROL THEORY,
C             ACADEMIC PRESS, 1970.
C     AUTHOR, C.KALLSTROM 1972-12-06.
C
C     A-      DYNAMICS MATRIX OF ORDER NX*NX.
C     B-      INPUT MATRIX OF ORDER NX*NU. NOT REFERENCED IN TRANS
C             IF NOMAT(1)=0.
C     C-      OUTPUT MATRIX OF ORDER NY*NX. NOT REFERENCED IN TRANS
C             IF IPROB=1 OR NOMAT(2)=0.
C     D-      MATRIX OF ORDER NY*NU. NOT REFERENCED IN TRANS IF IPROB=1.
C             CAN BE USED AT RETURN EVEN IF NOMAT(3)=0 WHEN IPROB=2.
C     QR1-    CRITERIA OR COVARIANCE MATRIX OF ORDER NX*NX.
C             NOT REFERENCED IN TRANS IF NOMAT(4)=0.
C     QR12-   CRITERIA MATRIX OF ORDER NX*NU OR COVARIANCE MATRIX OF
C             ORDER NX*NY. CAN BE USED AT RETURN EVEN IF NOMAT(5)=0.
C     QR2-    CRITERIA MATRIX OF ORDER NU*NU OR COVARIANCE MATRIX OF
C             ORDER NY*NY. CAN BE USED AT RETURN EVEN IF NOMAT(6)=0.
C     NX-     NUMBER OF STATES (MAX 20, MIN 1).
C     NU-     NUMBER OF CONTROL VARIABLES (MAX 20, MIN 1).
C             NOT REFERENCED IN TRANS IF IPROB=1, NOMAT(1)=0, NOMAT(5)=0
C             AND NOMAT(6)=0 OR IF IPROB=2, NOMAT(1)=0 AND NOMAT(3)=0.
C     NY-     NUMBER OF MEASUREMENTS (MAX 20, MIN 1).
C             NOT REFERNECED IN TRANS IF IPROB=1 OR IF IPROB=2,
C             NOMAT(2)=0, NOMAT(3)=0, NOMAT(5)=0 AND NOMAT(6)=0.
C     NOMAT-  INPUT VECTOR OF DIMENSION 6.
C             PUT NOMAT(1)=0 IF NO B, ELSE PUT NOMAT(1)=1.
C             PUT NOMAT(2)=0 IF NO C, ELSE PUT NOMAT(2)=1.
C             PUT NOMAT(3)=0 IF NO D, ELSE PUT NOMAT(3)=1.
C             PUT NOMAT(4)=0 IF NO QR1, ELSE PUT NOMAT(4)=1.
C             PUT NOMAT(5)=0 IF NO QR12, ELSE PUT NOMAT(5)=1.
C             PUT NOMAT(6)=0 IF NO QR2, ELSE PUT NOMAT(6)=1.
C             NOMAT(2) AND NOMAT(3) ARE NOT USED IF IPROB=1.
C     IPROB-  PUT IPROB=1 IF CONTROL PROBLEM.
C             PUT IPROB=2 IF ESTIMATION PROBLEM.
C     TSAMP-  SAMPLING INTERVAL.
C     EPS-    TEST QUANTITY. THE SERIES EXPANSIONS ARE TRUNCATED
C             WHEN THE ESTIMATED RELATIVE TRUNCATION ERRORS ARE LESS
C             THAN EPS.
C     NMAX-   THE SERIES EXPANSIONS ARE TRUNCATED AFTER AT LEAST
C             NMAX TERMS.
C     NOCONV- RETURNED CONTAINING 1 IF THE SERIES EXPANSIONS HAVE NOT
C             CONVERGED AFTER NMAX TERMS, 2 IF OVERFLOW IN COMPUTATIONS
C             ,ELSE RETURNED 0
C     IA-     DIMENSION PARAMETER OF A,B,QR1 AND QR12.
C     IC-     DIMENSION PARAMETER OF C,D AND QR2.
C
C     NOTE 1: THE MATRICES OF THE CALL ARE RETURNED CONTAINING
C     THE APPROPRIATE DISCRETE MATRICES.
C
C     NOTE 2: IT IS NOT NECESSARY TO DIMENSION A MATRIX, WHICH IS
C     NOT REFERENCED IN TRANS, IN THE CALLING PROGRAM.
C
C     NOTE 3: THE SYMMETRIC MATRICES OF THE CALL (QR1 AND QR2) MUST BE
```

```
C      COMPLETE. QR1 AND QR2 ARE SYMMETRIZED BEFORE RETURN.
C
C      NOTE 4: THE COMMON BLOCK /TRANS1/ CONTAINS THE USED NUMBER OF
C      TERMS NTERM (AT MOST EQUAL TO NMAX) AND THE MAXIMUM ESTIMATED
C      RELATIVE TRUNCATION ERROR ERR (TESTED AGAINST EPS) AFTER NTERM
C      TERMS.
C
C      NOTE 5: THE COMMON BLOCK /TRANS2/ CONTAINS WORK ARRAYS.
C
C      SUBROUTINE REQUIRED
C              NORM
C
       LOGICAL OVFLOW,LDUM
C
       DIMENSION A(IA,1),B(IA,1),C(IC,1),D(IC,1),QR1(IA,1),QR12(IA,1),
      *   QR2(IC,1),NOMAT(1)
C
       COMMON/TRANS1/ NTERM,ERR
C
       COMMON/TRANS2/ T(20,20),Y(20,20),S1(20,20),S2(20,20),S3(20,20)
C
```

APPENDIX D - STANDARD USER - DESIGNED PROGRAMS

Listings of the following standard user-designed programs are given:

MAIN
RDATA
SYST

Notice that MAIN and RDATA are complete programs, while it is necessary to incorporate a mathematical model defined by the user before the model subroutine SYST can be compiled and linked together with the program package LISPID. The FORTRAN elements MAIN, RDATA and SYST and the compiled elements of MAIN and RDATA are stored on the LISPID file.

```
C       PROGRAM MAIN
C
C       STANDARD MAIN PROGRAM FOR LISPID.
C
C       AUTHOR, C. KALLSTROM 1978-06-29.
C
C       SUBROUTINE REQUIRED
C          IOLISP
C             MPRI
C             RTTFF (WITH SUBROUTINES)
C             LISDAT
C          DATEXP
C             MCORE
C          RDATA
C             RTTFF (WITH SUBROUTINES)
C          LISPID (WITH SUBROUTINES)
C          GRDCHK (WITH SUBROUTINES)
C
        DIMENSION TH(40),NPR(3)
C
        COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
       *NX,NU,NY,NTH,NOMAT(6)
C
        COMMON/DATA/V(1)
C
        COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
       *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXPO,IXXO,IXAD,IXBD,IXCD,
       *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
       *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
        COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
       *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
       *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
       *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
        COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
       *IERRSY
C
        EXTERNAL SYST
C
        LP=6
C
C       PRINT HEADING.
C
        WRITE(LP,100)
    100 FORMAT(7H1LISPID/1X,6(1H*)/)
C
C       READ INPUT PARAMETERS FOR LISPID.
C
        CALL IOLISP(TH,1,0)
C
C       CHANGE STANDARD VALUES OF SOME INPUT PARAMETERS.
C
        MAXFN=500*NTH
        IDH=1
        DO 10 I=1,NTH
     10 DH(I)=HZ
C
C       ORGANIZE /DATA/ AND /INDEX/.
C
        CALL DATEXP
C
C       READ DATA.
```

```
C
      CALL RDATA
C
C     SIMULATE THE SYSTEM IF LOOP .NE. -1.
C
      IF(LOOP)18,12,12
   12 DO 14 I=1,3
      NPR(I)=NPRI(I)
   14 NPRI(I)=0
      IACCO=IACC
      IACC=0
      NPLO=NPLOTC
      NPLOTC=0
      NPL=NPLOT
      NPLOT=0
      LOP=LOOP
      LOOP=0
C
      CALL LISPID(SYST,TH,IERR)
C
      DO 16 I=1,3
   16 NPRI(I)=NPR(I)
      IACC=IACCO
      NPLOTC=NPLO
      NPLOT=NPL
      LOOP=LOP
C
C     PRINT INPUT PARAMETERS FOR LISPID.
C
   18 CALL IOLISP(TH,0,2)
C
C     IF LOOP .NE. -1 AND IF IERR .NE. 0 OR IERRSY .NE. 0 ,PRINT
C     WARNING AND PREPARE SIMULATION ONLY (NO PLOTTING).
C
      IF(LOOP)31,28,28
   28 IF(IERR)30,29,30
   29 IF(IERRSY)30,31,30
C
   30 WRITE(LP,102)IERR,IERRSY
  102 FORMAT(/36H SIMULATION ONLY, NO PLOTTING  IERR=,I2,9H  IERRSY=,
     *I2/)
C
      LOOP=0
      NPLOT=0
      IACC=0
      NPLOTC=0
      ICHK=0
C
C     PERFORM IDENTIFICATION OR SIMULATION.
C
   31 CALL LISPID(SYST,TH,IERR)
C
      IF(LOOP)99,32,32
C
C     PRINT FINAL VALUES.
C
   32 WRITE(LP,104)
  104 FORMAT(1H1)
C
      CALL IOLISP(TH,0,2)
C
      IF(ICHK.EQ.0)GO TO 99
C
      CALL GRDCHK(SYST,TH,NTH,XM,HZ)
```

```
C
C
   99 STOP
      END
```

```
      SUBROUTINE RDATA
C
C     STANDARD SUBROUTINE TO READ DATA FOR LISPID.
C
C     AUTHOR, C. KALLSTROM 1978-06-29.
C
C     SUBROUTINE REQUIRED
C        RTTFF (WITH SUBROUTINES)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXPO,IXXO,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      DEFINE UIN(I,J) = V(NP*J-NP+I)
      DEFINE YMS(I,J) = V(IXYMS +NP*J-NP+I)
      DEFINE TIM(I) = V(IXTIM+I)
C
      IC=1
      DO 50 I=1,NP
      IF(IT .EQ. 0) GO TO 10
      TIM(I)=RTTFF(IC)
10    IF(NU .EQ. 0) GO TO 30
      DO 20 J=1,NU
20    UIN(I,J)=RTTFF(IC)
30    DO 40 J=1,NY
40    YMS(I,J)=RTTFF(IC)
50    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE SYST(TH,NTH1,VLOSS,DUM1,DUM2,IDUM1,IDUM2,IER)
C
C     SUBROUTINE TO SUPPLY A MODEL AND COMPUTE LOSS FUNCTION.
C
C     AUTHOR:
C
C     TH-    PARAMETER VECTOR OF DIMENSION NTH1.
C     NTH1-  NUMBER OF PARAMETERS (MAX 40,MIN 2).
C            NTH1 IS EQUAL TO NTH IN /SYSPAR/ .
C     VLOSS-COMPUTED LOSS FUNCTION VALUE.
C     DUM1,DUM2,IDUM1,IDUM2 - DUMMY ARGUMENTS.
C     IER-   ERROR PARAMETER:
C            IER=0 IF OK.
C            IER=-1 IF ERROR IN SYST
C
C     COMMON BLOCK /COMSY/ (ALL VARIABLES EXCEPT IERRSY ARE GIVEN
C            VALUES IN IOLISP) :
C     PAR-   PARAMETER VECTOR OF DIMENSION NPAR. VECTOR TH(I)/SCAL(I)
C            IS A PART OF VECTOR PAR.
C     NPAR-  NUMBER OF PARAMETERS (MAX 40,MIN 2). NTH MUST BE .LE. NPAR.
C     ITH-   INTEGER VECTOR OF DIMENSION NTH CONTAINING THE INDICES OF
C            THE PARAMETERS OF VECTOR PAR, FOR WHICH THE MINIMIZATION
C            IS PERFORMED.
C     SCAL-  SCALING VECTOR OF DIMENSION NTH.
C            SCAL(I) MUST NOT BE EQUAL TO 0.
C     TS-    SAMPLING INTERVAL, ONLY USED IN SUBROUTINE SGAIN.
C     EPSI-  TEST QUANTITY, ONLY USED IN SUBROUTINE SGAIN.
C     NKAL-  MAXIMUM NUMBER OF ITERATIONS, ONLY USED IN
C            SUBROUTINE SGAIN (NO MAX,MIN 2).
C     VOLD-  PUT VOLD INITIALLY EQUAL TO A LARGE POSITIVE VALUE (E.G.
C            1.E+10). WHEN SUBROUTINE SYST IS CALLED ,THE LOSS FUNCTION
C            VALUE VLOSS IS STORED IN VOLD, IF THE COMPUTATION IS SUCCES-
C            FULL , ELSE VLOSS IS PUT EQUAL TO 10.*VOLD AND THE VALUE IS
C            THEN RESTORED IN VOLD.
C     IERRSY-ERROR PARAMETER:
C            IERRSY=0 IF OK.
C            IERRSY=1 IF THE COMPUTATION OF VLOSS HAS FAILED.
C
C     NOTE : IF ISYS=2 OR -2 THE SUBROUTINE SGAIN IS CALLED
C     TO COMPUTE THE DISCRETE STATIONARY FILTER GAIN AK.
C
C     SUBROUTINE REQUIRED
C        SGAIN
C           TRANS
C              NORM
C           KALMAN
C              NORM
C              DESYM
C              SOLVS
C           NORM
C        SINT (WITH SUBROUTINES)
C        OVFLOW
C
      LOGICAL OVFLOW
C
      DIMENSION TH(1)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
```

```
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
      COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
     *IERRSY
C
      DEFINE A(I,J)=V(IXA+NX*J-NX+I)
      DEFINE B(I,J)=V(IXB+NX*J-NX+I)
      DEFINE C(I,J)=V(IXC+NY*J-NY+I)
      DEFINE D(I,J)=V(IXD+NY*J-NY+I)
      DEFINE R1(I,J)=V(IXR1+NX*J-NX+I)
      DEFINE R12(I,J)=V(IXR12+NX*J-NX+I)
      DEFINE R2(I,J)=V(IXR2+NY*J-NY+I)
      DEFINE X0(I)=V(IXX0+I)
      DEFINE P0(I,J)=V(IXP0+NX*J-NX+I)
      DEFINE AK(I,J)=V(IXAK+NX*J-NX+I)
      DEFINE UIN(I,J)=V(NP*J-NP+I)
      DEFINE YMS(I,J)=V(IXYMS+NP*J-NP+I)
      DEFINE TIM(I)=V(IXTIM+I)
C
C     INITIALIZING.
C
      IER=0
      IERRSY=0
      LP=6
      VMAX=1.E20
C
C     UPDATE THE PARAMETER VECTOR PAR USING TH.
C
      DO 10 I=1,NTH
      I1=ITH(I)
   10 PAR(I1)=TH(I)/SCAL(I)
C
C     MATHEMATICAL MODEL.
C
C
C     COMPUTATION OF DISCRETE FILTER GAIN AK IF ISYS=2 OR -2.
C     A CALL TO SGAIN MUST ONLY BE PERFORMED IF NO VALUES
C     ARE ASSIGNED TO AK EARLIER, WHEN ISYS=2 OR -2.
C
  200 IF(IABS(ISYS) .NE. 2) GO TO 300
C
      CALL SGAIN(TS,EPSI,NKAL,IERROR)
C
      IF(IERROR .EQ. 0) GO TO 300
C
      IERRSY=1
      WRITE(LP,1000) IERROR
 1000 FORMAT(15H SGAIN: IERROR=,I2)
      GO TO 990
C
C     ITERATE THE SYSTEM EQUATIONS.
C
  300 NTIME=1
      NSTOP=NP-NPRED2
      IF(NPRED2.EQ.1)NSTOP=NSTOP+1
C
      CALL SINT(NTIME,NSTOP,1,0,DUMMY,IERR)
```

```
C
      IF(IERR .EQ. 0) GO TO 302
      IERRSY=1
      WRITE(LP,1002)IERR,NTIME
 1002 FORMAT(12H SINT: IERR=,I2,8H  NTIME=,I6)
      GO TO 990
C
C     COMPUTE THE LOSS.
C
  302 CALL SINT(NTIME,NSTOP,0,4,VLOSS,IERR)
C
      IF(IERR .EQ. 0) GO TO 980
      IERRSY=1
      WRITE(LP,1004)IERR
 1004 FORMAT(12H SINT: IERR=,I2)
      GO TO 990
C
C     CHECK IF OVERFLOW.
C
  980 IF(.NOT.OVFLOW(IDUM)) GO TO 999
C
      WRITE(LP,1006)
 1006 FORMAT(15H SYST: OVERFLOW)
C
C     IF FAILURE, PUT VLOSS=MIN( 10.*VOLD , VMAX).
C
  990 VLOSS=AMIN1(10.*VOLD,VMAX)
C
C     STORE VLOSS IN VOLD.
C
  999 VOLD=VLOSS
C
      RETURN
      END
```

APPENDIX E - MAP COMMANDS

The element LISMAP of the LISPID file contains suitable MAP commands for generation of an absolute element. A listing of LISMAP is shown on the next page. Notice that the command

```
IN XXX.SYST
```

must be changed to a proper reference to the compiled element of the user-designed subroutine SYST before LISMAP can be used. If the standard programs MAIN and RDATA of the LISPID file are not used, then further changes are required. See the test examples of Appendix F.

