How to Run IDPAC on Lund University Computing Centre

A Guide for Complete Idiots

Gutman, Per-Olof; Schönthal, Tomas; Wieslander, Johan

1980

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

CODEN: LUTFD2/(TFRT-7207)/1-042/(1980)

HOW TO RUN IDPAC ON LUND UNIVERSITY COMPUTING CENTRE

A GUIDE FOR COMPLETE IDIOTS

PER-OLOF GUTMAN
TOMAS SCHÖNTHAL
JOHAN WIESLANDER

DEPARTMENT OF AUTOMATIC CONTROL
LUND INSTITUTE OF TECHNOLOGY
AUGUST 1980

HOW TO RUN IDPAC
ON
LUND UNIVERSITY COMPUTING CENTRE

A guide for complete idiots written by and for

Per-Olof Gutman
Tomas Schönthal
Johan Wieslander

Department of Automatic Control
Lund Institute of Technology

August 11, 1980

Revised:

Organization
LUND INSTITUTE OF TECHNOLOGY
Department of Automatic Control
Box 725
S-220 07 LUND 7 SWEDEN

Document name
Internal report

Date of issue
August 1980

CODEN: LUTFD2/(TFRT-7207)/1-042/(1980)

Author(s)
Per-Olof Gutman, Tomas Schönthal, and Johan Wieslander

Sponsoring organization

Title and subtitle

How to run IDPAC on Lund Data Center.
A guide for complete idiots.

A4 A5

Abstract

A beginner's manual how to run the interactive identification software package IDPAC on Lund Data Center. The manual contains instructions how to start up a computer run, how to set certain IDPAC parameters, the so called global variables, and a wealth of practical advice, including hints about the macro facility. Also included is a detailed example.

Key words A4 A5

Command oriented, Data analysis, Spectral analysis, Identification, Parameter modelbuilding, Least Squares method, Maximum Likelihood method, Model analysis, Simulation.

Classification system and/or index terms (if any)

Supplementary bibliographical information

Language
English

ISSN and key title

ISBN

Recipient's notes

Number of pages
42

Price

Security classification

CONTENTS

PREFACE

****
* *
* SUMMARY OF IDPAC COMMANDS: see Appendix B. *
* *
* SUMMARY OF CLOSE DOWN, START UP, INITIALIZATION, AND *
* PAPER PLOTTING: see Appendix C. *
* *
****

This is the fourth volume in the series "A guide for complete idiots" edited by and for Per-Olof Gutman.

The first volume is called MANUAL: HOW TO GET THE PDP-15 STARTED, RUNNING, AND CLOSED DOWN and was written by Per-Olof Gutman. The second volume, which has no title, deals with the handling of LSI-11 and the use of the editor PAGED, a PASCAL-compiler, etc. This volume was expertly written by Leif Andersson, and is actually not for complete idiots, but could be used by us if we do not ask too many questions. The third volume HUR MAN HITTAR PROGRAMFEL (Swedish for HOW TO FIND PROGRAMMING ERRORS) is based on the bitter experiences of Matz Lennels'.

Coming volumes will include guides to the program packages SIMNON, MODPAC, SYNPAC, and POLPAC, and possibly a manual how to run the VAX.

This booklet is organized in the following manner:

The first chapter is called HOW TO CLOSE DOWN... The original idea to start with instructions how to close down, emanates from our deeply rooted conviction about the human nature: The person eager to start up will certainly take his time to find the appropriate chapter, while the tired human being, fed up with everything, is more likely to just turn the power off if he does not immediately find the advice he is looking for.

HOW TO START UP follows in the second chapter. The first and second chapters are dependent on the physical facilities, and as they change these chapters might become obsolete. Therefore the reader is advised to check that he holds the latest revised edition in his hand, and that the information is up-to-date.

Chapter 3 contains HOW TO INITIALIZE IDPAC... How successful you will be with IDPAC largely depends on how well you follow the points here.

IDPAC ON LUCC
PREFACE

Chapter 4 tells you how to plot your figures on the paper plotter Tektronix 4662 without wrinkling your paper or your face.

Chapter 5 and 6 are very valuable indeed: They contain some useful HINTS AND TRICKS and some common MISTAKES. Some people argue that these few pages are the best in this booklet.

The most valuable information is contained in chapter 7: The REFERENCES. Note that it is impossible to read this guide without having [2] available for quick reference.

Appendix A contains a simple, constructed EXAMPLE. If this is your first contact with IDPAC, we urge you to go through this example at the computer. Although the example gives some hints on identification, it is not a course.

Finally Appendix B contains a short summary of all IDPAC commands, and Appendix C a useful summary of chapters 1, 2, 3, and 4.

We wish you GOOD LUCK with IDPAC. Please feel free to contact the authors to discuss improvements of this guide, of IDPAC, or of the world. We will try our best.

Eva Dagnegård helped to type some of the text, and Britt-Marie Carlsson prepared some of the figures in their usual expertise way. We thank them sincerely for their assistance.

IDPAC ON LUCC
CLOSE DOWN

HOW TO CLOSE DOWN WITHOUT GETTING SCREWED

1. You would certainly like to save some of the IDPAC-files that you have created during this run. For this purpose use your mass storage file FNAME. Type:

   SAVFIL FNAME FIL1 FIL2 FIL3 ...

   where FNAME and FILi are defined in the chapter HOW TO INITIALIZE YOUR IDPAC RUN WITH ELEGANCE, point 1.

   Example: SAVFIL FNAME DATA-D TEXT SYSTEM-A

2. Press the SHIFT- and F1-keys simultaneously.

3. Type STOP

   If you have generated printer outputs during your IDPAC-run, for instance by using the commands LIST(LP) or SWITCH LOG ON then the following question will appear on the screen:

   Print file to be saved, deleted or printed?(S/D/P)>

   Answer S Your print file will be saved on a file whose name PRINTFILENAME will appear on the screen. To get the print out on your local printer SMSK36 (make sure its power is on and that the button AUTO is lit up) type

   @SYM PRINTFILENAME,,SMSK36

   The printed list will appear a few minutes later.

   If you answer P a list with your user identity (PEO) as a heading in giant letters will become available in a pigeon hole at Lund University Computing Centre. The pigeon holes are organized alphabetically according to the first letter of user identity. It takes about 2 daytime hours for your list to appear in it's pigeon hole.

   If you want to print part of your printer output, please contact Tomas or Tommy.

4. Type the following commands:

   @FIN
   @@TERM

5. Turn off the terminal power.
   Turn off the plotter power.
   Turn off line printer power.

IDPAC ON LUCC
CLOSE DOWN

Put the hat on the plotter pen.
Make sure that you leave the place neat.
Turn off the lights and lock the door if your are last.

HOW TO START UP WHEN USING THE TERMINAL TEKTRONIX 4025

Introduction

This start up instruction is valid if you use a direct link to connect your Tektronix 4025 terminal to the Univac-computer of Lund University Computing Centre.

Warning_1: Do not touch any other keys, buttons, or knobs than those described here. Not following this urgent advice might have catastrophic consequences. Especially, you must avoid the Master Reset button, which does not mean that you are the master of the set!

Warning_2: The exclamation mark key (!) is forbidden except when explicitly stated below.

We assume that you would like to run your program under the following conditions:

a. You use a Tektronix 4025 terminal, here called the terminal.

b. You use a Tektronix 4662 Interactive Digital Plotter, here called the plotter.

c. You use an OKI DP-125 printer, here called the line printer.

d. You use a direct link between the terminal room and the UNIVAC computer of Lund Data Central.

e. The input data files which you are going to give an IDPAC-treatment are already present on mass storage (suitably formatted), or you are going to create the data files manually from the keyboard.

Start up of the terminal

1. Make sure that the power mains are connected to the terminal and the plotter.

2. Find the "DATA C DIRECT" female connector between the two rightmost windows of the terminal room. Connect it to the "MODEM" male plug (or an extension thereof) of the plotter.

3. Make sure that the "TERMINAL" connection of the plotter is connected to the terminal.

4. Set the switches of the plotter correctly. Find them at the back of the plotter where they are marked A, B, C, D. Set A = 0, B = 2, C = A, D = 3 (a distinct position between 2 and 4).

5. Terminal power on. Find the switch on the right side.

6. Wait until the cursor appears on the screen. Then press the key TTY LOCK.

   Make sure that the key COMMAND LOCK OUT is released, i.e. that its red light is off.

```
*************************************************************
*                                                           *
* All commands are terminated by pressing the RETURN-key.   *
* In the sequel, commands are given without this info.      *
*                                                           *
*************************************************************
```

7. Check the terminal status: Type !SYS
   Check if TB=1200 and RB=1200. If not, type !BAU 1200
   Check if EC=R. If not, type !ECH R
   Check if PA=E. If not, type !PAR E
   Check again: type !SYS

   Note: TB and RB denote the transmission and reception baud rates, respectively. EC denotes echo; R means remote echo, L means local echo. PA stands for parity; E means even.

8. Press the RETURN-key. On the screen appears:

```
LOGON AS SLDC44
ENTER USERID/PASSWORD:
>
```

```
*************************************************************
*                                                           *
* To correct an incorrectly written character, press the    *
* key BK SPC , which moves the cursor one step to the       *
* left and enables you to retype the last character.        *
*                                                           *
* To erase the whole line, press the key RUB OUT.           *
*                                                           *
*************************************************************
```

9. Type U/P

   where

   U = user identity, e.g. your initials (maybe PEO), maximum 6 characters;
   P = your secret password (ask Tomas how to get one).

