88-32

Reniers D., Vandewalle J., De Moor B., ``A personal computer network for didactical purposes in engineering education", in *Proc. SEFI 1988 Annual Conference*, Leuven, Belgium, 30 Aug.-2 Sep. 1988, pp. 341-346., Lirias number: 181555.

A DIDACTICAL PROJECT WITH A PERSONAL COMPUTER NETWORK FOR ENGINEERING EDUCATION

D. Reniers, J. Vandewalle, B. De Moor

Department of Electrical Engineering, E.S.A.T. laboratory, Katholieke Universiteit Leuven.
Kardinaal Mercierlaan 94, 3030 Leuven, Belgium
Tel. 32 / 16 / 22.09.31
Telex 25941 elekul
Fax 32 / 16 / 22.18.55

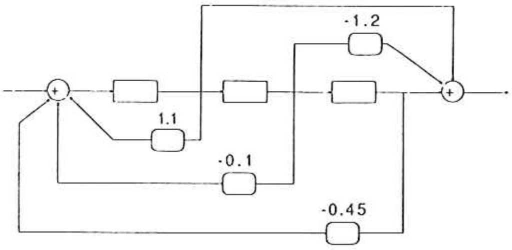
Abstract. At the ESAT laboratory (Electronics, Systems, Automation, Technology) of the K.U. Leuven extensive efforts have been made since 1986 on the implementation on a PC network of exercise sessions for students. The use of a computer permits the student to spend more time on testing and analyzing several solution methods, conditions, restrictions and properties of certain techniques etc. instead of waisting time on boring repetitive calculations. Each year 13 different exercise sessions take place on the network, like linear and non linear systems, switching and control theory, digital datatransmission etc. A lot of hardware and software is available on the network and can be used by assistants for their research or by students for their thesis.

Keywords. Computer software; computer testing; education; local area network; personal computer.

INTRODUCTION

For an engineering student it is necessary to take exercises in order to assimilate several concepts, relations and engineering techniques. This however can be very time consuming.

Let's take a typical example of the course Signals and Systems of the 3rd year of electrical engineering (see Fig. 1). It takes a student 30 minutes to solve this problem. The same student can read the exercise, model it, enter it in a PC and solve it in 14 minutes (see Fig. 2 a & b). This means that half of the time comes available for experimentation: what would happen if the poles were omitted or if that parameter changes, what would happen to the impulse response if I shift the complex poles etc.



a) Calculate the state equations and the difference equation
b) Calculate the impulse response: numerically by reasoning on the system
Calculate the general analytic expression, starting from the difference equation
c) Calculate the transfer function and verify it with the impulse response (use your tables);
Calculate the poles and zeroes
Is the system input-output stable?
d) Calculate the response for
u[k] = cos (2k)
u[k] = cos (2k+1) + 2 step (k)

Fig. 1. A typical exercise of the course signals and systems.

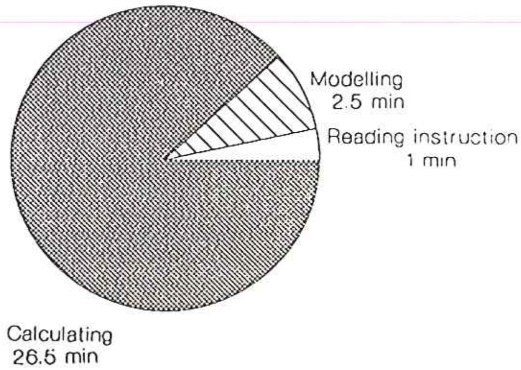


Fig. 2a. Solving the example with pen & paper (total time: 30 minutes)

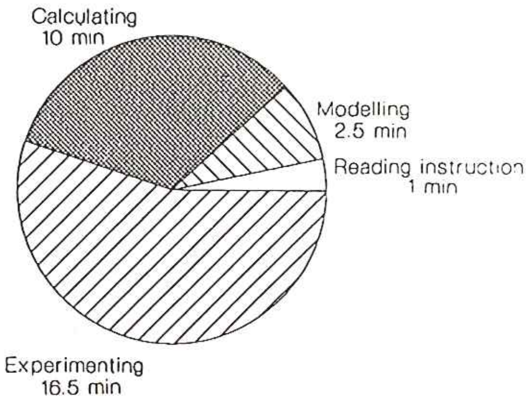


Fig. 2b. Solving the example on a PC with a scientific program (total time: 30 min)

PURPOSE OF THE NETWORK

A Personal Computer can be very useful for many exercises:
.A PC can make fast calculations which allows students to compute and study several times the same problem, using different inputs or parameters in order to investigate the influence of changing conditions.
.A PC has a lot of graphical possibilities: abstract principles like convolution, modulation, Bode plots, aliasing, bifurcation, root loci, transfer functions, limit cycles and Nyquist curves can easily be visualized (see Fig. 3, 4 & 5).