```
LIB LIS*LISPID.
LIB LDC*OLDCALCOMP.
NOT TPF$.
 .
SEG MAIN
IN LIS*LISPID.MAIN,.OVFLOW
 . ****************************************************
SEG IOSEG*,(MAIN)
IN LIS*LISPID.IOLISP,.DATEXP,.EIGS,.SYMIN,.SYSPRI,.MPRI,.RDATA
 . --------------------------------------------------
SEG SYSSEG*,(MAIN)
IN XXX.SYST
 . --------------------------------------------------
SEG PLOCOM*,(MAIN)
IN LDC*LIB.SEQ
 . ****************************************************
SEG PLTSEG*,(PLOCOM)
IN LIS*LISPID.LIPLOT
 . --------------------------------------------------
SEG RESSEG*,(PLOCOM)
IN LIS*LISPID.RESLIS
 . ****************************************************
SEG POWSEG*,(SYSSEG)
IN LIS*LISPID.POWBRE
 . --------------------------------------------------
SEG NUFSEG*,(SYSSEG)
IN LIS*LISPID.NUFLET
 . --------------------------------------------------
SEG GRDSEG*,(SYSSEG)
IN LIS*LISPID.GRASD,.GRDCHK
 . ****************************************************
SEG DATSEG*,( )
IN DATA
 . --------------------------------------------------
END . ZZZ
```

APPENDIX F - TEST EXAMPLES

Listings of the 4 test examples described in Section 6 are given in this appendix:

Example 1:

- SYST
- LMAP1
- DAT1

Example 2:

- MAIN2
- RDATA2
- SYST2
- LMAP2
- DAT2

Example 3:

- MAIN3
- RDATA3
- SYST3
- LMAP3
- DAT3

Example 4:

- MAIN4
- RDATA4
- SYST4
- LMAP4
- DAT4

```
      SUBROUTINE SYST(TH,NTH1,VLOSS,DUM1,DUM2,IDUM1,IDUM2,IER)
C
C     SUBROUTINE TO SUPPLY A MODEL AND COMPUTE LOSS FUNCTION.
C     SIMPLE MODEL OF SHIP STEERING.
C
C     AUTHOR: C. KALLSTROM 1978-06-29.
C
C     TH-    PARAMETER VECTOR OF DIMENSION NTH1.
C     NTH1-  NUMBER OF PARAMETERS (MAX 40,MIN 2).
C            NTH1 IS EQUAL TO NTH IN /SYSPAR/ .
C     VLOSS-COMPUTED LOSS FUNCTION VALUE.
C     DUM1,DUM2,IDUM1,IDUM2 - DUMMY ARGUMENTS.
C     IER-   ERROR PARAMETER:
C            IER=0 IF OK.
C            IER=-1 IF ERROR IN SYST
C
C     COMMON BLOCK /COMSY/ (ALL VARIABLES EXCEPT IERRSY ARE GIVEN
C            VALUES IN IOLISP) :
C     PAR-   PARAMETER VECTOR OF DIMENSION NPAR. VECTOR TH(I)/SCAL(I)
C            IS A PART OF VECTOR PAR.
C     NPAR-  NUMBER OF PARAMETERS (MAX 40,MIN 2). NTH MUST BE .LE. NPAR.
C     ITH-   INTEGER VECTOR OF DIMENSION NTH CONTAINING THE INDICES OF
C            THE PARAMETERS OF VECTOR PAR, FOR WHICH THE MINIMIZATION
C            IS PERFORMED.
C     SCAL-  SCALING VECTOR OF DIMENSION NTH.
C            SCAL(I) MUST NOT BE EQUAL TO 0.
C     TS-    SAMPLING INTERVAL, ONLY USED IN SUBROUTINE SGAIN.
C     EPSI-  TEST QUANTITY, ONLY USED IN SUBROUTINE SGAIN.
C     NKAL-  MAXIMUM NUMBER OF ITERATIONS, ONLY USED IN
C            SUBROUTINE SGAIN (NO MAX,MIN 2).
C     VOLD-  PUT VOLD INITIALLY EQUAL TO A LARGE POSITIVE VALUE (E.G.
C            1.E+10). WHEN SUBROUTINE SYST IS CALLED ,THE LOSS FUNCTION
C            VALUE VLOSS IS STORED IN VOLD, IF THE COMPUTATION IS SUCCES-
C            FULL , ELSE VLOSS IS PUT EQUAL TO 10.*VOLD AND THE VALUE IS
C            THEN RESTORED IN VOLD.
C     IERRSY=ERROR PARAMETER:
C            IERRSY=0 IF OK.
C            IERRSY=1 IF THE COMPUTATION OF VLOSS HAS FAILED.
C
C     NOTE : IF ISYS=2 OR -2 THE SUBROUTINE SGAIN IS CALLED
C     TO COMPUTE THE DISCRETE STATIONARY FILTER GAIN AK.
C
C     SUBROUTINE REQUIRED
C        SGAIN
C           TRANS
C              NORM
C           KALMAN
C              NORM
C              DESYM
C              SOLVS
C           NORM
C        SINT (WITH SUBROUTINES)
C        OVFLOW
C
      LOGICAL OVFLOW
C
      DIMENSION TH(1)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
```

```
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXPO,IXXO,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
      COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
     *IERRSY
C
      DEFINE A(I,J)=V(IXA+NX*J-NX+I)
      DEFINE B(I,J)=V(IXB+NX*J-NX+I)
      DEFINE C(I,J)=V(IXC+NY*J-NY+I)
      DEFINE D(I,J)=V(IXD+NY*J-NY+I)
      DEFINE R1(I,J)=V(IXR1+NX*J-NX+I)
      DEFINE R12(I,J)=V(IXR12+NX*J-NX+I)
      DEFINE R2(I,J)=V(IXR2+NY*J-NY+I)
      DEFINE XO(I)=V(IXXO+I)
      DEFINE PO(I,J)=V(IXPO+NX*J-NX+I)
      DEFINE AK(I,J)=V(IXAK+NX*J-NX+I)
      DEFINE UIN(I,J)=V(NP*J-NP+I)
      DEFINE YMS(I,J)=V(IXYMS+NP*J-NP+I)
      DEFINE TIM(I)=V(IXTIM+I)
C
C     INITIALIZING.
C
      IER=0
      IERRSY=0
      LP=6
      VMAX=1.E20
C
C     UPDATE THE PARAMETER VECTOR PAR USING TH.
C
      DO 10 I=1,NTH
      I1=ITH(I)
   10 PAR(I1)=TH(I)/SCAL(I)
C
C     SIMPLE SHIP STEERING MODEL.
C
      A(1,1)=-1./PAR(2)
      A(1,2)=0.
      A(2,1)=1.
      A(2,2)=0.
      B(1,1)=PAR(1)/PAR(2)
      B(2,1)=0.
      C(1,1)=0.
      C(1,2)=1.
      R1(1,1)=ABS(PAR(3))
      R1(1,2)=0.
      R1(2,1)=0.
      R1(2,2)=0.
      R2(1,1)=0.01
      PO(1,1)=1.
      PO(1,2)=0.
      PO(2,1)=0.
      PO(2,2)=1.
      XO(1)=0.
      XO(2)=147.15
C
C     COMPUTATION OF DISCRETE FILTER GAIN AK IF ISYS=2 OR -2.
C     A CALL TO SGAIN MUST ONLY BE PERFORMED IF NO VALUES
```

```
C       ARE ASSIGNED TO AK EARLIER, WHEN ISYS=2 OR -2.
C
  200 IF(IABS(ISYS) .NE. 2) GO TO 300
C
      CALL SGAIN(TS,EPSI,NKAL,IERROR)
C
      IF(IERROR .EQ. 0) GO TO 300
C
      IERRSY=1
      WRITE(LP,1000) IERROR
 1000 FORMAT(15H SGAIN: IERROR=,I2)
      GO TO 990
C
C     ITERATE THE SYSTEM EQUATIONS.
C
  300 NTIME=1
      NSTOP=NP-NPRED2
      IF(NPRED2.EQ.1)NSTOP=NSTOP+1
C
      CALL SINT(NTIME,NSTOP,1,0,DUMMY,IERR)
C
      IF(IERR .EQ. 0) GO TO 302
      IERRSY=1
      WRITE(LP,1002)IERR,NTIME
 1002 FORMAT(12H SINT: IERR=,I2,8H  NTIME=,I6)
      GO TO 990
C
C     COMPUTE THE LOSS.
C
  302 CALL SINT(NTIME,NSTOP,0,4,VLOSS,IERR)
C
      IF(IERR .EQ. 0) GO TO 980
      IERRSY=1
      WRITE(LP,1004)IERR
 1004 FORMAT(12H SINT: IERR=,I2)
      GO TO 990
C
C     CHECK IF OVERFLOW.
C
  980 IF(.NOT.OVFLOW(IDUM)) GO TO 999
C
      WRITE(LP,1006)
 1006 FORMAT(15H SYST: OVERFLOW)
C
C     IF FAILURE, PUT VLOSS=MIN( 10.*VOLD , VMAX).
C
  990 VLOSS=AMIN1(10.*VOLD,VMAX)
C
C     STORE VLOSS IN VOLD.
C
  999 VOLD=VLOSS
C
      RETURN
      END
```

```
LMAP1
 LIB LIS*LISPID.
 LIB LDC*OLDCALCOMP.
 NOT TPF$.
  .
 SEG MAIN
 IN LIS*LISPID.MAIN/.OVFLOW
  . ****************************************************
 SEG IOSEG*,(MAIN)
 IN LIS*LISPID.IOLISP/.DATEXP/.EIGS/.SYMIN/.SYSPRI/.MPRI/.RDATA
  . --------------------------------------------------
 SEG SYSSEG*,(MAIN)
 IN CK*TEST.SYST
  . --------------------------------------------------
 SEG PLOCOM*,(MAIN)
 IN LDC*LIB.SEQ
  . ****************************************************
 SEG PLTSEG*,(PLOCOM)
 IN LIS*LISPID.LIPLOT
  . --------------------------------------------------
 SEG RESSEG*,(PLOCOM)
 IN LIS*LISPID.RESLIS
  . ****************************************************
 SEG POWSEG*,(SYSSEG)
 IN LIS*LISPID.POWBRE
  . --------------------------------------------------
 SEG NUFSEG*,(SYSSEG)
 IN LIS*LISPID.NUFLET
  . --------------------------------------------------
 SEG GRDSEG*,(SYSSEG)
 IN LIS*LISPID.GRASD/.GRDCHK
  . ****************************************************
 SEG DATSEG*,( )
 IN DATA
  . --------------------------------------------------
 END . ZZZ
```

```
DAT1
 3    "NPAR
 3    "NTH
 -5.4     "PAR(1)
 18000.   "PAR(2)
 0.00079 "PAR(3)
 1    "ITH(1)
 2    "ITH(2)
 3    "ITH(3)
 0.01    "SCAL(1)
 1.E-6   "SCAL(2)
 10.     "SCAL(3)
 1    "IMIN
 1    "LOOP
 1    "NPRI(1)
 1    "NPRI(2)
 1    "NPRI(3)
 -1   "NPLOT
 40   "NP
 0    "IT
 2    "ISYS
 0    "MEAS
 1    "ISAMP
 10.  "TSAMP
 1    "NPRED1
 1    "NPRED2
 2    "NX
 1    "NU
 1    "NY
 1    "NOMAT(1)
 1    "NOMAT(2)
 0    "NOMAT(3)
 1    "NOMAT(4)
 0    "NOMAT(5)
 1    "NOMAT(6)
 2    "IACC
 0.   "HH
 4    "NPLOTC
 12   "NOL
 10.  "TS
 0.00001    "EPSI
 500  "NKAL
 1    "ICHK
 0    "ICR
    0.0   147.15
    0.0   147.13
    0.0   147.15
    0.0   147.24
    0.0   147.35
   10.0   147.41
   10.0   147.46
   10.0   147.24
   10.0   146.67
  -10.0   145.96
  -10.0   144.87
  -10.0   143.96
  -10.0   143.55
  -10.0   143.50
```

-10.0	143.81
10.0	144.38
10.0	145.17
10.0	145.77
10.0	145.92
-10.0	145.66
-10.0	145.20
-10.0	144.82
-10.0	144.84
-10.0	145.31
-10.0	146.10
10.0	147.17
10.0	148.43
10.0	149.44
10.0	150.05
10.0	150.29
0.0	150.21
0.0	149.79
0.0	149.55
0.0	149.37
10.0	149.26
10.0	149.15
10.0	148.82
10.0	148.16
-10.0	147.28
-10.0	146.29

```
C      PROGRAM MAIN2
C
C      MAIN PROGRAM FOR LISPID.
C      THE PARAMETERS ARE WRITTEN ON UNIT 106 WHEN LOOP .GT. 0.
C
C      AUTHOR, C. KALLSTROM 1978-07-19.
C
C      SUBROUTINE REQUIRED
C         IOLISP
C            MPRI
C            RTTFF (WITH SUBROUTINES)
C            LISDAT
C         DATEXP
C            MCORE
C         RDATA2
C            RTTFF (WITH SUBROUTINES)
C         LISPID (WITH SUBROUTINES)
C         MPRI
C         PARFIL
C         GRDCHK (WITH SUBROUTINES)
C
       DIMENSION TH(40),NPR(3)
C
       COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
      *NX,NU,NY,NTH,NOMAT(6)
C
       COMMON/DATA/V(1)
C
       COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
      *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
      *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
      *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
       COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
      *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
      *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
      *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
       COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
      *IERRSY
C
       EXTERNAL SYST2
C
       LP=6
C
C      PRINT HEADING.
C
       WRITE(LP,100)
   100 FORMAT(7H1LISPID/1X,6(1H*)/)
C
C      READ INPUT PARAMETERS FOR LISPID.
C
       CALL IOLISP(TH,1,0)
C
C      CHANGE STANDARD VALUES OF SOME INPUT PARAMETERS.
C
       MAXFN=500*NTH
       IDH=1
       DO 10 I=1,NTH
    10 DH(I)=HZ
       INU(2)=0
       INU(3)=0
C
```