IDPAC ON LUCC
START UP

On the screen will appear:

U/
>

10. Currently the following command is necessary. It will become redundant in the future:

@RUN PEO,999999,PUST

where PEO = run identity, e.g. your initials (max 6 characters); 999999 = account number (ask Tomas how to get one); PUST = project identity, preferrably the same name a the run identity, in order to minimize the number of mnemonics ou have to remember (max 6 characters).

The computer answers:

*UNIVAC 1100 OPERATING SYSTEM ...
LAST RUN 14:47 WEDNESDAY 26 DEC 79
NOW IS 11:54 THURSDAY 27 DEC 79
>

11. SKIP THIS POINT IF YOU ALREADY HAVE A MASS STORAGE FILE.

Create a file to save your data in:

@CAT,P FNAME

where FNAME = a file name chosen by you.

The computer answers:

READY
>

12. Assign your mass storage file to this run:

@ASG,AX FNAME

The answer is:

READY
>

13. SKIP THIS POINT IF YOU HAVE ALREADY REGISTERED YOUR MASS STORAGE FILE IN A PREVIOUS RUN.

Register your file:

@REGISTER FNAME./60

On the screen appears an explanation what REGISTER actually means.

14. Start IDPAC:

    @ADD ID*IDPAC.START

    You are now inside IDPAC. Wait until all trash has been printed on the screen and the > appears. Please read the contents of the screen. Before doing point 15 you can scroll the contents of the screen with the filled arrows on the keys 7 and 1 of the numeric pad.

    (Moses was the first guy to scroll. He wrote the Torah [Pentateuch] on scrolls.)

```
**************************************************************
*                                                            *
* To move the cursor on the screen, use the arrows on the    *
* keys 4, 8, 6, 2 of the numeric pad.                        *
*                                                            *
**************************************************************
```

15. Initialize the graphic mode by pressing the key F1. Press the RETURN-key.

    The screen has now been divided into 2 areas: one upper plot area, and one lower (and smaller) command area. The IDPAC commands are entered in the command area, while the outputs from IDPAC, including the plots, will appear in the plot area.

    Note: There exists a facility of graphic scrolling which is not meant for complete idiots. If you wish to push yourself up to a higher graphic stage, please contact Tomas.

```
**************************************************************
*                                                            *
* To scroll the contents of the command area, use the        *
* filled arrows on the keys 7 and 1 of the numeric pad.      *
*                                                            *
**************************************************************
```

Start up of the plotter and the line printer

Note that points 1 - 4 above pertain to the plotter, too.

16. Check that the pen works. Make sure that the pen is correctly located in its holder and that the hat is off. Put down the pen-arm in plotting position.

17. Plotter power on.

18. Insert paper: The LOAD-button must be in down position. Place a paper at the lower aluminum edge and the left indicator. Straighten the paper. Release the

LOAD-button.

19. Line printer power on. Find the switch on the top of the printer. The line printer is presently located in the Hilbert-room.

20. Top set the form, if necessary. This is done by lifting the lid, advancing the paper one step at a time by pressing the FEED-button intermittently until the top of the next form is barely visible. Then turn on the switch TOF SET.

21. Make sure that the button AUTO is lit up.

22. Before you start to run your IDPAC program proper, certain initializing commands are necessary. These are presented in the chapter HOW TO INITIALIZE IDPAC WITH ELEGANCE. Please turn to that chapter.

HOW TO INITIALIZE IDPAC WITH ELEGANCE

Introduction

Certain initializing actions are common to all IDPAC runs. These include getting IDPAC-files from your mass storage file, and setting certain global variables. This chapter covers these commands, as well as a suggestion of a macro that will considerably facilitate the initialization.

In this chapter, values in brackets, ( ), denote default values. A global variable has its default value, unless you assign another value to it in a LET-command.

If you are satisfied with all the default values you may skip points 3 - 9 below.

Actions

1. Type SWITCH LOG ON

   From now on, all your correct commands (exceptions, see the command SWITCH in [4]) will be logged into a file, and you will be able to retrieve the log as a printed output at the end of your run. See the chapter HOW TO CLOSE DOWN ..., point 3.

2. Almost certainly you would like to get IDPAC-files from your mass storage file FNAME. Type

   GETFIL FNAME FIL1 FIL2 FIL3 ...

   where

   FNAME = your mass storage file as defined in points 11, 12, 13 of the chapter HOW TO START UP...
   FILi = IDPAC-files, as defined in [2] (see e.g. the command LIST).

   FILi must be followed by -D when it is a data file, and -A when it is an aggregate file.

   Example: GETFIL FNAME DATA-D TEXT SYSTEM-A

3. Type LET TICK. = T

   where

   T = the time unit, in seconds, of the clock which was used e.g. during data logging. (1).

4. Type LET DELTA. = D

   where

   D = the sampling period of your data in number of seconds, with the constraint that D/T must be an integer, where T is defined in the command LET TICK. = T in point 3 just above. (1).

5. Type TURN TIME X

   where

   X = OFF, S, M or H. OFF stands for sample number, S for seconds, M for minutes, and H for hours.

   This command defines the scale of the horizontal axis of the plots, whether it will be in sample number, seconds, minutes, or hours. (OFF).

   Please refer to the command TURN in [2].

6. Type LET NPLX. = P

   where

   P = the length of the horizontal axis of the plots, in the unit you chose in point 5 above. (100).

   NPLX. is used in the command PLOT. Please see [2].

7. Type LET SCALES. = I

   where

   I = a flag that can be either 0 or 1, controlling the scale marks of the vertical and horizontal scale in plots generated by the command PLOT. (1).

   Please refer to the command PLOT in [2].

8. Type

   LET WMIN. = A
   LET WMAX. = B

   where

   A = the lower limit for angular frequency, in rad/s. (0.01).
   B = the upper limit for angular frequency, in rad/s. (100).

WMIN. and WMAX. are used in the commands ASPEC, CSPEC, SPTRF, etc. Please see these commands in [2].

Before setting WMIN. and WMAX. you should consider which angular frequency range is of interest in your problem.

9. Type LET NOF. = N

where

N = the number of frequency points to be computed by the commands ASPEC, CSPEC, SPTRF, etc. (100).

Please see [2].

10. Type HCOPY ON

This command enables paper plotting of the plot information that appears on the screen. (Notice that default is OFF).

11. Type WRITE

A list of the values of all global variables will appear on the screen.

Change of values of global variables

Of course, any of the above commands can be performed anytime during your IDPAC run. If you find that the value of a global variable is not wisely chosen, do not hesitate to change it.

Other global variables

Note that there are other global variables (i.e. PRINT.) that are used for certain commands, like ML, RESID, etc. Read about the command you want to use in [2], and set the appropriate global variables before you use the command. Of course some of these might be included in your initialization scheme.

How to use a macro

When you intend to make several IDPAC runs on the same set of data, the above described global variables will in general have identical values for every run. To ensure reproducibility throughout the processing of a set of measurement data, it is especially important that TICK. keeps its value. To speed up the execution of the initializing commands you can use a macro. A macro can be

described as a sort of subroutine, or a predefined set of commands that will be executed together.

a. The first time you run IDPAC on a set of data, go through points 1-11 above. During the run, change, if necessary, the values of the global variables so that you have a suitable set of initializing commands. List the values on the screen and on the lineprinter by using the command WRITE (see [2]).

   Before you conclude the run, create your macro. Here, an example of a macro is given. Type

```
MACRO INITIAL
SWITCH LOG ON
LET TICK. = 0.050
LET DELTA. = 10
TURN TIME S
LET NPLX. = 50
LET SCALES. = 1
LET WMIN. = 0.05
LET WMAX. = 5
LET NOF. = 20
HCOPY ON
END
```

   Now, test the macro, by running it. Type

```
INITIAL
```

   Finally, save the macro on your mass storage file. Type

```
SAVFIL FNAME INITIAL
```

b. The next time you use IDPAC you initialize simply by typing the following

```
GETFIL FNAME INITIAL DATA-D TEXT SYSTEM-A
INITIAL
```

   It is wise to keep the GETFIL of DATA-D, TEXT, etc outside the macro, because the files you would like to get might change from run to run.

c. To change a macro, you can either retype it according to point a above, or use the editor, see the command EDIT in [2].

   To get into the editor in our example, type

```
EDIT INITIAL
```

You will find more about macros in the chapter HINTS AND TRICKS.

HOW TO PAPER PLOT WITHOUT WRINKLING YOUR NOSE

How to remove a plotted paper and insert a fresh one

1. Press down the LOAD-button.
2. Remove old paper. Place a new paper at the lower aluminum edge and the left indicator arrow.
3. Release the LOAD-button.

How to paper plot

4. Issue the command HCOPY ON , if you have not done it before.
5. Fill the screen with plot information, e.g. by issuing the command PLOT.
6. Print HCOPY After a short while the pen starts to draw the picture of the screen on the paper.

```
**************************************************************
*                                                            *
* Sometimes the pen stops drawing before the picture is      *
* finished. Most often the pause is very short, and the      *
* pen resumes its actim. Now and then, however, the stop     *
* is permanent, although the picture is not completed. Why   *
* this happens is unknown. Try to press the RETURN-key       *
* once; this helps sometimes. If not, please call Tomas or   *
* Tommy.                                                     *
**************************************************************
```

7. If you have no more paper plots to make in the remainder of your run, you may disable paper plotting (this saves some money). Type HCOPY OFF

HINTS AND TRICKS

In this chapter certain useful hints, tricks, and shortcuts are presented. The list is short; you should complete it yourself. Inform the authors of this guide about your inventions, so that the next edition will be improved. Also tell about hints you would like to have included, but which you have not been able to invent yourself.