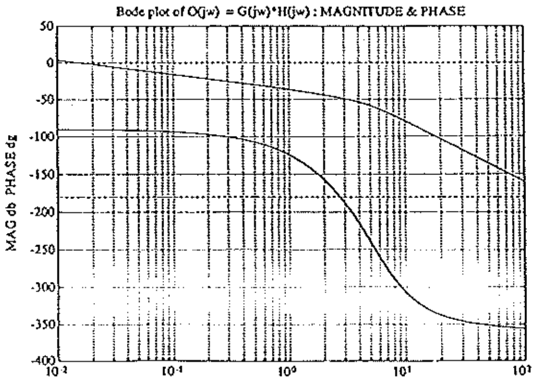


Fig. 3. A Bode plot from the session of control theory (PC-Matlab)

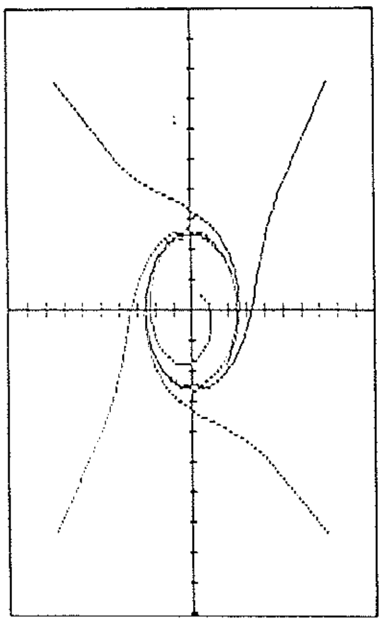


Fig. 4. Graphical representation of a limit cycle from a session of non linear systems (Phaser)

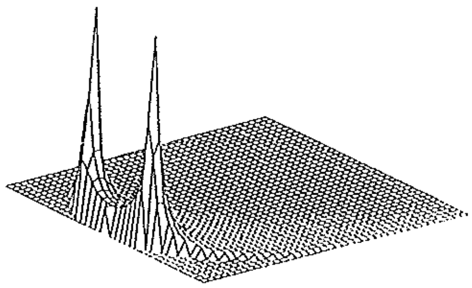


Fig. 5. The transfer function of a system in the complex plane from the session on time discrete systems (PC-Matlab)

Simulations of analog or digital systems at gate or transistor level can be done without any problem (see Fig. 6).
.A PC makes it possible for each student to work at his own speed and to repeat parts he didn't understand quite well yet.
.By making exercises on the network the engineering student gets an opportunity to work on a Personal Computer which is always a useful experience.

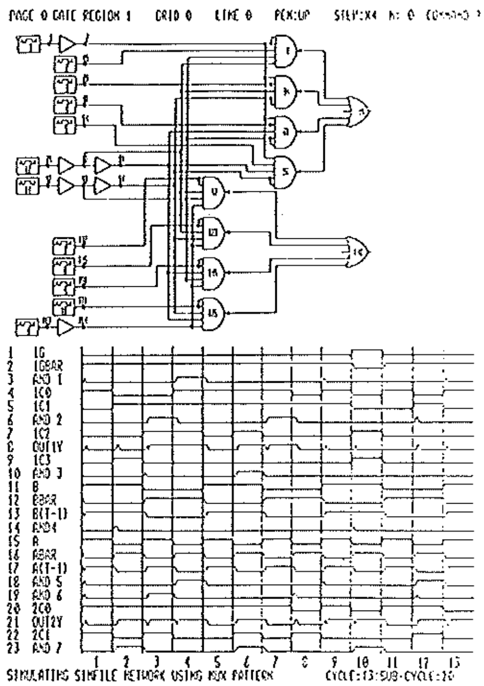


Fig. 6. A simulation of a digital circuit from the session of switching theory (Micro-Logic)

CONFIGURATION OF THE NETWORK & HARDWARE

In September 1986 ESAT started up a Local Area Network with 20 Personal Computers. PC's were chosen instead of other computers (a Vax-network or minicomputers e.g.) since the PC is nowadays a de facto standard in the industry, since it is more flexible and represents a smaller investment.