```
C     ORGANIZE /DATA/ AND /INDEX/.
C
      CALL DATEXP
C
C     READ DATA.
C
      CALL RDATA2
C
C     SIMULATE THE SYSTEM IF LOOP .NE. -1.
C
      IF(LOOP)18,12,12
   12 DO 14 I=1,3
      NPR(I)=NPRI(I)
   14 NPRI(I)=0
      IACCO=IACC
      IACC=0
      NPLO=NPLOTC
      NPLOTC=0
      NPL=NPLOT
      NPLOT=0
      LOP=LOOP
      LOOP=0
C
      CALL LISPID(SYST2,TH,IERR)
C
      DO 16 I=1,3
   16 NPRI(I)=NPR(I)
      IACC=IACCO
      NPLOTC=NPLO
      NPLOT=NPL
      LOOP=LOP
C
C     PRINT INPUT PARAMETERS FOR LISPID.
C
   18 CALL IOLISP(TH,0,2)
C
      IF(ISYS .NE. -2) GO TO 20
      WRITE(LP,101)
  101 FORMAT(/10H MATRIX R1)
C
      CALL MPRI(V(IXR1+1),NX,NX,NX,8,0,IER)
C
C     IF LOOP .NE. -1 AND IF IERR .NE. 0 OR IERRSY .NE. 0 ,PRINT
C     WARNING AND PREPARE SIMULATION ONLY (NO PLOTTING).
C
   20 IF(LOOP)31,28,28
   28 IF(IERR)30,29,30
   29 IF(IERRSY)30,31,30
C
   30 WRITE(LP,102)IERR,IERRSY
  102 FORMAT(/36H SIMULATION ONLY, NO PLOTTING  IERR=,I2,9H  IERRSY=,
     *I2/)
C
      LOOP=0
      NPLOT=0
      IACC=0
      NPLOTC=0
      ICHK=0
C
C     PERFORM IDENTIFICATION OR SIMULATION.
C
   31 CALL LISPID(SYST2,TH,IERR)
C
      IF(LOOP)99,32,32
```

```
C
C      PRINT FINAL VALUES.
C
   32 WRITE(LP,104)
  104 FORMAT(1H1)
C
      CALL IOLISP(TH,0,2)
C
      IF(ISYS .NE. -2) GO TO 34
      WRITE(LP,101)
C
      CALL MPRI(V(IXR1+1),NX,NX,NX,8,0,IER)
C
C      WRITE PARAMETERS ON UNIT 106 IF LOOP .GT. 0.
C
   34 IF(LOOP .LE. 0) GO TO 36
C
      CALL PARFIL(PAR,NPAR)
C
   36 IF(ICHK.EQ.0)GO TO 99
C
      CALL GRDCHK(SYST2,TH,NTH,XM,HZ)
C
C
   99 STOP
      END
```

```
      SUBROUTINE RDATA2
C
C     SUBROUTINE CALLED FROM MAIN2 TO READ DATA FOR LISPID.
C
C     AUTHOR, C. KALLSTROM 1978-07-19.
C
C     SUBROUTINE REQUIRED
C        RTTFF (WITH SUBROUTINES)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      DEFINE UIN(I,J) = V(NP*J-NP+I)
      DEFINE YMS(I,J) = V(IXYMS +NP*J-NP+I)
      DEFINE TIM(I) = V(IXTIM+I)
C
      IC=1
      DO 50 I=1,NP
      IF(IT .EQ. 0) GO TO 10
      TIM(I)=RTTFF(IC)
10    WW=RTTFF(IC)
      UIN(I,1)=WW
      UIN(I,2)=1.
      IF(I .EQ. 1) WOLD=WW
      UIN(I,3)=WOLD
      WOLD=WW
30    DO 40 J=1,NY
40    YMS(I,J)=RTTFF(IC)
50    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE SYST2(TH,NTH1,VLOSS,DUM1,DUM2,IDUM1,IDUM2,IER)
C
C     SUBROUTINE TO SUPPLY A MODEL AND COMPUTE LOSS FUNCTION.
C     SHIP STEERING MODEL (CONSTANT SPEED).
C
C     AUTHOR: C. KALLSTROM 1978-07-19.
C
C     TH-   PARAMETER VECTOR OF DIMENSION NTH1.
C     NTH1- NUMBER OF PARAMETERS (MAX 40,MIN 2).
C           NTH1 IS EQUAL TO NTH IN /SYSPAR/ .
C     VLOSS-COMPUTED LOSS FUNCTION VALUE.
C     DUM1,DUM2,IDUM1,IDUM2 - DUMMY ARGUMENTS.
C     IER-  ERROR PARAMETER:
C           IER=0 IF OK.
C           IER=-1 IF ERROR IN SYST
C
C     COMMON BLOCK /COMSY/ (ALL VARIABLES EXCEPT IERRSY ARE GIVEN
C           VALUES IN IOLISP) :
C     PAR-  PARAMETER VECTOR OF DIMENSION NPAR. VECTOR TH(I)/SCAL(I)
C           IS A PART OF VECTOR PAR.
C     NPAR- NUMBER OF PARAMETERS (MAX 40,MIN 2). NTH MUST BE .LE. NPAR.
C     ITH-  INTEGER VECTOR OF DIMENSION NTH CONTAINING THE INDICES OF
C           THE PARAMETERS OF VECTOR PAR, FOR WHICH THE MINIMIZATION
C           IS PERFORMED.
C     SCAL- SCALING VECTOR OF DIMENSION NTH.
C           SCAL(I) MUST NOT BE EQUAL TO 0.
C     TS-   SAMPLING INTERVAL, ONLY USED IN SUBROUTINE SGAIN.
C     EPSI- TEST QUANTITY, ONLY USED IN SUBROUTINE SGAIN.
C     NKAL- MAXIMUM NUMBER OF ITERATIONS, ONLY USED IN
C           SUBROUTINE SGAIN (NO MAX,MIN 2).
C     VOLD- PUT VOLD INITIALLY EQUAL TO A LARGE POSITIVE VALUE (E.G.
C           1.E+10). WHEN SUBROUTINE SYST IS CALLED ,THE LOSS FUNCTION
C           VALUE VLOSS IS STORED IN VOLD, IF THE COMPUTATION IS SUCCES-
C           FULL , ELSE VLOSS IS PUT EQUAL TO 10.*VOLD AND THE VALUE IS
C           THEN RESTORED IN VOLD.
C     IERRSY-ERROR PARAMETER:
C           IERRSY=0 IF OK.
C           IERRSY=1 IF THE COMPUTATION OF VLOSS HAS FAILED.
C
C     NOTE : IF ISYS=2 OR -2 THE SUBROUTINE SGAIN IS CALLED
C     TO COMPUTE THE DISCRETE STATIONARY FILTER GAIN AK.
C
C     SUBROUTINE REQUIRED
C        TRANS
C           NORM
C        COSDY
C           COSA
C              NORM
C              EXPAN
C        SGAIN
C           TRANS
C              NORM
C           KALMAN
C              NORM
C              DESYM
C              SOLVS
C           NORM
C        SINT (WITH SUBROUTINES)
C        OVFLOW
C
      LOGICAL OVFLOW
C
      DIMENSION TH(1),NNTR(6)
```

```
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXPO,IXXO,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
      COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
     *IERRSY
C
      COMMON/TRANS2/S1(400),DUM3(1600)
C
      DEFINE A(I,J)=V(IXA+NX*J-NX+I)
      DEFINE B(I,J)=V(IXB+NX*J-NX+I)
      DEFINE C(I,J)=V(IXC+NY*J-NY+I)
      DEFINE D(I,J)=V(IXD+NY*J-NY+I)
      DEFINE R1(I,J)=V(IXR1+NX*J-NX+I)
      DEFINE R12(I,J)=V(IXR12+NX*J-NX+I)
      DEFINE R2(I,J)=V(IXR2+NY*J-NY+I)
      DEFINE XO(I)=V(IXXO+I)
      DEFINE PO(I,J)=V(IXPO+NX*J-NX+I)
      DEFINE AK(I,J)=V(IXAK+NX*J-NX+I)
      DEFINE UIN(I,J)=V(NP*J-NP+I)
      DEFINE YMS(I,J)=V(IXYMS+NP*J-NP+I)
      DEFINE TIM(I)=V(IXTIM+I)
      DEFINE BD(I,J)=V(IXBD+NX*J-NX+I)
C
C     INITIALIZING.
C
      IER=0
      IERRSY=0
      LP=6
      VMAX=1.E20
C
C     UPDATE THE PARAMETER VECTOR PAR USING TH.
C
      DO 10 I=1,NTH
      I1=ITH(I)
   10 PAR(I1)=TH(I)/SCAL(I)
C
C     SHIP STEERING MODEL.
C
      ALF1=0.01745
      ALF2=1.944
C
      A(1,1)=PAR(1)*PAR(3)/PAR(2)
      A(1,2)=PAR(1)*PAR(4)
      A(1,3)=0.
      A(2,1)=PAR(1)*PAR(5)/(PAR(2)*PAR(2))
      A(2,2)=PAR(1)*PAR(6)/PAR(2)
      A(2,3)=0.
      A(3,1)=0.
      A(3,2)=1.
      A(3,3)=0.
C
```

```
      B(1,1)=ALF1*PAR(1)*PAR(1)*PAR(7)/PAR(2)
      B(1,2)=PAR(9)
      B(1,3)=0.
      B(2,1)=ALF1*PAR(1)*PAR(1)*PAR(8)/(PAR(2)*PAR(2))
      B(2,2)=PAR(10)
      B(2,3)=0.
      B(3,1)=0.
      B(3,2)=0.
      B(3,3)=0.
C
      C(1,1)=ALF2
      C(1,2)=PAR(12)*ALF2
      C(1,3)=0.
      C(2,1)=0.
      C(2,2)=0.
      C(2,3)=1./ALF1
C
      D(1,1)=0.
      D(1,2)=PAR(11)
      D(1,3)=0.
      D(2,1)=0.
      D(2,2)=0.
      D(2,3)=0.
C
      R1(1,1)=ABS(PAR(13))
      R1(2,2)=ABS(PAR(14))
      WW=R1(1,1)*R1(2,2)
      R1(1,2)=SQRT(WW)*SIN(PAR(15))
      R1(2,1)=R1(1,2)
      R1(1,3)=0.
      R1(2,3)=0.
      R1(3,1)=0.
      R1(3,2)=0.
      R1(3,3)=0.
C
      R2(1,1)=ABS(PAR(16))
      R2(1,2)=0.
      R2(2,1)=0.
      R2(2,2)=ABS(PAR(17))
C
      P0(1,1)=ABS(PAR(18))
      P0(1,2)=0.
      P0(1,3)=0.
      P0(2,1)=0.
      P0(2,2)=ABS(PAR(19))
      P0(2,3)=0.
      P0(3,1)=0.
      P0(3,2)=0.
      P0(3,3)=ABS(PAR(20))
C
      X0(1)=PAR(21)
      X0(2)=PAR(22)
      X0(3)=ALF1*PAR(23)
C
      WW=SIN(PAR(24))
      TD=TS*ABS(WW)
C
      IF(ISYS .GE. 0) GO TO 200
C
C     SAMPLING OF MODEL IF ISYS LESS THAN ZERO.
C
      IF(TS .GT. 0.) GO TO 100
      IERRSY=1
      WRITE(LP,2000)
```

```
 2000 FORMAT(23H SYST2: TS NON-POSITIVE)
      GO TO 990
C
  100 IF(ISYS .EQ. -1) GO TO 106
C
      DO 102 I=1,6
  102 NNTR(I)=0
      IF(NOMAT(4) .NE. 0) NNTR(4)=1
      DO 104 I=1,NX2
      V(IXPPD+I)=V(IXR1+I)
  104 V(IXAD+I)=V(IXA+I)
C
      CALL TRANS(V(IXAD+1),DUM,DUM,DUM,V(IXR1+1),DUM,DUM,
     *   NX,IDUM,IDUM,NNTR,2,TS,EPST,NMAX,NOC,NX,NY)
C
      IF(NOC .EQ. 0) GO TO 106
      IERRSY=1
      WRITE(LP,2002)
 2002 FORMAT(22H TRANS: NO CONVERGENCE)
      GO TO 990
C
  106 CALL COSDY(V(IXA+1),V(IXB+1),V(IXAD+1),V(IXBD+1),S1,
     *   NX,NU-1,TS,TD,NX)
C
      DO 108 I=1,NX2
  108 V(IXA+I)=V(IXAD+I)
C
      DO 110 I=1,NX
      B(I,1)=S1(I)
      B(I,2)=BD(I,2)+S1(I+NX)
  110 B(I,3)=BD(I,1)
C
C     COMPUTATION OF DISCRETE FILTER GAIN AK IF ISYS=2 OR -2.
C     A CALL TO SGAIN MUST ONLY BE PERFORMED IF NO VALUES
C     ARE ASSIGNED TO AK EARLIER, WHEN ISYS=2 OR -2.
C
  200 IF(IABS(ISYS) .NE. 2) GO TO 300
C
      CALL SGAIN(TS,EPSI,NKAL,IERROR)
C
      IF(ISYS .NE. -2) GO TO 204
C
      DO 202 I=1,NX2
  202 V(IXR1+I)=V(IXPPD+I)
C
  204 IF(IERROR .EQ. 0) GO TO 300
C
      IERRSY=1
      WRITE(LP,1000) IERROR
 1000 FORMAT(15H SGAIN: IERROR=,I2)
      GO TO 990
C
C     ITERATE THE SYSTEM EQUATIONS.
C
  300 NTIME=1
      NSTOP=NP-NPRED2
      IF(NPRED2.EQ.1)NSTOP=NSTOP+1
C
      CALL SINT(NTIME,NSTOP,1,0,DUMMY,IERR)
C
      IF(IERR .EQ. 0) GO TO 302
      IERRSY=1
      WRITE(LP,1002)IERR,NTIME
 1002 FORMAT(12H SINT: IERR=,I2,8H  NTIME=,I6)
```

```
      GO TO 990
C
C     COMPUTE THE LOSS.
C
  302 CALL SINT(NTIME,NSTOP,0,4,VLOSS,IERR)
C
      IF(IERR .EQ. 0) GO TO 980
      IERRSY=1
      WRITE(LP,1004)IERR
 1004 FORMAT(12H SINT: IERR=,I2)
      GO TO 990
C
C     CHECK IF OVERFLOW.
C
  980 IF(.NOT.OVFLOW(IDUM)) GO TO 999
C
      WRITE(LP,1006)
 1006 FORMAT(15H SYST: OVERFLOW)
C
C     IF FAILURE, PUT VLOSS=MIN( 10.*VOLD , VMAX).
C
  990 VLOSS=AMIN1(10.*VOLD,VMAX)
C
C     STORE VLOSS IN VOLD.
C
  999 VOLD=VLOSS
C
      RETURN
      END
```