1. You will soon find that certain commands often follow in sequence, like ML-RESID-DETER, or PLOT-HCOPY, etc. Use the macro facility to construct your own commands. You will find useful ideas in [6].

   Example: Assume that you would like to perform the command sequence ML-RESID-DETER with some plots inbetween for models of various order on data contained in different data files. Write the following macro:

```
MACRO MLID I FNAME NU " MLID = the name of the macro.
                      " I = model order,
                      " FNAME = data file name,
                      " NU = number of inputs,
                      " FNAME(NU+1) contains the output.
LET V=FNAME+I "Note: string concatenation, not addition.
              "See [4].
ML SYST(V)<FNAME I
RESID RSYST<SYST(V) FNAME
SUSPEND " enables you to turn pages on the screen
        " at your own pace. See [4].
"At this point you are assumed to be outside RESID.
DETER YDSYST<SYST(V) FNAME(1 - NU)
LET NUP1=NU+1 "Note: addition, not string concatenation.
VECOP DIFF<FNAME(NUP1)-YDSYST
PLOT FNAME(NUP1)/DIFF "Plant output & model difference
SUSPEND
END
```

   When you use the macro for an ML-identification of order 3 on the data file DATA that contains 4 inputs as its first four columns and the output as the fifth column, you type:

```
MLID 3 DATA 4
```

   Notice that the identified model is contained in the file SYST(DATA3).

2. Assume you have written a macro called TRIAL. You would like to test it. Type

```
SWITCH TRACE ON
SWITCH ECHO ON
SWITCH DATE OFF
TRIAL
```

The last command will start the execution of the macro. The previous ones will cause an echo on the screen of the commands that TRIAL consists of. Now you can follow how the macro proceeds, step by step.

If you would like to interrupt the macro, see point 5 below. What to do if there is an error in the macro is described in point 8 below.

Don't forget to SWITCH TRACE OFF, ECHO OFF, and DATE ON before you resume your regular operation.

3. Every command can be supplied with a comment. Do as follows:

   Type the command. Continue typing a quotation mark (") and then the comment on the same line. Then press the RETURN-key.

   Example: PLOT DATA " INPUT (1), SYSTEM OUTPUT (2)

   The comment will appear on the log (if you have made SWITCH LOG ON). The comment will also appear on the paper plot if you do HCOPY after a commented PLOT-command. And that is very useful.

   For another way to write a comment on a paper plot, see the command HCOPY T in [2].

4. Sometimes an iteration (i.e. ML) simply takes too long time, and you would like to interrupt it. Type @@X C

   On the screen will appear > , and you can continue as usual.

   However, the result of e.g. the ML-iteration so far is saved in the system file you specified in the ML-command. This system file you can use as usual, for instance to get starting values for a new ML-command. See the command ML and its subcommand INVAL in [2].

5. Maybe you would like to interrupt a macro. Type @@X C

   On the screen will appear > and you can continue as usual. (Warning: see point 4 in the chapter SOME COMMON MISTAKES...)

   You have also the chance to continue the macro at the point it was interrupted. Type RESUME

6. Sometimes the command PLOT, LIST, or WRITE produces too much or unwanted output on the plot area of the screen. You would like to interrupt the output. Distinctly press the key BREAK twice.

   The answer is: **OUTPUT INTERRUPT**

   Now you have three possibilities:

   a. You have changed your mind and would actually like to see the entire output. Type @@CONT and the output continues from the point of interruption.

   b. You would like to skip n (0<n<64) lines of the output, and resume it after that. Type @@SKIP n

   c. The remainder of the output is indeed rubbish, and you don't want to see a line more of it. Type @@X O (that last character is the letter O).

   After completion of the action, > will appear on the screen. You can continue as usual. (Warning: see point 4 in the chapter SOME COMMON MISTAKES...)

   Note that all output (e.g. coming from one or more commands inside a macro) to be produced before the next > is affected by your @@-command.

7. When you have several similar commands following one another, you can use the "comma"-feature to repeat some command words and decrease the amount of typing. This is best illustrated by two examples:

```
Example_1: >LIST(LP)(T) ML1 " comment 1
           >  , ,, ,,,, ML2 " comment 2

Example_2: >ML MODEL2 < DATA 2 " comment 1
           > , MODEL3 <  ,   3 " comment 2
```

   Note that the sign < must be typed explicitly.

   In the log, all commands will be printed in full.

8. An error might occur during the execution of a macro. The macro might have been incorrectly written, or it might not work under the current conditions, maybe depending on certain parameters, etc. In any case the execution will be interrupted, and an indication will be given on the screen about what is wrong. What you can do in this situation is illustrated in the following example. You have written this (erroneous) macro:

```
MACRO TRIAL
INSI U 100
NORM
X
PLAT U
END
```

When the macro is run, the following takes place on the screen:

```
>TRIAL "This is the command to start the macro
   <PLAT U
    INVALID COMMAND
  >
```

Now you have two possibilities to continue:

a. You can interrupt the further execution of the macro. Type END

   This will bring you to the main command level, and you can continue with any command. For instance you may type EDIT TRIAL , in order to correct your macro permanently.

b. It is possible to temporarily correct the macro and resume its execution. Type

```
PLOT U
RESUME
```

   Note that the macro is not corrected permanently; next time you run it the same error will occur. To correct it permanently you must issue an EDIT-command from the main command level.

9. Two almost equal time series, Y1 and Y2, are best displayed by forming their difference DY, and plotting one of them and DY in seperate subdiagrams:

```
>VECOP DY<Y1-Y2
>PLOT Y1/DY
```

   This procedure you can find in the chapter AN EXAMPLE.

10. The global variables returned by the commands STAT, ML, RESID, LS, etc can be used in various ingenious ways, e.g. for "smart" macros. Please see Tomas.

11. To create a system you can use the editor. A smarter way that saves you work is to run a dummy ML-identification of the desired order on arbitrary data files. In the editor you can change the coefficients of the polynomials of the resulting system to the desired ones. This method is used in the chapter AN EXAMPLE.

SOME COMMON MISTAKES YOU ARE PRONE TO MAKE

Here follows a list of some beginner's mistakes. It is by no means exhaustive, therefore empty space is provided so that you can add your own. Please inform the authors about newly discovered mistakes, so that they can be included in the next edition of this guide.

1. To print the letter O instead of the digit 0.

2. You perform a PLOT-command, whereby a plot appears on the screen. Then you do HCOPY, in order to get a paper plot. However you have forgotten the command HCOPY ON to enable paper plotting. So you type HCOPY ON. When this command is executed, the plot disappears from the screen. You are forced to redo the original PLOT-command.

   The moral is: Perform HCOPY ON before you type PLOT, for instance at the initialization.

3. To forget to set global variables that govern the execution, printing of results, etc of certain commands like ML, RESID, etc.

4. You try to issue a main level command when you are actually on a sublevel. As a matter of fact it is easy to see that you are on a sublevel: the prompting character > appears indented to the right.

   You will find yourself on a sublevel after certain main level commands which include subcommands, like INSI etc.; when you have interrupted a nested macro at a subordinated level; etc.

   You must leave the sublevel and return to the main level before you issue a main level command. Proceed as follows:

   i. If you are inside a macro, an END will take you out of the macro nest.

   ii. If you are inside a subcommand, an X or KILL will take you to the main command level.

   If you are in doubt where you are, type WRITE and the relevant information will appear on the screen.

REFERENCES

[1] J. Wieslander: Interactive Programs - General Guide. Report TFRT 3156.

[2] J. Wieslander: Idpac Commands - User's Guide. Report TFRT 3157.

[3] J. Wieslander: Interaction in Computer Aided Analysis and Design of Control Systems. Report TFRT 1019.

[4] J. Wieslander, H. Elmqvist: INTRAC - Languange Manual. Report TFRT 3149

[5] I. Gustavsson, A-B Nilsson: övningar for Idpac (in Swedish). Report TFRT 7169.

[6] I. Gustavsson: Några Macros for Idpac (in Swedish). Report TFRT 7170.

All these reports are published by the Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

AN EXAMPLE HOW TO MAKE YOU HAPPY

Introduction

The following three main fields of usage for IDPAC may be discerned:

1. Data manipulation and data analysis.

2. Correlation analysis and frequency analysis.

3. LS- and ML-identification.

The example in this chapter touches upon these three uses. The example is a constructed one, i.e. a dynamical system is defined, output data is generated with this system and corrupted with noise, and finally an attempt is made to ML-identify the system behind the data. All communication with IDPAC is presented, including the plots. It is easy for you, the reader, to reproduce this example on your own.

More thorough examples for each command can be found in [2]. Actually it is advisable to scan through [2] before using IDPAC. A good introductory example can be found in [3]. Reference [5] contains systematic exercises, and [6] some useful macros.