The network is an EtherLink network with 2 servers (the central memories of 70 and 20 Mbyte), each with a different network operating system: 3Plus and EtherSeries. The PC's are mostly Commodore PC 10, PC 20 or PC 40, but there's also an Olivetti M24, a Systech and an IBM PC. There are several printers available. There's also a link with a Vax computer.

SOFTWARE

There are a lot of software packages available on the network: scientific programs like PC-Matlab, Simnon, MS-Kermit, Phaser, Spice and Micro-Logic, graphical programs like the applications of Gem and PC-Storyboard, databases and editors like Volkswriter, WordPerfect, MS-Word and MicroTex etc.
Whenever it was possible, software packages were used to make the exercises. Scientific programs are normally powerful, fast, thoroughly tested and less expensive than self developed programs. Most exercise sessions are based on PC-Matlab, Phaser and Micro-Logic.

PC-Matlab is the PC version of Matlab (matrix laboratory). It is an interactive system with a matrix that doesn't require dimensioning as basic data element. It allows the user to solve many numerical problems in a very short time. The program is very user friendly and it is very easy to insert commands. Graphical routines for 2 and 3 dimensional plots are available. Users can define their own command files and functions using loops (for..., while...), tests (if...then...else) and some basic routines for input and output of results (input, disp, plot etc.).

Phaser is versatile, interactive, menu driven program for the simulation of systems of non linear differential and difference equations. The user first creates a suitable window configuration for displaying a combination of views. Then he can specify various choices in preparation for numerical computations. The solutions can be manipulated graphically.

Micro-Logic II is an interactive, digital logic drawing and simulation tool. It allows you to sketch (with a mouse) a logic drawing and automatically creates a netlist suitable for simulation. It provides a shape editor and contains a library of 200 components, including boolean functions, data channels, clocks, flip-flops and a number of common TTL/CMOS MSI logic functions.

STRUCTURE OF A SESSION

All the exercise sessions are menu driven and have the same pattern. A typical example is the session of linear time discrete systems. The main menu looks like this:

1. Help
2. Demo
3. Exercises
4. Define Systems and Signals
5. System Characteristics
6. Response to an Input or Initial State
0. Stop

Enter you choice (0 .. 6)

1. Help explains the user the purpose of the session and how the program can be used. Eventually it contains a summary of the subject-matter. There's a help function at each level of the menu.

2. Demo gives a survey of all possible operations on the basis of a typical example.

3. Exercises gives an amount of problems which can be solved by the student.

4. Define Systems and Signals enables the student to create his own systems and signals which can be analyzed and simulated later on. The menu looks like this:

SYSTEMS
1. Help
2. Matrices
3. Difference Equation
4. Transfer Function
5. Poles and Zeroes
6. Predefined systems

SIGNALS
7. Help
8. Library
9. Define points
10. Define signals
0. Main menu

Enter your choice (0 .. 10)

5. System Characteristics analyzes the defined systems and deals with all the properties of the systems:

NUMERICS
1. Help
2. Matrices
3. Difference Equation
4. Transfer Function

GRAPHICS
5. Help
6. Block Diagram
7. Impulse Response
8. Eigenvalues & -vectors
9. Poles, Zeroes & Gain
10. Frequency Response
11. 3D complex Transfer Function
0. Main menu

Enter your choice (0 .. 11)

6. Response to an Input or Initial State simulates a defined system with a defined signal.

All the student has to enter in the PC during this session are the numbers of the menu, numerical data and analytic expressions of the signals.