```
LMAP2
 LIB LIS*LISPID.
 LIB LDC*OLDCALCOMP.
 NOT TPF$.
  .
 SEG MAIN
 IN CK*TEST.MAIN2
 IN LIS*LISPID.OVFLOW
  . *******************************************************
 SEG IOSEG*,(MAIN)
 IN LIS*LISPID.IOLISP,.DATEXP,.EIGS,.SYMIN,.SYSPRI,.MPRI,.PARFIL
 IN CK*TEST.RDATA2
  . -----------------------------------------------------
 SEG SYSSEG*,(MAIN)
 IN CK*TEST.SYST2
  . -----------------------------------------------------
 SEG PLOCOM*,(MAIN)
 IN LDC*LIB.SEQ
  . *******************************************************
 SEG PLTSEG*,(PLOCOM)
 IN LIS*LISPID.LIPLOT
  . -----------------------------------------------------
 SEG RESSEG*,(PLOCOM)
 IN LIS*LISPID.RESLIS
  . *******************************************************
 SEG POWSEG*,(SYSSEG)
 IN LIS*LISPID.POWBRE
  . -----------------------------------------------------
 SEG NUFSEG*,(SYSSEG)
 IN LIS*LISPID.NUFLET
  . -----------------------------------------------------
 SEG GRDSEG*,(SYSSEG)
 IN LIS*LISPID.GRASD,.GRDCHK
  . *******************************************************
 SEG DATSEG*,( )
 IN DATA
  . -----------------------------------------------------
 END . ZZZ
```

```
DAT2
 24   "NPAR
 10   "NTH
 6.7    "PAR(1)
 350.
 -0.31
 -0.15
 -38.3
 -20.3
 0.14
 -1.15
 -0.00077
 -0.000071
 0.053
 164.4
 0.0000037
 0.027
 0.
 0.01
 0.01
 1.
 1.
 1.
 0.
 0.
 147.15
 2.0
 3    "ITH(1)
 4    "ITH(2)
 5    "ITH(3)
 6    "ITH(4)
 9    "ITH(5)
 10   "ITH(6)
 11   "ITH(7)
 13   "ITH(8)
 14   "ITH(9)
 24   "ITH(10)
 1.       "SCAL(1)
 1.       "SCAL(2)
 0.01     "SCAL(3)
 0.01     "SCAL(4)
 100.     "SCAL(5)
 1000.    "SCAL(6)
 10.      "SCAL(7)
 100.     "SCAL(8)
 1.       "SCAL(9)
 1.       "SCAL(10)
 2    "IMIN
 1    "LOOP
 1    "NPRI(1)
 1    "NPRI(2)
 1    "NPRI(3)
 1    "NPLOT
 38   "NP
 1    "IT
 -2   "ISYS
 0    "MEAS
 2    "ISAMP
```

```
10.  "TSAMP
5    "NPRED1
5    "NPRED2
3    "NX
3    "NU
2    "NY
1    "NOMAT(1)
1    "NOMAT(2)
1    "NOMAT(3)
1    "NOMAT(4)
0    "NOMAT(5)
1    "NOMAT(6)
1    "IACC
0.   "HH
4    "NPLOTC
12   "NOL
10.  "TS
0.00001    "EPSI
500  "NKAL
1    "ICHK
0    "ICR
  0.     0.0    0.00  147.15
 10.     0.0    0.05  147.13
 20.     0.0    0.06  147.15
 30.     0.0    0.00  147.24
 40.     0.0   -0.03  147.35
 50.    10.0   -0.03  147.41
 60.    10.0   -0.03  147.46
 70.    10.0   -0.03  147.24
 80.    10.0   -0.05  146.67
 90.   -10.0   -0.05  145.96
110.   -10.0   -0.05  143.96
120.   -10.0   -0.02  143.55
130.   -10.0   -0.05  143.50
140.   -10.0    0.02  143.81
150.    10.0    0.03  144.38
160.    10.0    0.02  145.17
170.    10.0   -0.02  145.77
180.    10.0    0.00  145.92
190.   -10.0    0.05  145.66
210.   -10.0    0.00  144.82
220.   -10.0    0.03  144.84
230.   -10.0    0.05  145.31
240.   -10.0    0.02  146.10
250.    10.0    0.00  147.17
260.    10.0    0.06  148.43
270.    10.0    0.03  149.44
280.    10.0    0.00  150.05
290.    10.0   -0.05  150.29
300.     0.0   -0.05  150.21
310.     0.0   -0.06  149.79
320.     0.0   -0.06  149.55
330.     0.0   -0.06  149.37
340.    10.0   -0.08  149.26
350.    10.0   -0.08  149.15
360.    10.0   -0.11  148.82
370.    10.0   -0.11  148.16
380.   -10.0   -0.08  147.28
390.   -10.0   -0.10  146.29
```

```
C       PROGRAM MAIN3
C
C       MAIN PROGRAM FOR LISPID.
C       THE PARAMETERS ARE WRITTEN ON UNIT 106 WHEN LOOP .GT. 0.
C
C       AUTHOR, C. KALLSTROM 1978-07-20.
C
C       SUBROUTINE REQUIRED
C          IOLISP
C             MPRI
C             RTTFF (WITH SUBROUTINES)
C             LISDAT
C          DATEXP
C             MCORE
C          RDATA3
C             RTTFF (WITH SUBROUTINES)
C          LISPID (WITH SUBROUTINES)
C          PARFIL
C          GRDCHK (WITH SUBROUTINES)
C
        DIMENSION TH(40),NPR(3)
C
        COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
       *NX,NU,NY,NTH,NOMAT(6)
C
        COMMON/DATA/V(1)
C
        COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
       *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXPO,IXXO,IXAD,IXBD,IXCD,
       *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
       *IXDLTR,IXPPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
        COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
       *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
       *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
       *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
        COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
       *IERRSY
C
        EXTERNAL SYST3
C
        LP=6
C
C       PRINT HEADING.
C
        WRITE(LP,100)
  100   FORMAT(7H1LISPID/1X,6(1H*)/)
C
C       READ INPUT PARAMETERS FOR LISPID.
C
        CALL IOLISP(TH,1,0)
C
C       CHANGE STANDARD VALUES OF SOME INPUT PARAMETERS.
C
        MAXFN=500*NTH
        IDH=1
        DO 10 I=1,NTH
   10   DH(I)=HZ
        INU(2)=0
C
C       ORGANIZE /DATA/ AND /INDEX/.
C
```

```
      CALL DATEXP
C
C     READ DATA.
C
      CALL RDATA3
C
C     SIMULATE THE SYSTEM IF LOOP .NE. -1.
C
      IF(LOOP)18,12,12
   12 DO 14 I=1,3
      NPR(I)=NPRI(I)
   14 NPRI(I)=0
      IACCO=IACC
      IACC=0
      NPLO=NPLOTC
      NPLOTC=0
      NPL=NPLOT
      NPLOT=0
      LOP=LOOP
      LOOP=0
C
      CALL LISPID(SYST3,TH,IERR)
C
      DO 16 I=1,3
   16 NPRI(I)=NPR(I)
      IACC=IACCO
      NPLOTC=NPLO
      NPLOT=NPL
      LOOP=LOP
C
C     PRINT INPUT PARAMETERS FOR LISPID.
C
   18 CALL IOLISP(TH,0,2)
C
C     IF LOOP .NE. -1 AND IF IERR .NE. 0 OR IERRSY .NE. 0 ,PRINT
C     WARNING AND PREPARE SIMULATION ONLY (NO PLOTTING).
C
   20 IF(LOOP)31,28,28
   28 IF(IERR)30,29,30
   29 IF(IERRSY)30,31,30
C
   30 WRITE(LP,102)IERR,IERRSY
  102 FORMAT(/36H SIMULATION ONLY, NO PLOTTING  IERR=,I2,9H  IERRSY=,
     *I2/)
C
      LOOP=0
      NPLOT=0
      IACC=0
      NPLOTC=0
      ICHK=0
C
C     PERFORM IDENTIFICATION OR SIMULATION.
C
   31 CALL LISPID(SYST3,TH,IERR)
C
      IF(LOOP)99,32,32
C
C     PRINT FINAL VALUES.
C
   32 WRITE(LP,104)
  104 FORMAT(1H1)
C
      CALL IOLISP(TH,0,2)
C
```

```
C     WRITE PARAMETERS ON UNIT 106 IF LOOP .GT. 0.
C
   34 IF(LOOP .LE. 0) GO TO 36
C
      CALL PARFIL(PAR,NPAR)
C
   36 IF(ICHK.EQ.0)GO TO 99
C
      CALL GRDCHK(SYST3,TH,NTH,XM,HZ)
C
C
   99 STOP
      END
```

```
      SUBROUTINE RDATA3
C
C     SUBROUTINE CALLED FROM MAIN3 TO READ DATA FOR LISPID.
C
C     AUTHOR, C. KALLSTROM 1978-07-20.
C
C     SUBROUTINE REQUIRED
C        RTTFF (WITH SUBROUTINES)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      DEFINE UIN(I,J) = V(NP*J-NP+I)
      DEFINE YMS(I,J) = V(IXYMS +NP*J-NP+I)
      DEFINE TIM(I) = V(IXTIM+I)
C
      IC=1
      DO 50 I=1,NP
      IF(IT .EQ. 0) GO TO 10
      TIM(I)=RTTFF(IC)
10    UIN(I,1)=RTTFF(IC)
      UIN(I,2)=1.
30    DO 40 J=1,NY
40    YMS(I,J)=RTTFF(IC)
50    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE SYST3(TH,NTH1,VLOSS,DUM1,DUM2,IDUM1,IDUM2,IER)
C
C     SUBROUTINE TO SUPPLY A MODEL AND COMPUTE LOSS FUNCTION.
C     SHIP STEERING MODEL (CONSTANT SPEED). VARYING SAMPLING INTERVAL.
C
C     AUTHOR: C. KALLSTROM 1978-07-20.
C
C     TH-    PARAMETER VECTOR OF DIMENSION NTH1.
C     NTH1-  NUMBER OF PARAMETERS (MAX 40,MIN 2).
C            NTH1 IS EQUAL TO NTH IN /SYSPAR/ .
C     VLOSS-COMPUTED LOSS FUNCTION VALUE.
C     DUM1,DUM2,IDUM1,IDUM2 - DUMMY ARGUMENTS.
C     IER-   ERROR PARAMETER:
C            IER=0 IF OK.
C            IER=-1 IF ERROR IN SYST
C
C     COMMON BLOCK /COMSY/ (ALL VARIABLES EXCEPT IERRSY ARE GIVEN
C            VALUES IN IOLISP) :
C     PAR-   PARAMETER VECTOR OF DIMENSION NPAR. VECTOR TH(I)/SCAL(I)
C            IS A PART OF VECTOR PAR.
C     NPAR-  NUMBER OF PARAMETERS (MAX 40,MIN 2). NTH MUST BE .LE. NPAR.
C     ITH-   INTEGER VECTOR OF DIMENSION NTH CONTAINING THE INDICES OF
C            THE PARAMETERS OF VECTOR PAR, FOR WHICH THE MINIMIZATION
C            IS PERFORMED.
C     SCAL-  SCALING VECTOR OF DIMENSION NTH.
C            SCAL(I) MUST NOT BE EQUAL TO 0.
C     TS-    SAMPLING INTERVAL, ONLY USED IN SUBROUTINE SGAIN.
C     EPSI-  TEST QUANTITY, ONLY USED IN SUBROUTINE SGAIN.
C     NKAL-  MAXIMUM NUMBER OF ITERATIONS, ONLY USED IN
C            SUBROUTINE SGAIN (NO MAX,MIN 2).
C     VOLD-  PUT VOLD INITIALLY EQUAL TO A LARGE POSITIVE VALUE (E.G.
C            1.E+10). WHEN SUBROUTINE SYST IS CALLED ,THE LOSS FUNCTION
C            VALUE VLOSS IS STORED IN VOLD, IF THE COMPUTATION IS SUCCES-
C            FULL , ELSE VLOSS IS PUT EQUAL TO 10.*VOLD AND THE VALUE IS
C            THEN RESTORED IN VOLD.
C     IERRSY-ERROR PARAMETER:
C            IERRSY=0 IF OK.
C            IERRSY=1 IF THE COMPUTATION OF VLOSS HAS FAILED.
C
C     NOTE : IF ISYS=2 OR -2 THE SUBROUTINE SGAIN IS CALLED
C     TO COMPUTE THE DISCRETE STATIONARY FILTER GAIN AK.
C
C     SUBROUTINE REQUIRED
C        SGAIN
C           TRANS
C              NORM
C           KALMAN
C              NORM
C              DESYM
C              SOLVS
C           NORM
C        SINT (WITH SUBROUTINES)
C        OVFLOW
C
      LOGICAL OVFLOW
C
      DIMENSION TH(1)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
```

```
       COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
      *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXPO,IXXO,IXAD,IXBD,IXCD,
      *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
      *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
       COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
      *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
      *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
      *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
       COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
      *IERRSY
C
       DEFINE A(I,J)=V(IXA+NX*J-NX+I)
       DEFINE B(I,J)=V(IXB+NX*J-NX+I)
       DEFINE C(I,J)=V(IXC+NY*J-NY+I)
       DEFINE D(I,J)=V(IXD+NY*J-NY+I)
       DEFINE R1(I,J)=V(IXR1+NX*J-NX+I)
       DEFINE R12(I,J)=V(IXR12+NX*J-NX+I)
       DEFINE R2(I,J)=V(IXR2+NY*J-NY+I)
       DEFINE XO(I)=V(IXXO+I)
       DEFINE PO(I,J)=V(IXPO+NX*J-NX+I)
       DEFINE AK(I,J)=V(IXAK+NX*J-NX+I)
       DEFINE UIN(I,J)=V(NP*J-NP+I)
       DEFINE YMS(I,J)=V(IXYMS+NP*J-NP+I)
       DEFINE TIM(I)=V(IXTIM+I)
C
C      INITIALIZING.
C
       IER=0
       IERRSY=0
       LP=6
       VMAX=1.E20
C
C      UPDATE THE PARAMETER VECTOR PAR USING TH.
C
       DO 10 I=1,NTH
       I1=ITH(I)
    10 PAR(I1)=TH(I)/SCAL(I)
C
C      SHIP STEERING MODEL.
C
       ALF1=0.01745
       ALF2=1.944
C
       A(1,1)=PAR(1)*PAR(3)/PAR(2)
       A(1,2)=PAR(1)*PAR(4)
       A(1,3)=0.
       A(2,1)=PAR(1)*PAR(5)/(PAR(2)*PAR(2))
       A(2,2)=PAR(1)*PAR(6)/PAR(2)
       A(2,3)=0.
       A(3,1)=0.
       A(3,2)=1.
       A(3,3)=0.
C
       B(1,1)=ALF1*PAR(1)*PAR(1)*PAR(7)/PAR(2)
       B(1,2)=PAR(9)
       B(2,1)=ALF1*PAR(1)*PAR(1)*PAR(8)/(PAR(2)*PAR(2))
       B(2,2)=PAR(10)
       B(3,1)=0.
       B(3,2)=0.
C
       C(1,1)=ALF2
       C(1,2)=PAR(12)*ALF2
```