The example

```
                §
PEO/SECRET
@CAT,P FILFIL
@ASG,AX FILFIL
@REGISTER FILFIL./60
@ADD IDPAC.START
```

"We start up IDPAC according to the chapter HOW TO START UP ..., whereby we assign the mass storage file FILFIL

```
>SWITCH LOG ON
>TURN TIME S
```

"This is all initialization we do because there is nothing to get from FILFIL.

```
>WRITE
>WRITE(LP)
```

"List the values of the global variables on the screen and on the line printer. We are satisfied with these values and do no LET-commands. See the chapter HOW TO INITIALIZE ..

```
18  GLOBAL VARIABLE(S), THE 18  FIRST ARE RESERVED

NPLX    .    = 100          NOF     .    = 100
INIML   .    = 0            PRIML   .    = 0
LIML    .    = 0            TIML    .    = 20
IFP     .    = 1            NU      .    = 9
PRINT   .    = 0            YMIN    .    = 0.
YMAX    .    = 0.           AMP     .    = 1.
DELTA   .    = 1.           UMIN    .    = 1.E-2
WMAX    .    = 100.         SCALES  .    = 1
FTEST   .    = 0            TICK    .    = 1.
```

§ See point 9-10 in the chapter HOW TO START UP.

```
>HCOPY ON                          "Enable paper plotting
>INSI DUM1 100
>RAMP
>X                                 "Create a dummy data file: a ramp
                                   in DUM1(1).
>VECOP DUM1(2)<DUM1(1)*DUM1(1)     "Square the ramp and put the result
                                   in DUM1(2).
>ML SYST<DUM1 2                    "Generate the structure of the second
                                   order system SYST by performing a
                                   dummy ML-identification on the DUM1
                                   data.
>LIST(FF)(T)SYST
>LIST(LP)(T)SYST                   "List the text file SYST on the screen
                                   and on the line printer.
```

```
BEGIN

"ML SYST<DUM1 2 "GENERATE THE STRUCTURE OF THE SECOND ORDER SYSTEM SYST
"80.01.21 - 14:34:06
"
DISCRETE MISO TRANSFER FUNCTION

SAMPLE INTERVAL 1. S

LAMBDA 0.100001 +- 7.07113E-3

LOSS FUNCTION 0.500009

AIC -164.728

APOLYNOMIAL
1. QÜ-0 + 14.5672 QÜ-1 - 15.5672 QÜ-2

BPOLYNOMIAL
 QÜ-1 * (18.5673 QÜ-0 + 14.5671 QÜ-1 )

CPOLYNOMIAL
1. QÜ-0 - 2.08555E-4 QÜ-1 - 1.94609E-4 QÜ-2

END
```

"The coefficients of the polynomials of SYST are just trash, and the will be changed later in the editor.

```
>ML NOISE<DUM1 1                   "Create the structure for the first
                                   order model for the noise, NOISE.
>LIST(FF)(T)NOISE
>LIST(LP)(T)NOISE                  "List NOISE
```

```
"ML NOISE<DUM1 1 "CREATE THE STRUCTURE OF NOISE
"80.01.21 - 14:38:37
"
DISCRETE MISO TRANSFER FUNCTION

SAMPLE INTERVAL 1. S

LAMBDA 0.206337 +- 1.45902E-2

LOSS FUNCTION 2.12874

AIC -25.8615

APOLYNOMIAL
1. QÜ-0 - 0.99966 QÜ-1

BPOLYNOMIAL
 QÜ-1 * (2.04036 QÜ-0 )

CPOLYNOMIAL
1. QÜ-0 + 0.815361 QÜ-1

END
```

```
>EDIT SYST                          "Edit SYST
   >DIS ON                          "Display on, i.e. follow your work
                                    on the screen.
   >L APOLY                         "Locate the string POLY
   >N                               "Go to the next line
   >R  1-1.5Q^-1+0.7Q^-2            "Replace this line with the polynomial
                                    1-1.5q^-1+0.7q^-2
   >N 3                             "Go down another three lines
   >R  Q^-1+0.5Q^-2
   >N 3
   >R  1.0-0.8Q^-1+0.2Q^-2
   >T                               "Go to the top of the file
   >B                               "Go to the bottom of the file, i.e.
                                    display the whole file on the way.
                                    Check that it is correct
   >E                               "Exit from the editor
>LIST(LP)(T)SYST                    "List the new system SYST on the line
                                    printer. SYST now is the system
                                    Ay(t) = Bu(t) + λCe(t), where A, B,
                                    λ , and C are given in the system
                                    description:
```

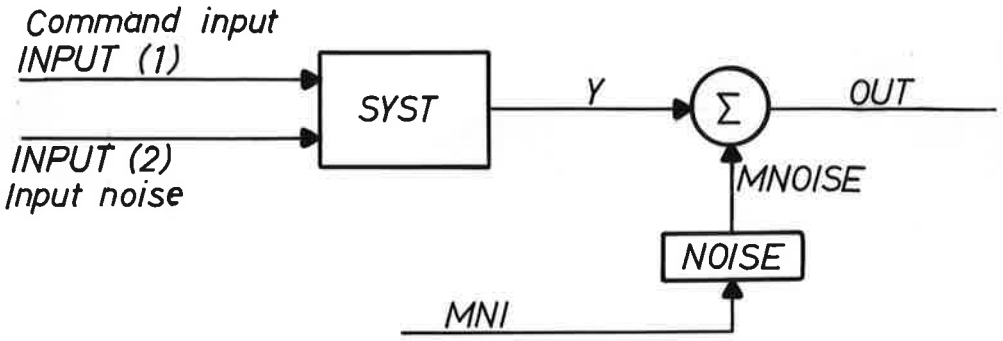
```
BEGIN

"ML SYST<DUM1 2 "GENERATE THE STRUCTURE OF THE SECOND ORDER SYSTEM SYST
"80.01.21 - 14:34:06
"
DISCRETE MISO TRANSFER FUNCTION

SAMPLE INTERVAL 1. S

LAMBDA 0.100001 +- 7.07113E-3

LOSS FUNCTION 0.500009

AIC -164.728

APOLYNOMIAL
1 - 1.5QÜ-1 + 0.7QÜ-2

BPOLYNOMIAL
QÜ-1 + 0.5QÜ-2

CPOLYNOMIAL
1. - 0.81QÜ-1 + 0.2QÜ-2

END
```

```
EDIT NOISE

    (subcommands)

LIST(LP)(T)NOISE
```

"Edit NOISE. Change the polynomials in the same way as above to yield: $Ay(t) = Bu(t)$. Notice that it is chosen to be a pure integrator.

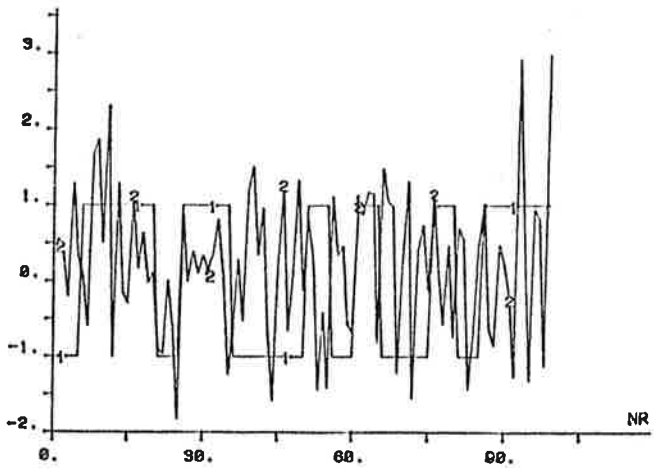
```
BEGIN

"ML NOISE<DUM1 1 "CREATE THE STRUCTURE OF NOISE
"80.01.21 - 14:38:37
"
DISCRETE MISO TRANSFER FUNCTION

SAMPLE INTERVAL 1. S

LAMBDA 0.206337 +- 1.45902E-2

LOSS FUNCTION 2.12874

AIC -25.8615

APOLYNOMIAL
1. QÜ-0 - 1. QÜ-1

BPOLYNOMIAL
QÜ-1

CPOLYNOMIAL
0

END
```

"We intend to create a system output according to this figure:




```
>INSI INPUT 100                    "Create the command input into the
                                   first column of the file INPUT, i.e.
                                   INPUT(1)
>PRBS 5                            "Make it a Pseudo Random Binary Signal
                                   with basic period = 5
>X
>INSI INPUT(2) 100                 "Create the input noise into INPUT(2)
>NORM                              "Make it Normal(0,1)
>X
>PLOT INPUT                        "Plot the command input and input noise
>HCOPY                             "Paper plot
```

C(88)INPUT 'PLOT THE COMMAND INPUT (1) AND INPUT NOISE (2) OF THE FILE INPUT
88.01.22 - 14:04:28



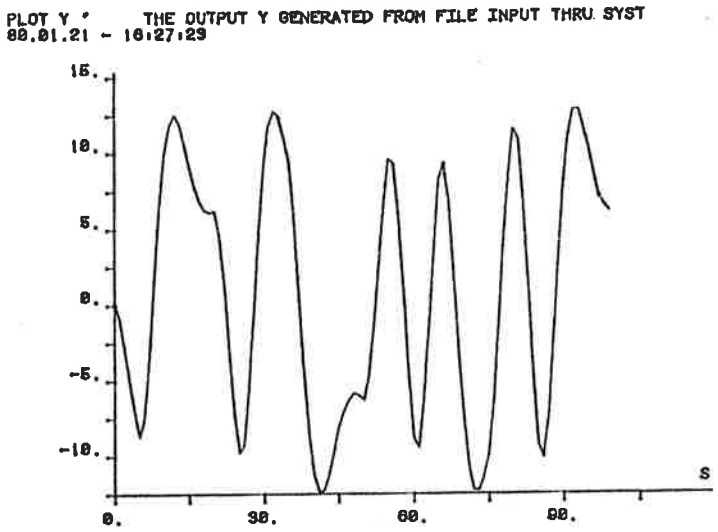
```
>DSIM Y<SYST INPUT
```

"Simulate the system SYST with INPUT(1) as the input u(t), INPUT(2) as the input e(t), and Y as the output y(t)

```
 PLOT Y
```