LIST OF EXISTING SESSIONS

In Belgium, the engineering education consists of two parts: the candidatures and the technical years. During their first two years all engineering students have to take the same courses. They all have to learn several aspects of mathematics, the principles of physics, chemics, mechanics etc. After these candidatures, the students can choose between several specializations like electrical engineering, mechanical, computer, chemical etc. All the exercise sessions on the network are made by the students of electrical engineering: 2 sessions by the students of the 1st technical year, 6 by the students of the 2nd year and 5 by the students of the 3rd year.

1. Time Discrete Systems

This session deals with linear time invariant time discrete systems. The student can define and simulate systems in order to get familiar with the different forms of presentation of a system (block diagram, state equation, difference equation, transfer function) and with its properties like impulse response, eigenvalues, poles etc. See also Fig. 5, 7 & 8.

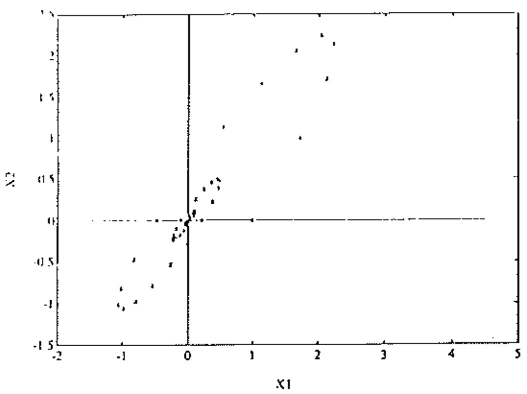


Fig. 7. The evolution of a system in the phase plane after an impulse input - time discrete systems (PC-Matlab)

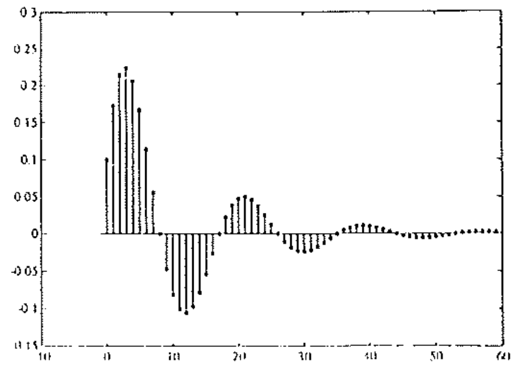


Fig. 8. The impulse response of a self defined system - Time discrete systems (PC-Matlab)

2. Signal Analysis

This session permits the student to define signals and manipulate them using filters, Fourier and inverse Fourier transformation, convolution etc.

3. Control Theory

The student defines linear systems and signals and studies poles and zeroes in the complex plane, an analytic approximation of a system by a second order system, time and frequency domain analysis, root loci (see Fig. 9), phase and gain margin, Bode, Nyquist and Nichols plots (see Fig. 4).

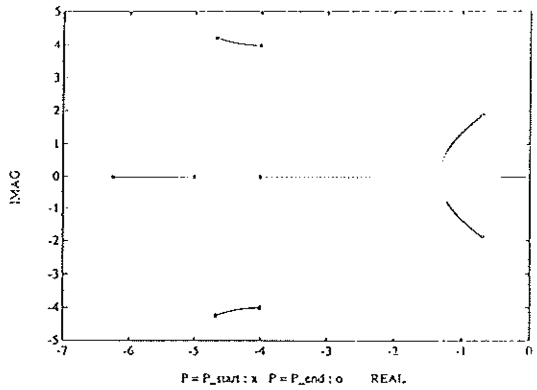


Fig. 9. A root locus - control theory (PC-Matlab)

4. Switching Theory

The student has to design logic circuits at gate level and simulate them: he has to study a multiplexer, a R-S flip-flop and a shift register and to realize a J-K flip-flop and a 4 bit counter, given the state table (see Fig. 6).

5. Information Transmission

This program displays spectra of different signals and learns the user about amplitude modulation (AM) and demodulation and about frequency modulation (FM) and demodulation.

6. Phase Locked Loops

The definition of a linear phase locked loop (PLL) is given, the linear PLL is simulated (explanation of its transfer function, time and frequency response) and some application of a PLL are mentioned: amplitude modulation and demodulation and the controller.

7. Digital oscilloscope

This self developed program deals with the sampling of a signal and the influence of the sample frequency on the reconstruction of the original signal, with the pros and cons of a digital oscilloscope in comparison with a analogue oscilloscope and with the signal processing possibilities who enable the user to compare the original signal with the signal after processing, in the time domain or after a fast Fourier transform (FFT).