```
      C(1,3)=0.
      C(2,1)=0.
      C(2,2)=0.
      C(2,3)=1./ALF1
C
      D(1,1)=0.
      D(1,2)=PAR(11)
      D(2,1)=0.
      D(2,2)=0.
C
      R1(1,1)=ABS(PAR(13))
      R1(2,2)=ABS(PAR(14))
      WW=R1(1,1)*R1(2,2)
      R1(1,2)=SQRT(WW)*SIN(PAR(15))
      R1(2,1)=R1(1,2)
      R1(1,3)=0.
      R1(2,3)=0.
      R1(3,1)=0.
      R1(3,2)=0.
      R1(3,3)=0.
C
      R2(1,1)=ABS(PAR(16))
      R2(1,2)=0.
      R2(2,1)=0.
      R2(2,2)=ABS(PAR(17))
C
      P0(1,1)=ABS(PAR(18))
      P0(1,2)=0.
      P0(1,3)=0.
      P0(2,1)=0.
      P0(2,2)=ABS(PAR(19))
      P0(2,3)=0.
      P0(3,1)=0.
      P0(3,2)=0.
      P0(3,3)=ABS(PAR(20))
C
      X0(1)=PAR(21)
      X0(2)=PAR(22)
      X0(3)=ALF1*PAR(23)
C
C     COMPUTATION OF DISCRETE FILTER GAIN AK IF ISYS=2 OR -2.
C     A CALL TO SGAIN MUST ONLY BE PERFORMED IF NO VALUES
C     ARE ASSIGNED TO AK EARLIER, WHEN ISYS=2 OR -2.
C
  200 IF(IABS(ISYS) .NE. 2) GO TO 300
C
      CALL SGAIN(TS,EPSI,NKAL,IERROR)
C
      IF(IERROR .EQ. 0) GO TO 300
C
      IERRSY=1
      WRITE(LP,1000) IERROR
 1000 FORMAT(15H SGAIN: IERROR=,I2)
      GO TO 990
C
C     ITERATE THE SYSTEM EQUATIONS.
C
  300 NTIME=1
      NSTOP=NP-NPRED2
      IF(NPRED2.EQ.1)NSTOP=NSTOP+1
C
      CALL SINT(NTIME,NSTOP,1,0,DUMMY,IERR)
C
      IF(IERR .EQ. 0) GO TO 302
```

```
      IERRSY=1
      WRITE(LP,1002)IERR,NTIME
 1002 FORMAT(12H SINT: IERR=,I2,8H  NTIME=,I6)
      GO TO 990
C
C     COMPUTE THE LOSS.
C
  302 CALL SINT(NTIME,NSTOP,0,4,VLOSS,IERR)
C
      IF(IERR .EQ. 0) GO TO 980
      IERRSY=1
      WRITE(LP,1004)IERR
 1004 FORMAT(12H SINT: IERR=,I2)
      GO TO 990
C
C     CHECK IF OVERFLOW.
C
  980 IF(.NOT.OVFLOW(IDUM)) GO TO 999
C
      WRITE(LP,1006)
 1006 FORMAT(15H SYST: OVERFLOW)
C
C     IF FAILURE, PUT VLOSS=MIN( 10.*VOLD , VMAX).
C
  990 VLOSS=AMIN1(10.*VOLD,VMAX)
C
C     STORE VLOSS IN VOLD.
C
  999 VOLD=VLOSS
C
      RETURN
      END
```

```
LMAP3
 LIB LIS*LISPID.
 LIB LDC*OLDCALCOMP.
 NOT TPF$.
  .
 SEG MAIN
 IN CK*TEST.MAIN3
 IN LIS*LISPID.OVFLOW
  . *******************************************************
 SEG IOSEG*,(MAIN)
 IN LIS*LISPID.IOLISP,.DATEXP,.EIGS,.SYMIN,.SYSPRI,.MPRI,.PARFIL
 IN CK*TEST.RDATA3
  . ----------------------------------------------------
 SEG SYSSEG*,(MAIN)
 IN CK*TEST.SYST3
  . ----------------------------------------------------
 SEG PLOCOM*,(MAIN)
 IN LDC*LIB.SEQ
  . *******************************************************
 SEG PLTSEG*,(PLOCOM)
 IN LIS*LISPID.LIPLOT
  . ----------------------------------------------------
 SEG RESSEG*,(PLOCOM)
 IN LIS*LISPID.RESLIS
  . *******************************************************
 SEG POWSEG*,(SYSSEG)
 IN LIS*LISPID.POWBRE
  . ----------------------------------------------------
 SEG NUFSEG*,(SYSSEG)
 IN LIS*LISPID.NUFLET
  . ----------------------------------------------------
 SEG GRDSEG*,(SYSSEG)
 IN LIS*LISPID.GRASD,.GRDCHK
  . *******************************************************
 SEG DATSEG*,( )
 IN DATA
  . ----------------------------------------------------
 END . ZZZ
```

```
DAT3
 23  "NPAR
 12  "NTH
 6.7    "PAR(1)
 350.
 -0.31
 -0.15
 -38.3
 -20.3
 0.14
 -1.15
 -0.00077
 -0.000071
 0.053
 164.4
 0.0000037
 0.027
 0.
 0.01
 0.01
 50.
 5.
 50.
 0.
 0.
 147.15
 3    "ITH(1)
 4    "ITH(2)
 5    "ITH(3)
 6    "ITH(4)
 9    "ITH(5)
 10   "ITH(6)
 11   "ITH(7)
 13   "ITH(8)
 14   "ITH(9)
 18   "ITH(10)
 19   "ITH(11)
 20   "ITH(12)
 1.       "SCAL(1)
 1.       "SCAL(2)
 0.01     "SCAL(3)
 0.01     "SCAL(4)
 100.     "SCAL(5)
 1000.    "SCAL(6)
 10.      "SCAL(7)
 100.     "SCAL(8)
 1.       "SCAL(9)
 0.1      "SCAL(10)
 1.       "SCAL(11)
 0.1      "SCAL(12)
 1    "IMIN
 0    "LOOP
 1    "NPRI(1)
 1    "NPRI(2)
 1    "NPRI(3)
 -1   "NPLOT
 38   "NP
 1    "IT
```

```
3    "ISYS
0    "MEAS
3    "ISAMP
-1.  "TSAMP
3    "NPRED1
5    "NPRED2
3    "NX
2    "NU
2    "NY
1    "NOMAT(1)
1    "NOMAT(2)
1    "NOMAT(3)
1    "NOMAT(4)
0    "NOMAT(5)
1    "NOMAT(6)
0    "IACC
0.   "HH
4    "NPLOTC
12   "NOL
-1.  "TS
0.00001    "EPSI
500  "NKAL
0    "ICHK
0    "ICR
   0.     0.0    0.00   147.15
  10.     0.0    0.05   147.13
  20.     0.0    0.06   147.15
  30.     0.0    0.00   147.24
  40.     0.0   -0.03   147.35
  50.    10.0   -0.03   147.41
  60.    10.0   -0.03   147.46
  70.    10.0   -0.03   147.24
  82.    10.0   -0.05   146.67
  90.   -10.0   -0.05   145.96
 110.   -10.0   -0.05   143.96
 120.   -10.0   -0.02   143.55
 130.   -10.0   -0.05   143.50
 140.   -10.0    0.02   143.81
 150.    10.0    0.03   144.38
 160.    10.0    0.02   145.17
 170.    10.0   -0.02   145.77
 180.    10.0    0.00   145.92
 190.   -10.0    0.05   145.66
 210.   -10.0    0.00   144.82
 220.   -10.0    0.03   144.84
 230.   -10.0    0.05   145.31
 240.   -10.0    0.02   146.10
 250.    10.0    0.00   147.17
 260.    10.0    0.06   148.43
 270.    10.0    0.03   149.44
 284.    10.0    0.00   150.05
 290.    10.0   -0.05   150.29
 300.     0.0   -0.05   150.21
 310.     0.0   -0.06   149.79
 320.     0.0   -0.06   149.55
 330.     0.0   -0.06   149.37
 340.    10.0   -0.08   149.26
 350.    10.0   -0.08   149.15
 360.    10.0   -0.11   148.82
 370.    10.0   -0.11   148.16
 380.   -10.0   -0.08   147.28
 390.   -10.0   -0.10   146.29
```

```
C      PROGRAM MAIN4
C
C      MAIN PROGRAM FOR LISPID.
C      THE PARAMETERS ARE WRITTEN ON UNIT 106 WHEN LOOP .GT. 0.
C
C      AUTHOR, C. KALLSTROM 1978-07-21.
C
C      SUBROUTINE REQUIRED
C         IOLISP
C            MPRI
C            RTTFF (WITH SUBROUTINES)
C            LISDAT
C         DATEXP
C            MCORE
C         RDATA4
C            RTTFF (WITH SUBROUTINES)
C         LISPID (WITH SUBROUTINES)
C         MPRI
C         PARFIL
C         GRDCHK (WITH SUBROUTINES)
C
      DIMENSION TH(40),NPR(3)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXPPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
      COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
     *IERRSY
C
      COMMON/SPEED/VSP(40)
C
      EXTERNAL SYST4
C
      LP=6
C
C      PRINT HEADING.
C
      WRITE(LP,100)
  100 FORMAT(7H1LISPID/1X,6(1H*)/)
C
C      READ INPUT PARAMETERS FOR LISPID.
C
      CALL IOLISP(TH,1,0)
C
C      CHANGE STANDARD VALUES OF SOME INPUT PARAMETERS.
C
      MAXFN=500*NTH
      IDH=1
      DO 10 I=1,NTH
   10 DH(I)=HZ
      INU(2)=0
```

```
C
C      ORGANIZE /DATA/ AND /INDEX/.
C
       CALL DATEXP
C
C      READ DATA.
C
       CALL RDATA4
C
C      SIMULATE THE SYSTEM IF LOOP .NE. -1.
C
       IF(LOOP)18,12,12
    12 DO 14 I=1,3
       NPR(I)=NPRI(I)
    14 NPRI(I)=0
       IACCO=IACC
       IACC=0
       NPLO=NPLOTC
       NPLOTC=0
       NPL=NPLOT
       NPLOT=0
       LOP=LOOP
       LOOP=0
C
       CALL LISPID(SYST4,TH,IERR)
C
       DO 16 I=1,3
    16 NPRI(I)=NPR(I)
       IACC=IACCO
       NPLOTC=NPLO
       NPLOT=NPL
       LOOP=LOP
C
C      PRINT SPEED MEASUREMENTS.
C
    18 WRITE(LP,101)
   101 FORMAT(//27H SPEED MEASUREMENTS (KNOTS))
C
       CALL MPRI(VSP,1,NP,1,8,0,IER)
C
C      PRINT INPUT PARAMETERS FOR LISPID.
C
       CALL IOLISP(TH,0,2)
C
C      IF LOOP .NE. -1 AND IF IERR .NE. 0 OR IERRSY .NE. 0 ,PRINT
C      WARNING AND PREPARE SIMULATION ONLY (NO PLOTTING).
C
    20 IF(LOOP)31,28,28
    28 IF(IERR)30,29,30
    29 IF(IERRSY)30,31,30
C
    30 WRITE(LP,102)IERR,IERRSY
   102 FORMAT(/36H SIMULATION ONLY, NO PLOTTING  IERR=,I2,9H  IERRSY=,
      *I2/)
C
       LOOP=0
       NPLOT=0
       IACC=0
       NPLOTC=0
       ICHK=0
C
C      PERFORM IDENTIFICATION OR SIMULATION.
C
    31 CALL LISPID(SYST4,TH,IERR)
```

```
C
      IF(LOOP)99,32,32
C
C     PRINT FINAL VALUES.
C
   32 WRITE(LP,104)
  104 FORMAT(1H1)
C
      CALL IOLISP(TH,0,2)
C
C     WRITE PARAMETERS ON UNIT 106 IF LOOP .GT. 0.
C
   34 IF(LOOP .LE. 0) GO TO 36
C
      CALL PARFIL(PAR,NPAR)
C
   36 IF(ICHK.EQ.0)GO TO 99
C
      CALL GRDCHK(SYST4,TH,NTH,XM,HZ)
C
C
   99 STOP
      END
```

```
      SUBROUTINE RDATA4
C
C     SUBROUTINE CALLED FROM MAIN4 TO READ DATA FOR LISPID.
C
C     AUTHOR, C. KALLSTROM 1978-07-21.
C
C     SUBROUTINE REQUIRED
C        RTTFF (WITH SUBROUTINES)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXRPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/SPEED/VSP(40)
C
      DEFINE UIN(I,J) = V(NP*J-NP+I)
      DEFINE YMS(I,J) = V(IXYMS +NP*J-NP+I)
      DEFINE TIM(I) = V(IXTIM+I)
C
      IC=1
      DO 50 I=1,NP
      IF(IT .EQ. 0) GO TO 10
      TIM(I)=RTTFF(IC)
10    UIN(I,1)=RTTFF(IC)
      UIN(I,2)=1.
30    DO 40 J=1,NY
40    YMS(I,J)=RTTFF(IC)
      VSP(I)=RTTFF(IC)
50    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE SYST4(TH,NTH1,VLOSS,DUM1,DUM2,IDUM1,IDUM2,IER)
C
C     SUBROUTINE TO SUPPLY A MODEL AND COMPUTE LOSS FUNCTION.
C     SHIP STEERING MODEL (VARYING SPEED). VARYING SAMPLING INTERVAL.
C
C     AUTHOR: C. KALLSTROM 1978-07-21.
C
C     TH-    PARAMETER VECTOR OF DIMENSION NTH1.
C     NTH1-  NUMBER OF PARAMETERS (MAX 40,MIN 2).
C            NTH1 IS EQUAL TO NTH IN /SYSPAR/ .
C     VLOSS-COMPUTED LOSS FUNCTION VALUE.
C     DUM1,DUM2,IDUM1,IDUM2 - DUMMY ARGUMENTS.
C     IER-   ERROR PARAMETER:
C            IER=0 IF OK.
C            IER=-1 IF ERROR IN SYST
C
C     COMMON BLOCK /COMSY/ (ALL VARIABLES EXCEPT IERRSY ARE GIVEN
C            VALUES IN IOLISP) :
C     PAR-   PARAMETER VECTOR OF DIMENSION NPAR. VECTOR TH(I)/SCAL(I)
C            IS A PART OF VECTOR PAR.
C     NPAR-  NUMBER OF PARAMETERS (MAX 40,MIN 2). NTH MUST BE .LE. NPAR.
C     ITH-   INTEGER VECTOR OF DIMENSION NTH CONTAINING THE INDICES OF
C            THE PARAMETERS OF VECTOR PAR, FOR WHICH THE MINIMIZATION
C            IS PERFORMED.
C     SCAL-  SCALING VECTOR OF DIMENSION NTH.
C            SCAL(I) MUST NOT BE EQUAL TO 0.
C     TS-    SAMPLING INTERVAL, ONLY USED IN SUBROUTINE SGAIN.
C     EPSI-  TEST QUANTITY, ONLY USED IN SUBROUTINE SGAIN.
C     NKAL-  MAXIMUM NUMBER OF ITERATIONS, ONLY USED IN
C            SUBROUTINE SGAIN (NO MAX,MIN 2).
C     VOLD-  PUT VOLD INITIALLY EQUAL TO A LARGE POSITIVE VALUE (E.G.
C            1.E+10). WHEN SUBROUTINE SYST IS CALLED ,THE LOSS FUNCTION
C            VALUE VLOSS IS STORED IN VOLD, IF THE COMPUTATION IS SUCCES-
C            FULL , ELSE VLOSS IS PUT EQUAL TO 10.*VOLD AND THE VALUE IS
C            THEN RESTORED IN VOLD.
C     IERRSY=ERROR PARAMETER:
C            IERRSY=0 IF OK.
C            IERRSY=1 IF THE COMPUTATION OF VLOSS HAS FAILED.
C
C     NOTE : IF ISYS=2 OR -2 THE SUBROUTINE SGAIN IS CALLED
C     TO COMPUTE THE DISCRETE STATIONARY FILTER GAIN AK.
C
C     SUBROUTINE REQUIRED
C        SGAIN
C           TRANS
C              NORM
C           KALMAN
C              NORM
C              DESYM
C              SOLVS
C           NORM
C        SINT (WITH SUBROUTINES)
C        OVFLOW
C
      LOGICAL OVFLOW
C
      DIMENSION TH(1)
C
      COMMON/SYSPAR/NP,IT,ISYS,MEAS,ISAMP,TSAMP,NPRED1,NPRED2,
     *NX,NU,NY,NTH,NOMAT(6)
C
      COMMON/DATA/V(1)
C
```