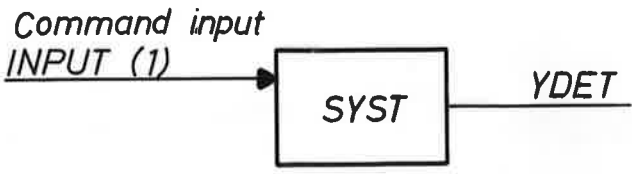
"THE OUTPUT Y GENERATED FROM FILE INPUT THRU SYST

```
 HCOPY
```




```
>DETER YDET<SYST INPUT(1)
```

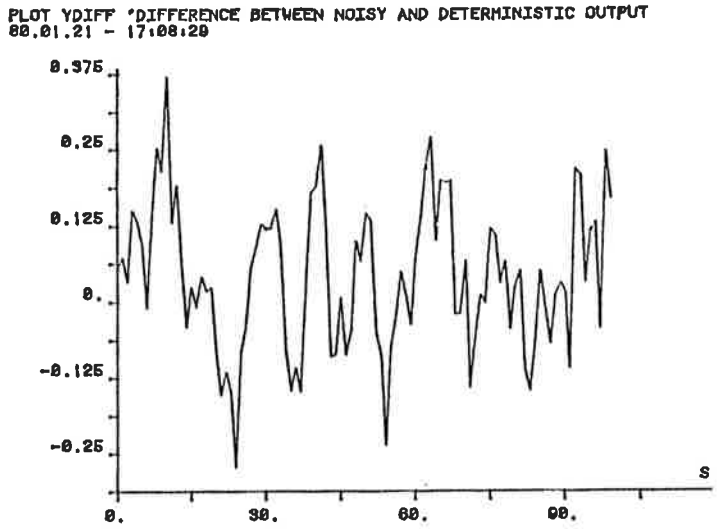
"For comparison's sake, the deterministic output YDET from SYST with INPUT(1) as control input, and no noise, is simulated:



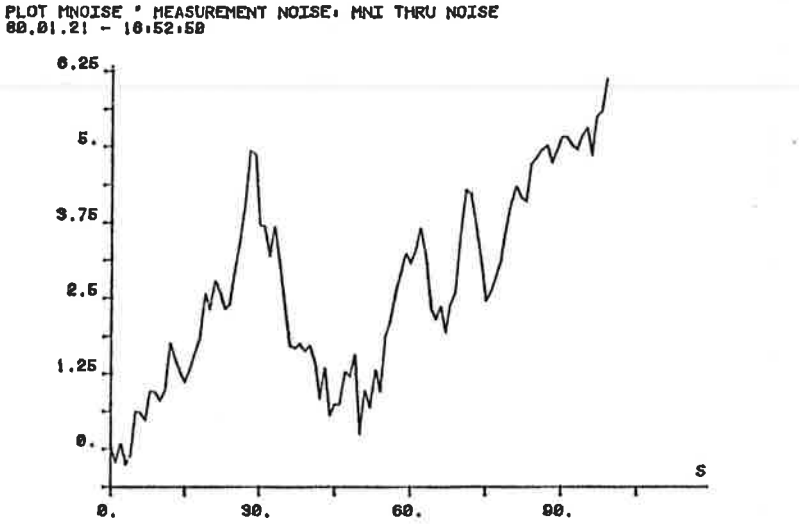

```
>VECOP YDIFF<Y-YDET
```

"Form the difference between Y (= the output from SYST corrupted by system noise) and YDET (=the deterministic output from SYST)

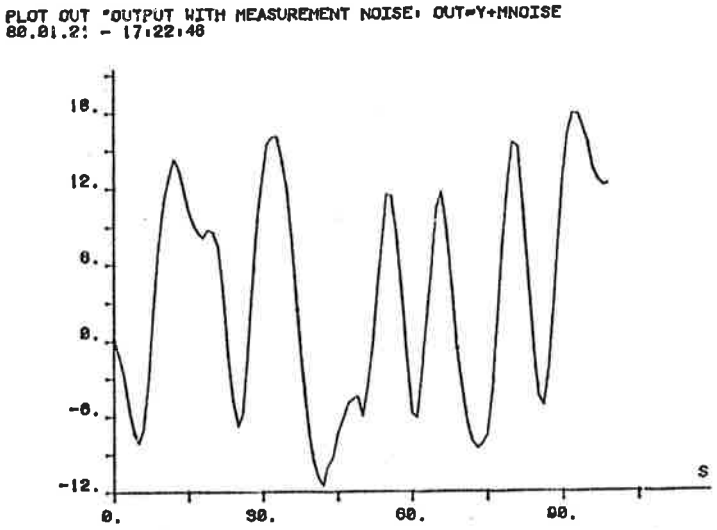
```
>PLOT YDIFF
>HCOPY
```




```
>INSI MNI                          "Create a random signal MNI that will
                                    serve as input into NOISE
>NORM 0.0 0.5                      "Make it Normal(0,0.5)
>X
 DETER MNOISE<NOISE MNI            "Compute the deterministic output =
                                    = MNOISE from the system NOISE with
                                    MNI as the input

>PLOT MNOISE
>HCOPY                             "Paper plot MNOISE
```




```
>VECOP OUT<Y+MNOISE                "Create the total system output,
                                    corrupted by measurement noise

>PLOT OUT   "comment
>HCOPY
```



"Now the final part of the data generation is at hand: to create a file (WORK) whose first column contains the command input, and whose second column contains the measured output

```
>MOVE    WORK(1)<   INPUT(1)
>MOVE    WORK(2)<   OUT
>SAVFIL FILFIL WORK-D
```

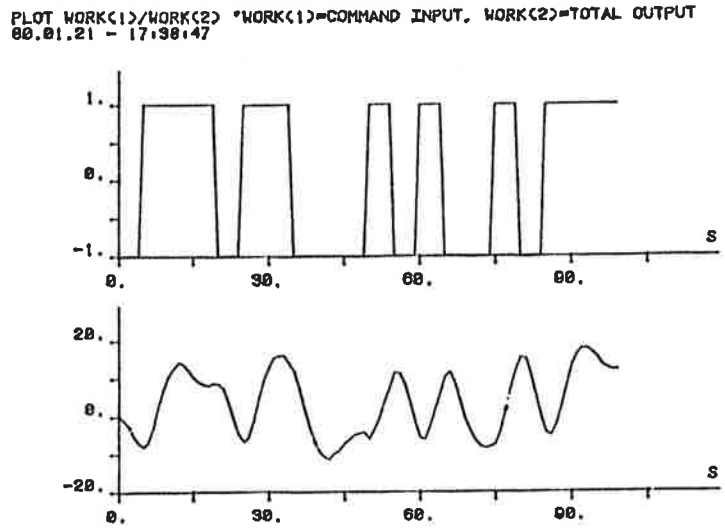
"WORK(1):=INPUT(1) = command input
"WORK(2):=OUT = 'measured' output
"Save the file WORK
"In a real identification run, it is the file WORK you will have got from your identification experiment. We assume that this is the case here, i.e. the identification run starts here, (initialization done).

```
>GETFIL FILFIL WORK-D
```

"Get the file WORK which contains the experiment data: WORK(1) contains the command input to the unknown system and WORK(2) contains the output measurements.

```
>PLOT WORK(1)/WORK(2)
>HCOPY
```

"Plot the contents of WORK, for instance in a split screen diagram



```
>FHEAD WORK
   >LOOK
   >X
```

"Look at the head of the file WORK (Unfortunately there is no printout here from this command, but there is a lot to be seen on the screen.)

```
>STAT WORK(2)
```

"Check the statistical properties of the output:

```
>STAT WORK(2) "STATISTICAL PROPERTIES OF THE OUTPUT
>80.01.21 - 17:47:40
SUM       =    331.341
MEAN      =    3.31341
VARIANCE  =    72.3562
ST.DEV.   =    8.50625
MINIMUM   =   -11.4800
MAXIMUM   =    17.8435
LENGTH    =     100
```

"We notice that the mean is non-zero. To avoid initial value-complications in the identification routines we wish to remove the mean:

```
>TREND TEMPO<WORK(2)  0
```

"Remove a 0:th order trend, i.e. a constant, from WORK(2) and put the result in TEMPO:

```
>TREND TEMPO<WORK(2) 0 "REMOVE ZEROTH ORDER TREND, I.E. A CONSTANT
>80.01.21 - 17:52:53

ESTIMATED PARAMETERS FOR TREND CORRECTION
OF WORK     ( 2) FROM POINT NO     1 TO POINT NO   100

C0    3.22311
>PLOT TEMPO
>STAT TEMPO "STATISTICAL PROPERTIES OF DETRENDED OUTPUT
>80.01.21 - 18:01:24
SUM       =    9.03027
MEAN      =   9.030274-002
VARIANCE  =    72.3562
ST.DEV.   =    8.50625
MINIMUM   =   -14.7031
MAXIMUM   =    14.6203
LENGTH    =     100
```

"Check the detrended output by plotting it and looking at the statistical properties.

```
>MOVE    DATA(1)<   WORK(1)
>MOVE    DATA(2)<   TEMPO
>SAVFIL FILFIL DATA-D
```

"We are satisfied with the detrended output, and move the detrended input-output data into the file DATA, which we save.