8. Logic Analyzer

The student learns the functioning and the use of a logic analyzer. A circuit with a counter and a ROM is used as a unit under test.

9. Non Linear Systems 1

The student calculates and interprets graphically non linear differential equations and non linear systems: presentation of trajectories, limit cycles (see Fig. 4), simulation of harmonic oscillators, of the three body problem (see Fig. 10) and of the jump phenomenon.

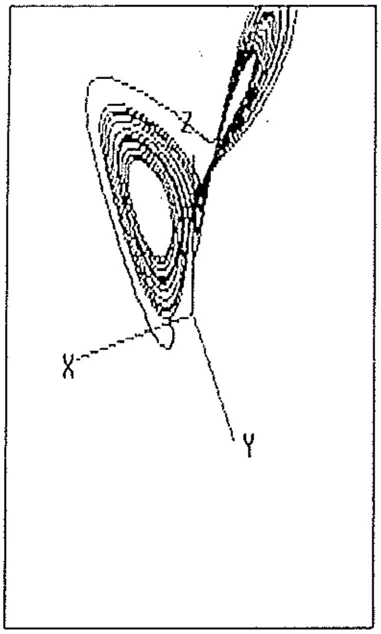


Fig. 10. Simulation of a three body problem (circulation of a planet in the gravity field of 2 other planets) - non linear systems (Phaser)

10. Non Linear systems 2

The student investigates the stability of non linear systems with the method of Popov, describing functions, linearization and Nyquist and the response on an initial state with a certain input.

11. Digital Electronic Systems

The student has to design and simulate logic circuits on gate level. He has to study a master-slave flip-flop and has to design two sequential circuits in pulse mode.

12. Autonomous Shift Registers

The students have to design and simulate a pseudo random noise generator with ordinary components as flip-flops and exors.

13. Algebraic Codes

This program enables the user to design and simulate an encoder-decoder that can restore a single bit error in a codeword by adding redundancy to this word.

REALIZATIONS

Evaluation of the PC Exercises

Since it was the main purpose of the network to improve didactically the exercises for the students, it was important to ask the student for his opinion. After each session the student gets the opportunity to fill in a form anonymously; he can tell whether he totally agrees, agrees, has no opinion about, doesn't agree or totally disagrees with some statements. The answers to the most relevant questions will be discussed.

The majority of the students (totally) agrees that they understand the theory better after they've made the exercises. To the questions if the programs are easy to use and fast enough, opinions are divided; here we have a certain duality: the easier the exercise session are to use, the slower they are and the faster we make the programs, the harder they are to use. So for each session we have tried to make the best compromise between speed and user friendliness.

Depending on the session 60 to 85% of the students (totally) agrees that a PC session is more interesting than a classic ex cathedra exercise session; 0 to 5% totally disagrees.

Research

The available software, especially PC-Matlab, is also frequently used by engineering assistants for research. Furthermore, each year approximately 10 students make their thesis on the network.

CONCLUSION

The realization of a network of 20 Personal Computers is a success. It is intensively used by the students for exercise sessions and research; their reactions are very positive. Although substantial financial and human efforts already have been made, there's still work to be done: expansion of the hardware (more PC's, memory capacity and printing facilities), adding new software and programming other exercise sessions.

REFERENCES

Hill,F.J., and G.P.Peterson (1981). Switching Theory and Logical Design, 3rd ed. John Wiley & Sons, New York. pp. 232-387.

Koçak,H. (1985). Differential and Difference Equations through computer Experiments. Springer-Verlag, New York.

Moler,C., J.Little, and S.Bangert (1987). PC-Matlab for MS-DOS Personal Computers. The MathWorks Inc., Sherborn.

Press,W.H., B.P.Flannery, S.A.Teukolsky, and W.T. Vetterling (1987). Numerical Recipes: the Art of Scientific Computing. Cambridge University Press, Cambridge.

Reniers,D., and others (1987). Software for educative purposes in control theory. Benelux Meeting on Systems & Control, 194.

Thompson,A.V. (1983,1987). Logic Design and Simulation System. Spectrum, Sunnyvale.

Shanmugan,K.S. (1979). Digital and Analog Communication Systems. John Wiley & Sons, New York.