```
      COMMON/INDEX/IXYMS,IXTIM,IXYMOD,IXERMD,IXEPS,IXIEPS,IXA,IXB,
     *IXC,IXD,IXR1,IXR12,IXR2,IXAK,IXP0,IXX0,IXAD,IXBD,IXCD,
     *IXDD,IXR1D,IXR12D,IXR2D,IXAKD,IXP,IXX,IXRR,IXPPD,
     *IXDLTR,IXPPD,IXRNY,IXV1,IXV2,NX2,NY2,NXNY,NXNU,NYNU
C
      COMMON/LISCON/LOOP,NPRI(3),NPLOT,ITRAN,NMAX,EPST,EPSK,CHISQ,IMIN,
     *DFN,HZ,EPZ,MODE,MAXFN,IPRINT,XM(50),DIST,SCALX,TEPS,NSTP,ILLC,
     *IPRIN,IND,IF,IPLOT,ITEXT,IIY,SX,SY,IDH,HH,DH(40),IACC,NPLOTC,NOL,
     *INU(20),INY(20),ICHK,LDUM1,LDUM2,RDUM1,RDUM2
C
      COMMON/COMSY/PAR(40),NPAR,ITH(40),SCAL(40),TS,EPSI,NKAL,VOLD,
     *IERRSY
C
      COMMON/SPEED/VSP(40)
C
      DEFINE A(I,J)=V(IXA+NX*J-NX+I)
      DEFINE B(I,J)=V(IXB+NX*J-NX+I)
      DEFINE C(I,J)=V(IXC+NY*J-NY+I)
      DEFINE D(I,J)=V(IXD+NY*J-NY+I)
      DEFINE R1(I,J)=V(IXR1+NX*J-NX+I)
      DEFINE R12(I,J)=V(IXR12+NX*J-NX+I)
      DEFINE R2(I,J)=V(IXR2+NY*J-NY+I)
      DEFINE X0(I)=V(IXX0+I)
      DEFINE P0(I,J)=V(IXP0+NX*J-NX+I)
      DEFINE AK(I,J)=V(IXAK+NX*J-NX+I)
      DEFINE UIN(I,J)=V(NP*J-NP+I)
      DEFINE YMS(I,J)=V(IXYMS+NP*J-NP+I)
      DEFINE TIM(I)=V(IXTIM+I)
C
C     INITIALIZING.
C
      IER=0
      IERRSY=0
      LP=6
      VMAX=1.E20
C
C     UPDATE THE PARAMETER VECTOR PAR USING TH.
C
      DO 10 I=1,NTH
      I1=ITH(I)
   10 PAR(I1)=TH(I)/SCAL(I)
C
C     SHIP STEERING MODEL.
C
      ALF1=0.01745
      ALF2=1.944
C
      A11=PAR(3)/PAR(2)
      A12=PAR(4)
      A(1,3)=0.
      A21=PAR(5)/(PAR(2)*PAR(2))
      A22=PAR(6)/PAR(2)
      A(2,3)=0.
      A(3,1)=0.
      A(3,2)=1.
      A(3,3)=0.
C
      B11=ALF1*PAR(7)/PAR(2)
      B(1,2)=PAR(9)
      B21=ALF1*PAR(8)/(PAR(2)*PAR(2))
      B(2,2)=PAR(10)
      B(3,1)=0.
      B(3,2)=0.
C
```

```
      C(1,1)=ALF2
      C(1,2)=PAR(12)*ALF2
      C(1,3)=0.
      C(2,1)=0.
      C(2,2)=0.
      C(2,3)=1./ALF1
C
      D(1,1)=0.
      D(1,2)=PAR(11)
      D(2,1)=0.
      D(2,2)=0.
C
      R1(1,1)=ABS(PAR(13))
      R1(2,2)=ABS(PAR(14))
      WW=R1(1,1)*R1(2,2)
      R1(1,2)=SQRT(WW)*SIN(PAR(15))
      R1(2,1)=R1(1,2)
      R1(1,3)=0.
      R1(2,3)=0.
      R1(3,1)=0.
      R1(3,2)=0.
      R1(3,3)=0.
C
      R2(1,1)=ABS(PAR(16))
      R2(1,2)=0.
      R2(2,1)=0.
      R2(2,2)=ABS(PAR(17))
C
      P0(1,1)=ABS(PAR(18))
      P0(1,2)=0.
      P0(1,3)=0.
      P0(2,1)=0.
      P0(2,2)=ABS(PAR(19))
      P0(2,3)=0.
      P0(3,1)=0.
      P0(3,2)=0.
      P0(3,3)=ABS(PAR(20))
C
      X0(1)=PAR(21)
      X0(2)=PAR(22)
      X0(3)=ALF1*PAR(1)
C
C     COMPUTATION OF DISCRETE FILTER GAIN AK IF ISYS=2 OR -2.
C     A CALL TO SGAIN MUST ONLY BE PERFORMED IF NO VALUES
C     ARE ASSIGNED TO AK EARLIER, WHEN ISYS=2 OR -2.
C
  200 IF(IABS(ISYS) .NE. 2) GO TO 300
C
      CALL SGAIN(TS,EPSI,NKAL,IERROR)
C
      IF(IERROR .EQ. 0) GO TO 300
C
      IERRSY=1
      WRITE(LP,1000) IERROR
 1000 FORMAT(15H SGAIN: IERROR=,I2)
      GO TO 990
C
C     ITERATE THE SYSTEM EQUATIONS.
C
  300 NTIME=1
      NSTOP=NP-NPRED2
      IF(NPRED2.EQ.1)NSTOP=NSTOP+1
      INI=1
  301 WW=VSP(NTIME)/ALF2
```

```
      A(1,1)=WW*A11
      A(1,2)=WW*A12
      A(2,1)=WW*A21
      A(2,2)=WW*A22
      B(1,1)=WW*WW*B11
      B(2,1)=WW*WW*B21
C
      CALL SINT(NTIME,NTIME,INI,0,DUMMY,IERR)
C
      INI=0
      IF(IERR .EQ. 0) GO TO 302
      IERRSY=1
      WRITE(LP,1002)IERR,NTIME
 1002 FORMAT(12H SINT: IERR=,I2,8H  NTIME=,I6)
      GO TO 990
C
  302 IF(NTIME .EQ. NSTOP) GO TO 304
      NTIME=NTIME+1
      GO TO 301
C
C     COMPUTE THE LOSS.
C
  304 CALL SINT(NTIME,NSTOP,0,4,VLOSS,IERR)
C
      IF(IERR .EQ. 0) GO TO 980
      IERRSY=1
      WRITE(LP,1004)IERR
 1004 FORMAT(12H SINT: IERR=,I2)
      GO TO 990
C
C     CHECK IF OVERFLOW.
C
  980 IF(.NOT.OVFLOW(IDUM)) GO TO 999
C
      WRITE(LP,1006)
 1006 FORMAT(15H SYST: OVERFLOW)
C
C     IF FAILURE, PUT VLOSS=MIN( 10.*VOLD , VMAX).
C
  990 VLOSS=AMIN1(10.*VOLD,VMAX)
C
C     STORE VLOSS IN VOLD.
C
  999 VOLD=VLOSS
C
      RETURN
      END
```

```
LMAP4
 LIB LIS*LISPID.
 LIB LDC*OLDCALCOMP.
 NOT TPF$.
  .
 SEG MAIN
 IN CK*TEST.MAIN4
 IN LIS*LISPID.OVFLOW
  . ********************************************************
 SEG IOSEG*,(MAIN)
 IN LIS*LISPID.IOLISP,.DATEXP,.EIGS,.SYMIN,.SYSPRI,.MPRI,.PARFIL
 IN CK*TEST.RDATA4
  . ------------------------------------------------------
 SEG SYSSEG*,(MAIN)
 IN CK*TEST.SYST4
  . ------------------------------------------------------
 SEG PLOCOM*,(MAIN)
 IN LDC*LIB.SEQ
  . ********************************************************
 SEG PLTSEG*,(PLOCOM)
 IN LIS*LISPID.LIPLOT
  . ------------------------------------------------------
 SEG RESSEG*,(PLOCOM)
 IN LIS*LISPID.RESLIS
  . ********************************************************
 SEG POWSEG*,(SYSSEG)
 IN LIS*LISPID.POWBRE
  . ------------------------------------------------------
 SEG NUFSEG*,(SYSSEG)
 IN LIS*LISPID.NUFLET
  . ------------------------------------------------------
 SEG GRDSEG*,(SYSSEG)
 IN LIS*LISPID.GRASD,.GRDCHK
  . ********************************************************
 SEG DATSEG*,( )
 IN DATA
  . ------------------------------------------------------
 END . ZZZ
```

```
DAT4
 22   "NPAR
 12   "NTH
 147.15    "PAR(1)
 350.
 -0.31
 -0.15
 -38.3
 -20.3
 0.14
 -1.15
 -0.00077
 -0.000071
 0.053
 164.4
 0.0000037
 0.027
 0.
 0.01
 0.01
 50.
 5.
 50.
 0.
 0.
 3    "ITH(1)
 4    "ITH(2)
 5    "ITH(3)
 6    "ITH(4)
 9    "ITH(5)
 10   "ITH(6)
 11   "ITH(7)
 13   "ITH(8)
 14   "ITH(9)
 18   "ITH(10)
 19   "ITH(11)
 20   "ITH(12)
 1.       "SCAL(1)
 1.       "SCAL(2)
 0.01     "SCAL(3)
 0.01     "SCAL(4)
 100.     "SCAL(5)
 1000.    "SCAL(6)
 10.      "SCAL(7)
 100.     "SCAL(8)
 1.       "SCAL(9)
 0.1      "SCAL(10)
 1.       "SCAL(11)
 0.1      "SCAL(12)
 1    "IMIN
 1    "LOOP
 1    "NPRI(1)
 1    "NPRI(2)
 1    "NPRI(3)
 -1   "NPLOT
 38   "NP
 1    "IT
```

```
3    "ISYS
0    "MEAS
3    "ISAMP
-1.  "TSAMP
1    "NPRED1
1    "NPRED2
3    "NX
2    "NU
2    "NY
1    "NOMAT(1)
1    "NOMAT(2)
1    "NOMAT(3)
1    "NOMAT(4)
0    "NOMAT(5)
1    "NOMAT(6)
2    "IACC
0.   "HH
4    "NPLOTC
12   "NOL
-1.  "TS
0.00001    "EPSI
500  "NKAL
1    "ICHK
0    "ICR
  0.      0.0    0.00   147.15   13.04
 10.      0.0    0.05   147.13   13.07
 20.      0.0    0.06   147.15   13.20
 30.      0.0    0.00   147.24   13.17
 40.      0.0   -0.03   147.35   13.15
 50.     10.0   -0.03   147.41   13.10
 60.     10.0   -0.03   147.46   13.09
 70.     10.0   -0.03   147.24   13.10
 82.     10.0   -0.05   146.67   13.12
 90.    -10.0   -0.05   145.96   13.10
110.    -10.0   -0.05   143.96   13.02
120.    -10.0   -0.02   143.55   13.04
130.    -10.0   -0.05   143.50   13.04
140.    -10.0    0.02   143.81   12.99
150.     10.0    0.03   144.38   12.99
160.     10.0    0.02   145.17   13.01
170.     10.0   -0.02   145.77   13.04
180.     10.0    0.00   145.92   13.04
190.    -10.0    0.05   145.66   13.04
210.    -10.0    0.00   144.82   12.98
220.    -10.0    0.03   144.84   12.98
230.    -10.0    0.05   145.31   12.98
240.    -10.0    0.02   146.10   12.93
250.     10.0    0.00   147.17   12.88
260.     10.0    0.06   148.43   12.89
270.     10.0    0.03   149.44   12.88
284.     10.0    0.00   150.05   13.10
290.     10.0   -0.05   150.29   12.88
300.      0.0   -0.05   150.21   12.83
310.      0.0   -0.06   149.79   12.86
320.      0.0   -0.06   149.55   12.88
330.      0.0   -0.06   149.37   12.86
340.     10.0   -0.08   149.26   12.88
350.     10.0   -0.08   149.15   12.86
360.     10.0   -0.11   148.82   12.89
370.     10.0   -0.11   148.16   12.86
380.    -10.0   -0.08   147.28   12.86
390.    -10.0   -0.10   146.29   12.86
```

APPENDIX G - EXECUTION OF A TEST EXAMPLE

A complete description of the generation and execution of test example 1 (see Section 6 and Appendix F) is given in this appendix. The commands are **typed** from a terminal and the run is executed in demand mode. It is assumed that the LISPID file has been transfered from magnetic tape to disc according to Appendix B before the run starts.

```
 ENTER USERID/PASSWORD:
>U/P*
*UNIVAC 1100 OPERATING SYSTEM  VER. 32.R2BX-D399(RSI)*
>@RUN CK,208876,CK
NOW IS 14:47 TUESDAY 18 JUL 78
>@ASG,AX LIS*LISPID.
READY
>@CAT,P TEST.,F2
READY
>@ASG,AX TEST.
READY
>@REGISTER TEST./60
LDC FILE REGISTRATION PROCESSOR. EXECUTION STARTED 78-07-18 AT 14:48:37

THE FILE CK*TEST IN PROJECT CK
HAS BEEN REGISTERED ON ACCOUNT 208876.
IT WILL BE DELETED WHEN IT HAS NOT BEEN REFERENCED FOR 60 DAYS.