"A more experienced 'identifier' might have discerned or checked for a higher order trend. However, to detrend WORK with a first order trend, and then do ML-identification, is left to you, the reader, as an exercise. When you will get a better result than the one presented here you will rightly feel proud!

```
                                    "We are now ready for the ML-identification.
                                    First we set some global variables that
                                    govern the ML-command and its printout,
                                    please see  the command ML in [2].
>LET INIML. = 0                     " We do not wish an estimate of initial values.
>LET PRIML. = 1                     "We wish some line printer printouts, see [2].
>LET ITML. = 100                    "Set the maximum number of iterations = 100.
>ML MODEL1<DATA 1                   "Identify an ML-model of order 1 (MODEL1)
                                    based on the data contained in DATA.

>ML MODEL1<DATA 1 "ML MODEL OF ORDER 1
>80.01.21 - 18:08:48

STARTING VALUES GIVE:

LAMBDA            8.50673      +-    .601516
LOSS FUNCTION     3618.22
AIC               717.959

CONVERGENCE (DV/V<  1.0-006)
*****************************

A1               -.860884      +-  3.069322-002
B1                1.89797      +-   .248267
C1                .602807      +-  6.507926-002

LAMBDA            1.59704      +-   .112928
LOSS FUNCTION     127.528
AIC               383.419

GRADIENT OF THE LOSS FUNCTION
    8.37632E-6     4.30876E-6     3.66848E-5

SECOND DERIVATIVE MATRIX
        2904.69       -99.6623       -56.0148
       -99.6623        56.8127        98.5363
       -56.0148        98.5363        778.114

EIGENVALUES OF THE SECOND DERIVATIVE MATRIX
        2909.84         789.11         40.6752

INVERSE OF THE SECOND DERIVATIVE MATRIX
    3.69361E-4     7.71211E-4    -7.10726E-5
    7.71211E-4      2.4166E-2    -3.00474E-3
   -7.10726E-5    -3.00474E-3     1.66055E-3


------------------ ITERATION COUNT:      5 ----

>ML MODEL2<DATA 2                   "ML-model of order 2
                                    Notice that the Akaike test quantity
                                    decreased from 383.419 for MODEL1 to
                                    196.931 for MODEL2:
```

```
>ML MODEL2<DATA 2 "ML MODEL OF ORDER 2
>80.01.21 - 18:09:33

STARTING VALUES GIVE:

LAMBDA            8.50673      +-    .601516
LOSS FUNCTION    3618.22
AIC               723.959

CONVERGENCE (DV/V<  1.0-006)
******************************

A1              -1.46183      +-  2.960333-002
A2               .654664      +-  2.567164-002
B1               1.06147      +-  9.964297-002
B2               .581002      +-   .140936
C1              -.320696      +-   .117355
C2               .299732      +-  8.772228-002

LAMBDA           .610009      +-  4.313413-002
LOSS FUNCTION    18.6055
AIC              196.931

GRADIENT OF THE LOSS FUNCTION
    1.13845E-5     1.7643E-5   -7.41333E-7    -8.3819E-7     1.15857E-6

    5.08502E-7

SECOND DERIVATIVE MATRIX
       8409.66        7727.11       -257.845       -515.306       -32.4159
       7727.11        8331.93       -54.4242       -248.842        5.99361
      -257.845       -54.4242        120.606        95.2614        3.18003
      -515.306       -248.842        95.2614        119.563       -4.7273
      -32.4159        5.99361        3.18003       -4.7273        41.3849
      -48.2675       -19.6744         5.1555       -1.60052        26.3734

      -48.2675
      -19.6744
        5.1555
      -1.60052
       26.3734
       65.4627

EIGENVALUES OF THE SECOND DERIVATIVE MATRIX
       16119.6         745.609          106.69          78.786         25.2247

       12.6958

INVERSE OF THE SECOND DERIVATIVE MATRIX
     2.3551E-3   -1.96195E-3   -2.24554E-3    7.97948E-3    3.02048E-3
   -1.96195E-3    1.77107E-3    1.41372E-3   -5.99634E-3   -2.47529E-3
   -2.24554E-3    1.41372E-3    2.66822E-2   -2.82634E-2   -6.29457E-3
    7.97948E-3   -5.99634E-3   -2.82634E-2    5.33791E-2    1.41763E-2
    3.02048E-3   -2.47529E-3   -6.29457E-3    1.41763E-2    3.70111E-2
    3.01885E-4   -1.75025E-4   -1.48725E-3      1.901E-3   -1.25854E-2

    3.01885E-4
   -1.75025E-4
   -1.48725E-3
      1.901E-3
   -1.25854E-2
    2.06799E-2


------------------ ITERATION COUNT:     5 -----------------
```

```
>ML MODEL3<DATA 3                    "ML-model of order 3.

STARTING VALUES GIVE:

LAMBDA           8.50673      +-    .601516
LOSS FUNCTION    3618.22
AIC              729.959

CONVERGENCE (DV/V<  1.0-006)
*****************************

A1              -.584412
A2              -.625480
A3               .572692
B1               1.04732
B2               1.51327
B3               .527696
C1               .574906
C2              4.393209-002
C3               .205892

LAMBDA           .607222      +-  4.293706-002
LOSS FUNCTION    18.4359
AIC              202.015

GRADIENT OF THE LOSS FUNCTION
     -0.168298       0.272556      -0.132668    -8.54369E-3      -0.167868

     -8.031E-3       -0.16566     6.62795E-2      -0.117796

SECOND DERIVATIVE MATRIX
       2495.47         2254.29         1759.96        -62.9787          -157.1
       2254.29          2473.9         2231.48        -23.7252        -60.7391
       1759.96         2231.48         2449.77         29.1721        -21.3563
      -62.9787        -23.7252         29.1721         60.6526         6.75461
        -157.1        -60.7391        -21.3563         6.75461           60.42
      -216.418        -154.185         -57.656         33.1235         6.45193
      -41.4851         35.0374        -23.5005        -2.40215         1.27321
        12.717        -44.3148         37.7309         8.32279        -2.55596
      -43.8073         19.0608        -33.2028        -4.67118         3.76937

      -216.418        -41.4851          12.717        -43.8073
      -154.185         35.0374        -44.3148         19.0608
       -57.656        -23.5005         37.7309        -33.2028
       33.1235        -2.40215         8.32279        -4.67118
       6.45193         1.27321        -2.55596         3.76937
         60.026        -1.47577        -2.90854        0.547031
      -1.47577         74.7396        -57.6376           54.002
      -2.90854        -57.6376           89.78        -50.3239
      0.547031           54.002        -50.3239         76.8249

EIGENVALUES OF THE SECOND DERIVATIVE MATRIX
       6657.53         751.252         239.794         72.1531         52.6546

       34.0866         22.6511         12.2912        -0.823102

PSEUDO INVERSE WAS USED IN THIS ITERATION

----------------- ITERATION COUNT:    23 -----------------
```

"Notice that the Akaike test quantity increased, and that a pseudo inverse was used. Therefore we decide to settle for the second order model MODEL2. The other quantities of MODEL2 seem all right too: i.e. the standard deviation estimates. the gradient, etc.

```
>LIST(LP)(T)MODEL2
```

"List MODEL2 on the lineprinter. Notice that it is not too far from our original system SYST (page A3).

```
>LIST(LP)(T) MODEL2
>80.01.21 - 18:15:23
 BEGIN

 "ML MODEL2<DATA 2 "ML MODEL OF ORDER 2
 "80.01.21 - 18:09:37
 "
 DISCRETE MISO TRANSFER FUNCTION

 SAMPLE INTERVAL 1. S

 LAMBDA 0.610009 +- 4.31341E-2

 LOSS FUNCTION 18.6055

 AIC 196.931

 APOLYNOMIAL
 1. QÜ-0 - 1.46183 QÜ-1 + 0.654664 QÜ-2

 BPOLYNOMIAL
  QÜ-1 * (1.06147 QÜ-0 + 0.581002 QÜ-1 )

 CPOLYNOMIAL
 1. QÜ-0 - 0.320696 QÜ-1 + 0.299732 QÜ-2

 END
```

```
>SAVFIL FILFIL MODEL1 MODEL2 MODEL3
```

"Save for future use MODEL1, MODEL2, MODEL3

"We would like to test MODEL2 further by performing a test of the residuals using the command RESID

```
>LET PRINT. = 1
```

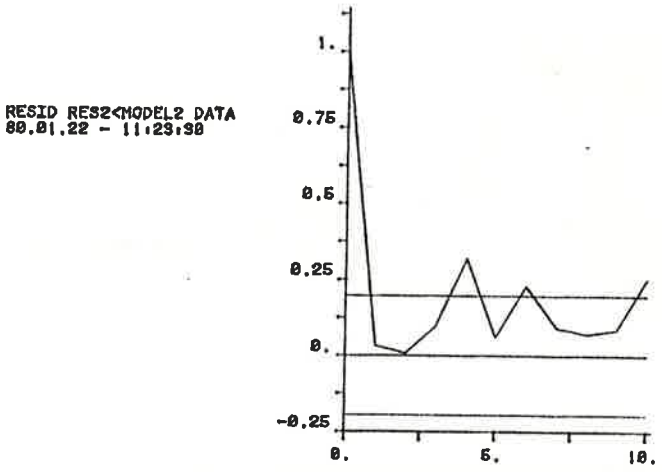
"Get a line printer printout from RESID, etc See [2].

```
>RESID RES2<MODEL2 DATA
```

"Compute the residuals and perform certain tests. Put the residuals in RES2.

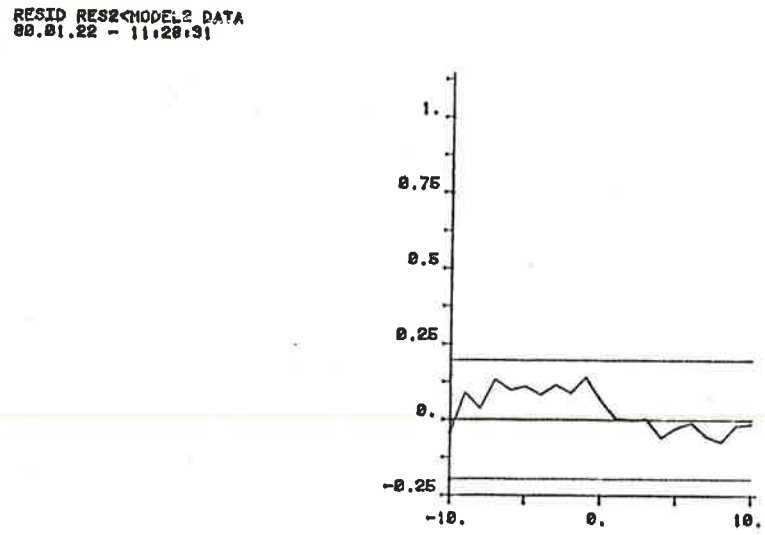
```
     > HCOPY
```

"Paper plot the autocorrelation function of the residuals.



>PAGE "Plot the cross correlation between the residuals and the input on the screen.

>HCOPY "Paper plot the cross correlation.



"We notice that the residuals are not white, and the the residuals are un-correlated with the input within the 95% confidence limit.

>KILL "Finish the RESID subcommand string.
"At the left side of the above two figures some text is produced which is not copied onto the paper plot. However, if the global variable PRINT. = 1, the text appears in the line printer log too:

```
>RESID RES2<MODEL2 DATA
>80.01.22 - 11:23:17


CHI-SQUARE GOODNESS-OF-FIT TEST, LENGTH:      100
**************************************************

MEAN VALUE                      -.110983
STANDARD DEVIATION               .602850

DEGREES OF FREEDOM                    17
TEST QUANTITY                    13.2246
CORRECTION FROM THE TAILS        .859000

SKEWNESS                        -1.42853
KURTOSIS                         5.00318

ABSOLUTE FREQUENCIES
        1          1          0          5          3
        7          7          9         10         14
       17         10          9          4          3
        0          0          0

RELATIVE CUMULATIVE FREQUENCIES
     .010       .020       .020       .070       .100
     .170       .240       .330       .430       .570
     .740       .840       .930       .970      1.000
    1.000      1.000      1.000

VARIANCE OF THE RESIDUALS:
  .359794


NUMBER OF CHANGES OF SIGN
OF THE RESIDUALS:    43

5 PERCENT TOLERANCE LIMITS:
   39    59


TEST OF INDEPENDENCE OF THE
RESIDUALS

E(RES(T)*RES(T+TAU))
FOR:    0 < TAU <    11

TEST QUANTITY:   25.0955
DEGREES OF FREEDOM:    10


TEST OF NORMALITY

TEST QUANTITY:   13.2246
DEGREES OF FREEDOM:    17
```

"See the command RESID in [2] for CHI^2 0.05 level confidence limits. For 17 degrees of freedom this limit is ≈ 20. As 13.2246<20 we do not reject the hypothesis that the residuals are normal.

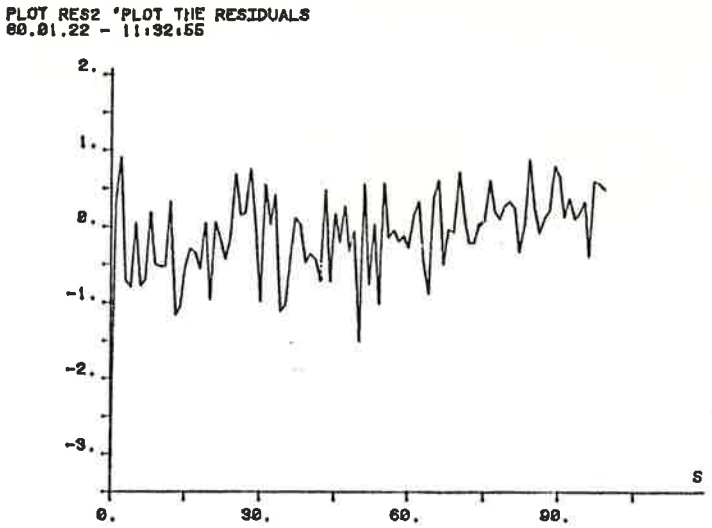
"0.05 confidence limit = 18
The residuals are not independent. This, together with their not whiteness (see two figures back) means that there is more information hidden in the data. Try to find a better model!

continued

```
AUTO  COVARIANCES (1.96*SIGMA LIMIT:    .196000    )
-------------------------------------------------

  1.00000       3.006723-002  5.844327-003  9.664526-002   .317572
 6.041423-002    .228760      9.063861-002  7.051644-002  8.619987-002
  .251447

TEST OF INDEPENDENCE BETWEEN
RESIDUALS AND INPUT:      1

E(RES(T)*U(T+TAU))
FOR:      2 < TAU <     13

TEST QUANTITY:    2.99154
DEGREES OF FREEDOM:     10              "0.05 confidence limit = 18


E(RES(T)*U(T+TAU))
FOR:    -10 < TAU <      1

TEST QUANTITY:    7.50036
DEGREES OF FREEDOM:     10              "0.05 confidence limit = 18
                                        The tests indicate independence


CROSS COVARIANCES (1.96*SIGMA LIMIT:    .196000    )
-------------------------------------------------

-4.376755-002  8.802500-002  3.675604-002   .132083      9.655825-002
  .109746      8.204934-002   .113929      8.860658-002   .140669
 6.343494-002
 4.246012-003 -2.154486-003  2.720057-003 -5.957195-002 -2.868474-002
-9.634783-003 -5.596384-002 -7.493080-002 -1.948130-002 -1.366796-002
```

```
>PLOT RES2
>HCOPY
```

"It is also possible to plot the residuals. We discern a trend, don't we?




```
>DETER YMOD2< MODEL2 DATA(1)
```

"Another quantity of interest is model output, i.e. the output, YMOD2, from MODEL2 when the command input DATA(1)is the input, and no noise is present.

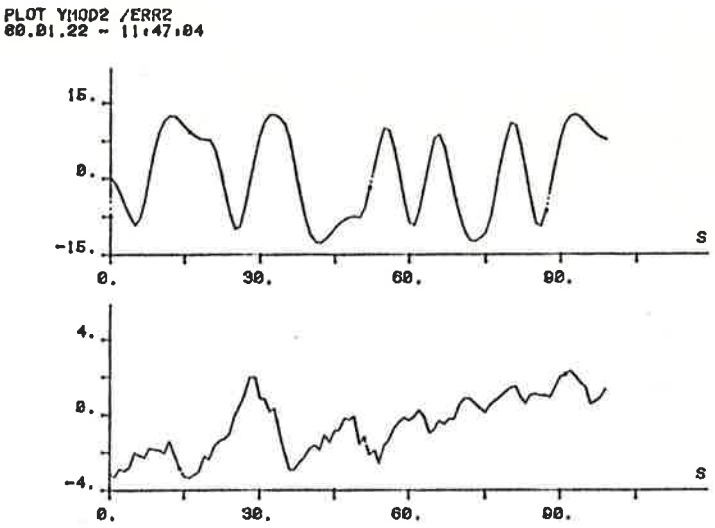
```
>VECOP ERR2<DATA(2)-YMOD2
```

"Compute error between process (measurement) output and model output.

```
>PLOT  YMOD2/ERR2
```

"Plot model output and output error in a split screen diagram.

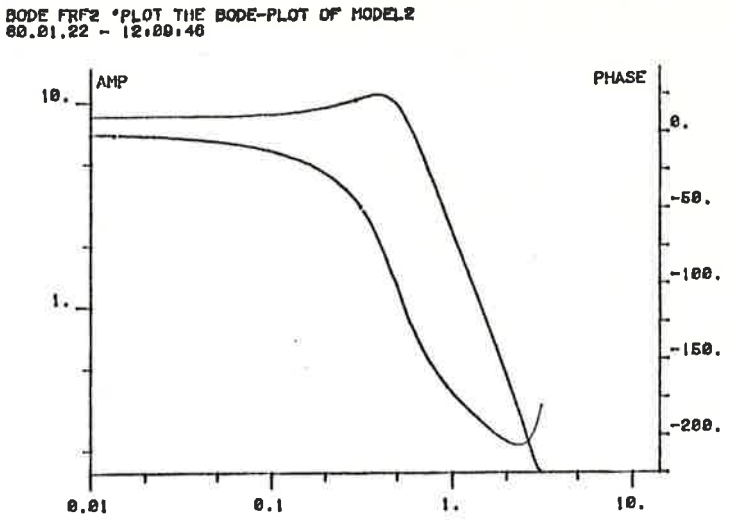
```
> HCOPY
```



"The trend.

```
>SPTRF FRF2<MODEL2 B/A
```
"Finally, compute the frequency responce of the transfer function defined by B/A of the system MODEL2.

```
>BODE FRF2
>HCOPY
```
"Plot the frequency responce in a Bode plot.