END OF EXECUTION.
>@COPY,S LIS*LISPID.SYST,TEST.
FURPUR 27R1 RL72-9 07/18/78 14:49:13
 1 SYM
>@COPY,S LIS*LISPID.LISMAP,TEST.LMAP1
 1 SYM
>@ED,U TEST.SYST
CASE UPPER ASSUMED
ED 15R2-TUE-07/18/78-14:50:04-(0,1)
EDIT
0:>N 3
C      SUBROUTINE TO SUPPLY A MODEL AND COMPUTE LOSS FUNCTION.
3:>I C      SIMPLE MODEL OF SHIP STEERING.
4:>N 2
C      AUTHOR:
6:>C /:/: C. KALLSTROM 1978-06-29./
C      AUTHOR: C. KALLSTROM 1978-06-29.
6:>L MATHEMATICAL MODEL
C      MATHEMATICAL MODEL.
104:>R C      SIMPLE SHIP STEERING MODEL.
104:>N
C
105:>
INPUT
106I:>       A(1,1)=-1./PAR(2)
107I:>       A(1,2)=0.
108I:>       A(2,1)=1.
109I:>       A(2,2)=0.
110I:>       B(1,1)=PAR(1)/PAR(2)
111I:>       B(2,1)=0.
112I:>       C(1,1)=0.
113I:>       C(1,2)=1.
114I:>       R1(1,1)=ABS(PAR(3))
115I:>       R1(1,2)=0.
116I:>       R1(2,1)=0.
117I:>       R1(2,2)=0.
118I:>       R2(1,1)=0.01
119I:>       P0(1,1)=1.
120I:>       P0(1,2)=0.
121I:>       P0(2,1)=0.
122I:>       P0(2,2)=1.
123I:>       X0(1)=0.
124I:>       X0(2)=147.15
125I:>
EDIT
124:>EXI
LINES:181 FIELDATA
```

```
>@SUSPEND
SUSPENDED
>@FOR,DS TEST.SYST
>@RESUME,E
READ-ONLY MODE
CASE UPPER ASSUMED
ED 15R2-TUE-07/18/78-15:01:42-(0,)
EDIT
0:>F END FOR
END FOR
231:>EXI
NO CORRECTIONS APPLIED.
>@RESUME,P ,PR
SENT BY CK TO PR
>@ED,U TEST.LMAP1
CASE UPPER ASSUMED
ED 15R2-TUE-07/18/78-15:02:40-(0,1)
EDIT
0:>L XXX
IN XXX.SYST
12:>C /XXX/CK*TEST/
IN CK*TEST.SYST
12:>EXI
LINES:35 FIELDATA
>@PREP TEST.
FURPUR 27R1 RL72-9 07/18/78 15:04:24
END PREP.
>@PRT,T TEST.

CK*TEST(1)
FOR    SYST(1)
REL    SYST
ELT    LMAP1(1)
>@SUSPEND
SUSPENDED
>@MAP,S TEST.LMAP1,TEST.LABS1
>@RESUME,E
READ-ONLY MODE
CASE UPPER ASSUMED
ED 15R2-TUE-07/18/78-15:07:27-(0,)
EDIT
0:>L ZZZ
   35.        END . ZZZ
37:>P 10
   35.        END . ZZZ

ADDRESS LIMITS     001000 036721      15314 IBANK WORDS DECIMAL
                   040000 074346      14567 DBANK WORDS DECIMAL
SEGMENT LOAD TABLE        040000 040047
INDIRECT LOAD TABLE       040050 040424
STARTING ADDRESS   015470



46:>F END MAP
END MAP
367:>EXI
NO CORRECTIONS APPLIED.
```

```
>@RESUME,P ,PR
SENT BY CK TO PR
>@ED,I TEST.DAT1
CASE UPPER ASSUMED
ED 15R2-TUE-07/18/78-15:15:43-(,0)
INPUT
1I:>3    "NPAR
2I:>3    "NTH
3I:>-5.4      "PAR(1)
4I:>18000.   "PAR(2)
5I:>0.00079 "PAR(3)
6I:>1    "ITH(1)
7I:>2    "ITH(2)
8I:>3    "ITH(3)
9I:>0.01     "SCAL(1)
10I:>1.E-6   "SCAL(2)
11I:>10.      "SCAL(3)
12I:>1    "IMIN
13I:>1    "LOOP
14I:>1    "NPRI(1)
15I:>1    "NPRI(2)
16I:>1    "NPRI(3)
17I:>-1   "NPLOT
18I:>40   "NP
19I:>0    "IT
20I:>2    "ISYS
21I:>0    "MEAS
22I:>1    "ISAMP
23I:>10.  "TSAMP
24I:>1    "NPRED1
25I:>1    "NPRED2
26I:>2    "NX
27I:>1    "NU
28I:>1    "NY
29I:>1    "NOMAT(1)
30I:>1    "NOMAT(2)
31I:>0    "NOMAT(3)
32I:>1    "NOMAT(4)
33I:>0    "NOMAT(5)
34I:>1    "NOMAT(6)
35I:>2    "IACC
36I:>0.   "HH
37I:>4    "NPLOTC
38I:>12   "NOL
39I:>10.  "TS
40I:>0.00001     "EPSI
41I:>500  "NKAL
42I:>1    "ICHK
43I:>0    "ICR
44I:>  0.0   147.15
45I:>  0.0   147.13
46I:>  0.0   147.15
47I:>  0.0   147.24
48I:>  0.0   147.35
49I:> 10.0   147.41
50I:> 10.0   147.46
51I:> 10.0   147.24
52I:> 10.0   146.67
53I:>-10.0   145.96
54I:>-10.0   144.87
55I:>-10.0   143.96
56I:>-10.0   143.55
57I:>-10.0   143.50
58I:>-10.0   143.81
```

```
59I:> 10.0  144.38
60I:> 10.0  145.17
61I:> 10.0  145.77
62I:> 10.0  145.92
63I:>-10.0  145.66
64I:>-10.0  145.20
65I:>-10.0  144.82
66I:>-10.0  144.84
67I:>-10.0  145.31
68I:>-10.0  146.10
69I:> 10.0  147.17
70I:> 10.0  148.43
71I:> 10.0  149.44
72I:> 10.0  150.05
73I:> 10.0  150.29
74I:>  0.0  150.21
75I:>  0.0  149.79
76I:>  0.0  149.55
77I:>  0.0  149.37
78I:> 10.0  149.26
79I:> 10.0  149.15
80I:> 10.0  148.82
81I:> 10.0  148.16
82I:>-10.0  147.28
83I:>-10.0  146.29
84I:>
EDIT
83:>EXI
LINES:83 FIELDATA
>@SUSPEND
SUSPENDED
>@PRT,S TEST.DAT1
>@RESUME,P ,PR
SENT BY CK TO PR
>@PRT,T TEST.
FURPUR 27R1 RL72-8 07/19/78 15:38:48

CK*TEST(1)
FOR    SYST(1)
REL    SYST
ELT    LMAP1(1)
ABS    LABS1
ELT    DAT1(0)
>@ASG,T PLOT$.
READY
>@SUSPEND
SUSPENDED
>@XQT TEST.LABS1
>@ADD,P TEST.DAT1
>@RESUME,P ,PR
SENT BY CK TO PR
```

```
>@FIN



RUNID CK      ACCT 209976       PROJ CK            PRIO M&M
THE FOLLOWING PLOT IS ON THE FILE PLOT$0000010
DEBIT$PLOTTMINUT$PQR
DEBIT$PLOTTMINUT$PQR
DEBIT$PLOTTMINUT$PQR
DEBIT$PLOTTMINUT$PQR
DEBIT$PLOTTMINUT$PQR
THE FOLLOWING PLOT IS ON THE FILE PLOT$0000020
DEBIT$PLOTTMINUT$PQR
DEBIT$PLOTTMINUT$PQR
DEBIT$PLOTTMINUT$PQR
START 14:47:37 JUL 18,1978      FIN 15:41:39 JUL 18,1978
  PRIS KR   TIME: TOTAL:00:02:56.561  CPU:00:00:20.474  MEM:00:00:51.802
    47.18         CC/ER:00:01:48.713  I/O:00:00:47.373 WAIT:00:48:32.418
*TERMINAL INACTIVE*
>@@TERM
```

APPENDIX H - EXAMPLES OF PRINTS AND PLOTS

The complete output from test example 1 (see Section 6 and Appendix F and G) is shown in this appendix. The prints and plots are discussed in Section 7.

```
LISPID
******



FADD/P TEST.DAT1

 SIZE OF /DATA/ :      314



 VECTOR PAR
   -5.3999999        18000.000        .78999998-03


 VECTOR ITH
   1    2    3


 VECTOR TH
   -.54000000-01     .18000000-01     .78999999-02


 VECTOR SCAL
    .10000000-01     .10000000-05    10.000000


 NPAR                      3
 NTH                       3
 TS            10.000000
 EPSI          .10000000-04
 NKAL                    500

 MATRIX R1
    .78999998-03     .00000000
    .00000000        .00000000

 MATRIX R1D
    .78956125-02     .39478062-01
    .39478062-01     .26322363

 MATRIX R2D
    .10000000-01

 MATRIX P0
    1.0000000        .00000000
    .00000000        1.0000000

 MATRIX P
    .10548699-01     .67190321-01
    .67190321-01     .56199184

 MATRIX C*P*CT+R2D
    .57199324

 MATRIX KT
    .11740197        2.1568632
```

PRINT FROM LISPID
****

INITIAL PARAMETER VECTOR TH
-.54000000-01 .18000000-01 .78999999-02

```
NTH                          3
NP                          40
IT                           0
ISYS                         2
MEAS                         0
ISAMP                        1
TSAMP        10.000000
NPRED1                       1
NPRED2                       1
NX                           2
NU                           1
NY                           1
NOMAT          1  1  0  1  0  1

LOOP                         1
NPRI                   1  1  1
NPLOT                       -1
ITRAN                        0
NMAX                       500
EPST         .10000000-05
EPSK         .10000000-08
CHISQ        6.0000000
IMIN                         1
IND                          8
IF                           0
IPLOT                        1
ITEXT                        1
IIY                          1
SX           16.000000
SY           6.0000000
IDH                          1
HH           .00000000
IACC                         2
NPLOTC                       4
NOL                         12

IABSIS                       2
IM2                          0
ICOS                         1
ML                           1
NOM            1  1  0  0  0  0
NOTRAN         1  0  0  0  0  0
```

```
INITIAL SYSTEM
*************

THE CONTINUOUS SYSTEM
*********************

MATRIX A
  -.55555556-04     .00000000
   1.0000000        .00000000

MATRIX B
  -.30000000-03
   .00000000

THE DISCRETE SYSTEM
*******************

MATRIX A
   .99944460        .00000000
   9.9972228        1.0000000

MATRIX B
  -.29991668-02
  -.14997222-01

MATRIX C
   .00000000        1.0000000

MATRIX K
   .11740197
   2.1568632

INITIAL STATE VECTOR X0
   .00000000        147.15000


INITIAL LOSS
VLOSS          .16585392-01
VLOS1          .16585392-01
NVLOS1                   40
```

```
PRINT FROM NUFLET
*****************

ITERATION NO.       0
NO. OF CALLS OF FUNCT     4
FUNCTION VALUE      .16585392-01
PARAMETER VECTOR
  -.54000000-01     .18000000-01     .78999999-02
GRADIENT VECTOR
   .13819430-01     .11404604+00     .20593870-02

ITERATION NO.       1
NO. OF CALLS OF FUNCT    17
FUNCTION VALUE      .16585392-01
PARAMETER VECTOR
  -.54000000-01     .18000000-01     .78999999-02
GRADIENT VECTOR
  -.49418303-03    -.16330625-01     .90582761-03

ITERATION NO.       2
NO. OF CALLS OF FUNCT    24
FUNCTION VALUE      .16585385-01
PARAMETER VECTOR
  -.53999938-01     .18002041-01     .78998867-02
GRADIENT VECTOR
  -.49418303-03    -.16330625-01     .90582761-03

IEXIT = 0
```

```
PRINT FROM LISPID
*****************


FINAL LOSS
VLOSS          .16585385-01
VLOS1          .16585385-01
NVLOS1                   40
VLOSDT        6.4824262

FINAL PARAMETER VECTOR TH
  -.53999938-01     .18002041-01     .78998867-02

ESTIMATED STANDARD DEVIATION STDEV
  -1.0000000       -1.0000000        .19611432-01

GRADIENT VECTOR G
  -.32619573-03    -.10932369-02     .11151618-02

SECOND DERIVATIVE MATRIX V2
   28.486364        86.167517       -2.2642004
   86.167517        257.97076       -7.0122381
  -2.2642004       -7.0122381        2.3261333

INVERSE OF SECOND DERIVATIVE MATRIX V2INV
  -3.3534345        1.1234465        .12253226
   1.1234465       -.37214755       -.28321358-01
   .12253226       -.28321358-01     .46379177
```

```
EIGENVALUES OF V2
   286.91363          2.1374953          -.26787387

EIGENVECTORS (COLUMNWISE) OF V2
   .31640877          .36480502-01        .94792124
   .94826977          .15100992-01       -.31710627
  -.25882747-01       .99922027          -.29815272-01

COVARIANCE MATRIX OF RESIDUALS
   .16585385-01

NORMALIZED COVARIANCE MATRIX OF RESIDUALS COV
   1.0000000

COVARIANCE MATRIX OF MODEL ERRORS
   6.4824262

NORMALIZED COVARIANCE MATRIX OF MODEL ERRORS COVDT
   1.0000000
```

```
FINAL SYSTEM
************

THE CONTINUOUS SYSTEM
*********************

MATRIX A
  -.55549257-04     .00000000
   1.0000000        .00000000

MATRIX B
  -.29996564-03
   .00000000

THE DISCRETE SYSTEM
*******************

MATRIX A
   .99944466        .00000000
   9.9972231        1.0000000

MATRIX B
  -.29988234-02
  -.14995505-01

MATRIX C
   .00000000        1.0000000

MATRIX K
   .11740186
   2.1568618

INITIAL STATE VECTOR XO
   .00000000        147.15000
```

PRINT FROM LISPID
****

TIME	U 1
.00	.00000000
10.00	.00000000
20.00	.00000000
30.00	.00000000
40.00	.00000000
50.00	10.000000
60.00	10.000000
70.00	10.000000
80.00	10.000000
90.00	-10.000000
100.00	-10.000000
110.00	-10.000000
120.00	-10.000000
130.00	-10.000000
140.00	-10.000000
150.00	10.000000
160.00	10.000000
170.00	10.000000
180.00	10.000000
190.00	-10.000000
200.00	-10.000000
210.00	-10.000000
220.00	-10.000000
230.00	-10.000000
240.00	-10.000000
250.00	10.000000
260.00	10.000000
270.00	10.000000
280.00	10.000000
290.00	10.000000
300.00	.00000000
310.00	.00000000
320.00	.00000000
330.00	.00000000
340.00	10.000000
350.00	10.000000
360.00	10.000000
370.00	10.000000
380.00	-10.000000
390.00	-10.000000

TIME	Y 1	YMOD 1	ERRMOD 1	EPS 1	IEPS
.00	147.15000	147.15000	.00000000	.00000000	0
10.00	147.13000	147.15000	-.20000458-01	-.20000458-01	0
20.00	147.15000	147.15000	.00000000	.43136597-01	0
30.00	147.24000	147.15000	.90000153-01	.63571930-01	0
40.00	147.35000	147.15000	.19999886	.92887878-02	0
50.00	147.41000	147.15000	.26000023	-.52507401-01	0
60.00	147.46000	147.00005	.45995331	.14808464	0
70.00	147.24000	146.55029	.68970680	.75187683-02	0
80.00	146.67000	145.80091	.86908913	-.54014206-01	0
90.00	145.96000	144.75206	1.2079411	.16793823	0
100.00	144.87000	143.70382	1.1661835	-.40591240	0
110.00	143.96000	142.95604	1.0039616	-.59524536-01	0
120.00	143.55000	142.50856	1.0414429	.21607399	0
130.00	143.50000	142.36121	1.1387939	.26918411-01	0
140.00	143.81000	142.51382	1.2961807	.52078247-01	0
150.00	144.38000	142.96623	1.4137707	-.48377991-01	0
160.00	145.17000	143.41836	1.7516365	.22704506	0
170.00	145.77000	143.57036	2.1996365	.75428009-01	0
180.00	145.92000	143.42240	2.4976025	-.16562653	0
190.00	145.66000	142.97463	2.6853695	-.85277557-01	0
200.00	145.20000	142.52714	2.6728592	-.18385696	0
210.00	144.82000	142.37978	2.4402199	-.18970490	0
220.00	144.84000	142.53238	2.3076172	.13298035	0
230.00	145.31000	142.98478	2.3252182	.13251686	0
240.00	146.10000	143.73681	2.3631859	-.28076172-02	0
250.00	147.17000	144.78831	2.3816910	-.21318436-01	0
260.00	148.43000	145.83919	2.5908051	.19402504	0
270.00	149.44000	146.58961	2.8503857	.20414352-01	0
280.00	150.05000	147.03973	3.0102654	-.10616112	0
290.00	150.29000	147.18972	3.1002769	-.53426743-01	0
300.00	150.21000	147.03975	3.1702518	-.97198486-02	0
310.00	149.79000	146.73993	3.0500717	-.18765259	0
320.00	149.55000	146.44027	3.1097260	.20947647	0
330.00	149.37000	146.14079	3.2292137	.30170441-01	0
340.00	149.26000	145.84146	3.4185333	.61491013-01	0
350.00	149.15000	145.39236	3.7576427	.13968658	0
360.00	148.82000	144.64362	4.1763840	.56758881-01	0
370.00	148.16000	143.59541	4.5645924	-.41677475-01	0
380.00	147.28000	142.24790	5.0320988	.85081100-01	0
390.00	146.29000	140.90117	5.3888340	-.12317657	0

PRINT FROM RESLIS
****

SAMPLES= 40

TIME LAGS= 12

NPLOTC= 4

INSIGNALS INCLUDED: 1

RESIDUALS INCLUDED: 1

5 % TOL. LIMITS OF SIGN CHANGES: 12 25

DEGREES OF FREEDOM OF CHI2-TEST OF INDEPENDENCE= 5

DEGREES OF FREEDOM OF CHI2-TEST OF NORMALITY= 17

5% SIGMA LIMIT OF CORRELATIONS: .310

AUTOCORRELATION FUNCTIONS

TAU	EPS 1
0	1.000
1	-.066
2	-.176
3	.106
4	-.235
5	-.190
6	-.112
7	-.118
8	.014
9	.076
10	.458
11	.113
12	-.181

CROSS CORRELATIONS BETWEEN INPUT 1 AND RESIDUALS

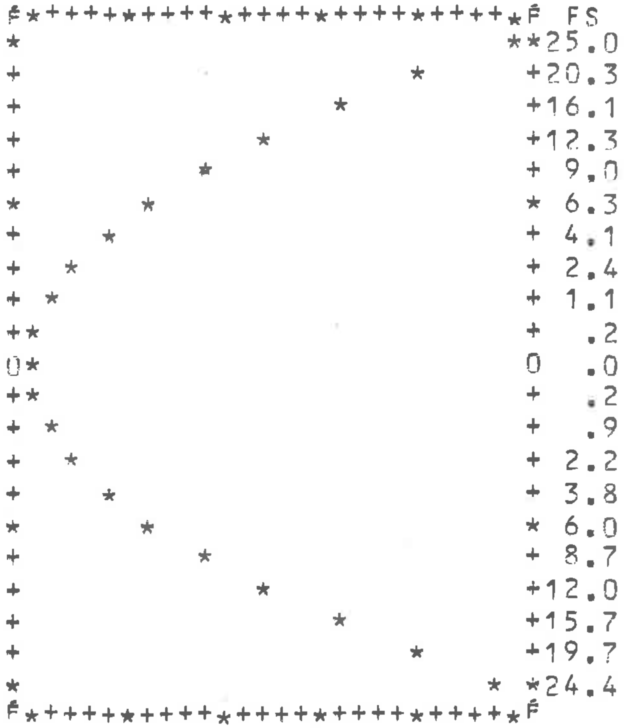
TAU	EPS 1
-12	.136
-11	.142
-10	.064
-9	.098
-8	-.268
-7	-.404
-6	-.257
-5	-.151
-4	.127
-3	.257
-2	.376
-1	.264
0	.134
1	.235
2	-.373
3	-.494
4	-.191
5	-.101
6	.154
7	.317
8	.294
9	.278

10	.092
11	-.011
12	-.258

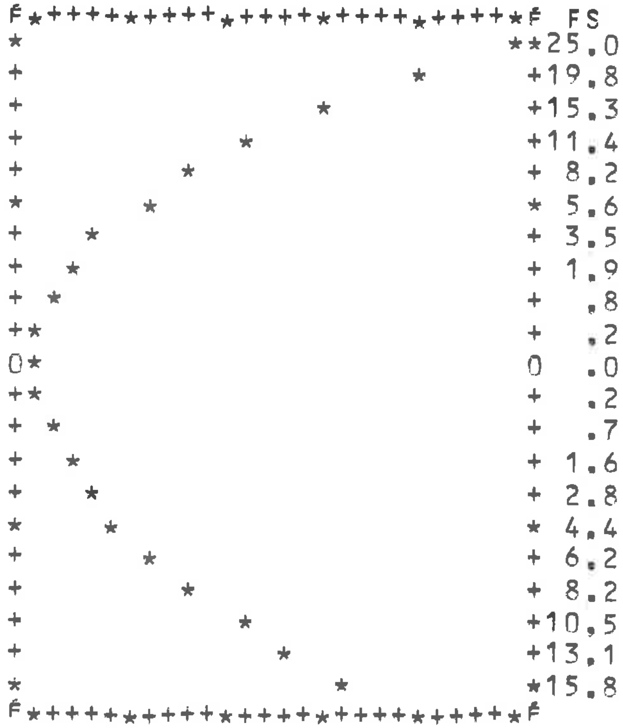
TESTQUANTITIES OF RESIDUALS

	EPS 1
MEAN VALUE	.007
STANDARD DEVIATION	.130
NUMBER OF SIGN CHANGES	18
MAX. ABS(RES(T))	3.117
% RES OUTSIDE .26% LIMITS	2.500
CHI2-TEST OF INDEPENDENCE	5.515
CHI2-TEST OF NORMALITY	14.778

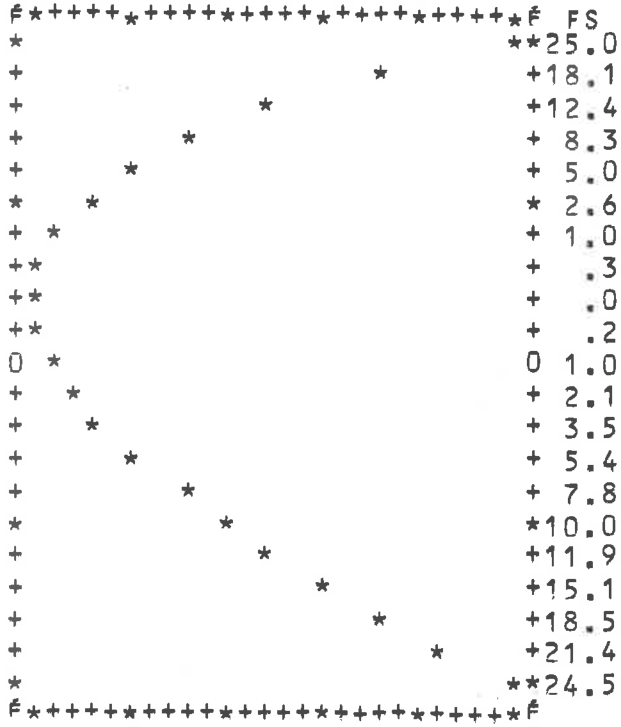
```
VECTOR PAR
  -5.3999938         18002.041         .78998867-03


VECTOR ITH
  1    2    3


VECTOR TH
  -.53999938-01      .18002041-01      .78998867-02


VECTOR SCAL
   .10000000-01      .10000000-05      10.000000


NPAR                           3
NTH                            3
TS              10.000000
EPSI            .10000000-04
NKAL                         500

MATRIX R1
   .78998867-03      .00000000
   .00000000         .00000000

MATRIX R1D
   .78954999-02      .39477499-01
   .39477499-01      .26321987

MATRIX R2D
   .10000000-01

MATRIX P0
   1.0000000         .00000000
   .00000000         1.0000000

MATRIX P
   .10548555-01      .67189435-01
   .67189435-01      .56198484

MATRIX C*P*CT+R2D
   .57198612

MATRIX KT
   .11740186         2.1568618
```



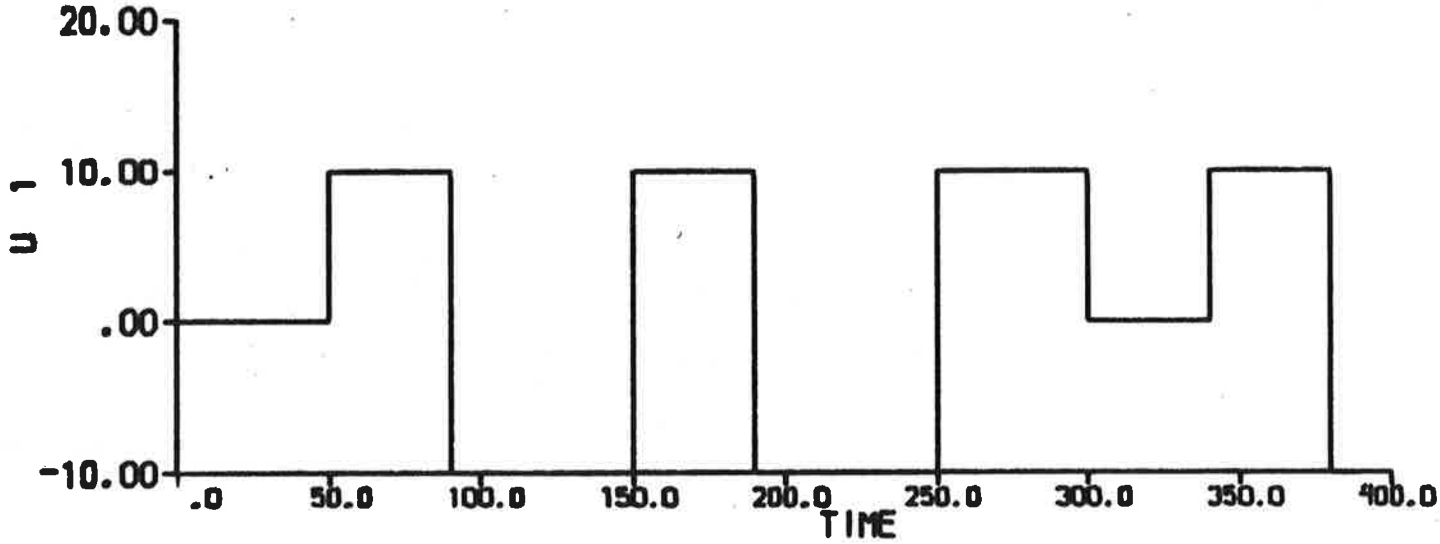
FV 1= .166-01 + .232-05*FS
X-STEP=0.2*HH*XM= .2000-03
X 1= -.53999938-01
F(X)= .16585385-01
G1= .13036-01 G2= -.33015-03
G4= -.32858-03

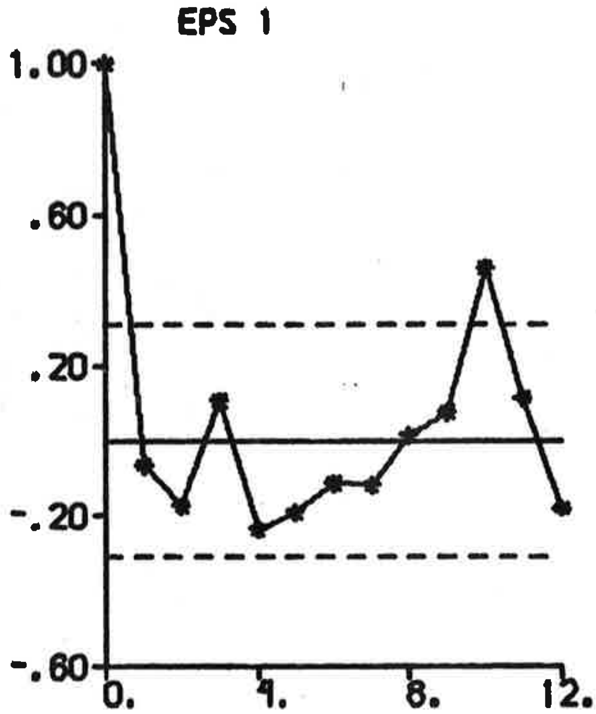


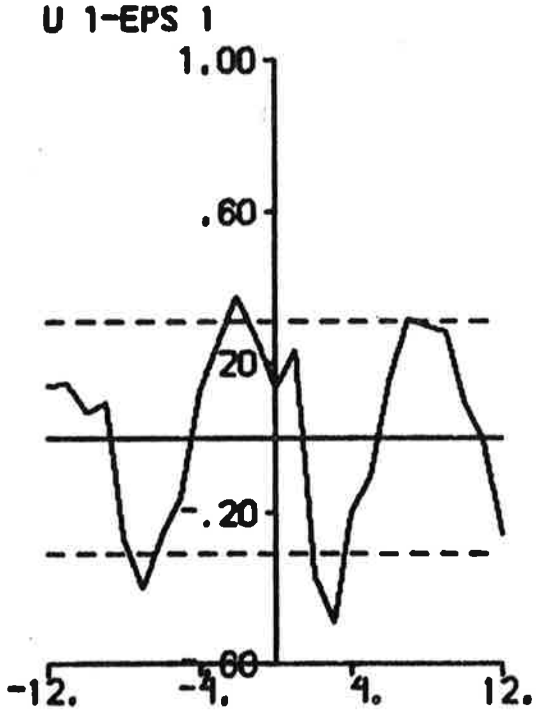
FV 2= .166-01 + .263-04*FS
X-STEP=0.2*HH*XM= .2000-03
X 2= .18002041-01
F(X)= .16585385-01
G1= .11434+00 G2= -.15897-01
G4= -.11161-02



FV 3= .166-01 + .226-06*FS
X-STEP=0.2*HH*XM= .2000-03
X 3= .78998867-02
F(X)= .16585385-01
G1= .20370-02 G2= .84331-03
G4= .11332-02


20.00
10.00
.00
-10.00
U 1
.0
50.0
100.0
150.0
200.0
250.0
300.0
350.0
400.0
TIME


EPS 1
1.00
.60
.20
-.20
-.60
0.
4.
8.
12.


U 1-EPS 1
1.00
.60
.20
-.20
-.60
-12.
-4.
4.
12.