"The upward tail of the phase curve is due to the fact that it is a discrete time system.

```
SAVFIL FILFIL FRF2-D
STOP
```
"Save the frequency responce file.
"Stop this IDPAC-run.
"Close down according to the chapter HOW TO CLOSE DOWN ...

Conclusion

This example is not meant to be a course in identification. Neither does it use more than a small fraction of the available IDPAC commands. Its sole purpose is to familiarize the reader with the interactive way IDPAC is used.

We suggest that you create a dummy example of your own, where you test the effect of other commands. Notice that there are sometimes many ways to do the same thing, e.g. to use the ML-command for LS-identification. Use 2 interactively: whenever you want to do a certain thing, look for a command or a sequence of commands than will do it; whenever you don't know what to do, look for ideas.

When you find things you can not do, do not hesitate to inform the authors. Your action will help to improve IDPAC.

SUMMARY OF IDPAC COMMANDS

```
ACOF FNAM1[(C1)] < FNAM2[(C2)] NOL

ASPEC FRF[(F)] < FNAM2[(C2)] NOL

BODE [(SW)] FRF1[(F11 F12 ..)] [FRF2[(F21 .. )] ..
   SUBCOMMANDS:
   PAGE
   KILL
   SW: 'AP'/'A'/'P'/'AO'

CCOF FNAM1[(C1)] < FNAM2(C21 C22) NOL
      OR
CCOF FNAM1[(C1)] < FNAM2[(C2)] FNAM3[(C3)] NOL

CONC [DNAM1] < DNAM2 DNAM3

CONV DNAME < FNAM[(C1..)] NCOLX [TSAMP]

CSPEC FRF[(F)] < FNAM2(C21 C22) NOL
      OR
CSPEC FRF[(F)] < FNAM2[(C2)] FNAM3[(C3)] NOL

CUT [DNAM1] < DNAM2 IB IE

DELET FNAM1[(DMODE1)] [FNAM2[(DMODE2)] ... ]
   DMODE: 'D'/'T'

DETER DNAM1[(C1)] < SNAME DNAM2[(C21 ..)]
                  [DNAM3[(C31 ..)] [....]..] [NP]

DFT [(SW)] [(WND)] FRF < DATA[(C)]
   SW:'AMP'/'POW'
   WND: 'BC'/'BH'

DSIM DNAM1[(C1)] < SNAME DNAM2[(C21 ..)]
                 [DNAM3[(C31 ..)] [....]..] [NP]

EDIT FNAME

FHEAD FILE [LOOK [NR]]
   SUBCOMMANDS:
   INDEX VALUE
   LOOK [NR]
   KILL
   X

FILT PNAME < FITYP NO DELTAT OML [OMH]
   FITYP: 'LP'/'BP'/'HP'/'BS'

FORMAT [FFILE] < BFILE[(C1 C2 .. )]

FROP [FRF1[(F1)]] < FRF2[(F2)] OP FRF3[(F3)]
   OP: '+'/'-'/'*'/'/'

FTEST FNAME [(DMODE)]
   DMODE: 'D'/'T'

GETFIL PROGFILE FILESPEC [FILESPEC ... ]

IDFT DATA < FRF[(F)]

INSI FNAME [(C)] NP [TSAMP]
   SUBCOMMANDS:
   PRBS [IBP [NBIT ] ]
   NORM [RMEAN SIGMA]
   RECT [A B]
   SINE [OMEGA FI]
   ZERO
   STEP
   RAMP [A B]
   PULSE [LENGTH]
   SRTW [PS]
   LOOK
   KILL
   X

LIST [(DEV)][(FEED)][(DMODE)] FNAME[(A1 A2..)] [IF NUM]
   DEV: 'DIS'/'LP'/'TP'
   FEED: 'LF'/'FF'
   DMODE: 'D'/'T'/'DS'/'TS'

LS [(SW)] SYST < SFIL
   SUBCOMMANDS:
   SAVE STDEV
   SAVE COMAT
   KILL
   X
   SW: 'VOID'/'SC'

ML [(SW)] SYST < DATA[(C1 .. )] NO
   SUBCOMMANDS:
   INVAL 'ABC'/'C' SYST
   FIX A (2) [VA2] (3) [VA3] B (21) [VB21] .....
   SAVE [STDEV] [GRAD] [EVALS] [COMAT]
   LOOK
   KILL
   X
   SW: 'VOID'/'SC'

MOVE [(OUTP)] [(DMODE)] [ FILOUT [(C11..)]]  <
      [(INP)]  FILIN [(C21..)]
   OUTP: 'DK'/'DT'/'PP'
   DMODE: 'D'/'T'
   INP: 'DK'/'DT'/'PR'

PICK FNAM1 < FNAM2 NR

PLMAG DATA [(C)]
   SUBCOMMANDS:
   B[LOCK] NB
   P[LBEG] NR
   A[LTER] NR [NUM]
   PA[GE]
   D[ELET] NR [NUM]
   KILL
   X

PLOT [(NP)] [(OPT1)] FNAM1[(C11..)]
      [[(OPT2)] [FNAM2[(C21..)]] .. ] [YMI YMA]
   OPT: 'LI'/'HP'/'NL'
   SUBCOMMANDS:
   KILL
   PAGE
   SKIP [N]

RANPA SNAM1 < SNAM2

RESID RES[(C1)] < SYST DATA[(C11 C12 .. )]
   SUBCOMMANDS:
   KILL
   PAGE
   TABLE

SAVFIL PROGFILE FILESPEC [FILESPEC ..]

SCLOP [FNAM1[(C1)] < FNAM2[(C2)] OP CONST
   OP: '+'/'-'/'*'/'/'

SLIDE [FNAM1] < FNAM2 K1 K2 K3 ..

SPTRF [(SW)] FRF[(F1)] < SYST TPN / TPD
   SW: 'AMP'/'POW'

SQR RFIL < FNAME [(C1 C2 ..)] SFIL

STAT FNAME [(C)]

STRUC SNAM2
   OR
STRUC [SNAM2] < SNAM1
   SUBCOMMANDS:
   REVERT
   NA    [SW] NVAL
   NU    [SW] NVAL
   NB    [SW] NV1 ... NVNU
   KB    [SW1] NV1 ... NVNU
   FIX A(N) [VN] (M) [VM] ...
        B NU1 (N1) [V1] (N2) ... B NU2 ...
   UNFIX A(..  N .. M ..) B NU1 (N1 ... NN) ..
   SW : 'MAX' / 'ACT'
   SW1: SW / 'MIN'
   KILL
   X

TREND [FNAM1[(C1)]] < FNAM2[(C2)] NO [IF IL]

TURN SWITCH STATE
   SWITCH: 'TEXT'/'TIME'/'GRAPH'/'DK'
   STATE: 'ON'/'OFF'    ('H'/'M'/'S'  FOR 'TIME')

VECOP [DNAM1[(C1)] < DNAM2[(C2)] OP DNAM3[(C3)]
   OP: '+'/'-'/'*'/'/'
```

SUMMARY OF CLOSE DOWN, START UP, INITIALIZATION, AND PAPER PLOTTING WHEN USING 4025 WITH ATTACHED 4006 PLOTTER

Close down of terminal

1. SAVFIL FNAME DATA-D SYSTEM
2. Press SHIFT-F1
3. STOP
4. S
5. @SYM PRINTFILENAME,,SMSK36
6. @FIN
7. @@TERM
8. Terminal power off.

Close down of plotter 4662 and line printer SMSK36

1. Plotter power off.
2. Hat on plotter pen.
3. Line printer power off.

Start up of terminal T4025

1. Terminal power on. Switch on right side.
2. TTY LOCK down, COMMAND LOCK OUT released.
3. !SYS (!BAU 1200, !ECH R, !PAR E)
4. Press RETURN
5. U/P
6. @RUN HEJ,999999,HEJ
7. @ASG,AX FNAME
8. @ADD ID*IDPAC.START
11. Press F1
12. Press RETURN
    You are now inside IDPAC. Initialize.

Start up of plotter and line printer SMSK36

1. Hat of plotter pen. Put down pen arm.
2. Plotter power on.
3. Line printer power on.
4. Top set if necessary.
6. Make sure that the button AUTO is lit up.

Initialize

1. SWITCH LOG ON
2. GETFIL FNAME FILON DATA-D
3. LET TICK. = <integer> (1)
4. LET DELTA. = <integer> (1)
5. TURN TIME {OFF/S/M/H} (OFF)
6. LET NPLX. = <integer> (100)
7. LET SCALES. = {0/1} (1)
8. LET WMIN. = <positive real number> (0.01)
9. LET WMAX. = <positive real number> (100)
10. LET NOF. = <integer> (100)
11. HCOPY ON

How to paper plot

1. Press the LOAD-button of the plotter
2. Insert paper
3. Release the LOAD-button
4. Print PLOT
5. Print HCOPY