Remote Control Laboratory with Moodle Booking System

Suzana Uran, Darko Hercog, Karel Jezernik
Faculty of Electrical Engineering and Computer Science
University of Maribor
Maribor, Slovenia
Email: suzana.uran@uni-mb.si, darko.hercog@uni-mb.si, karel.jezernik@uni-mb.si

Abstract— **In order to enhance learning of control design, minimise the gap between theory and practice and especially to support learning by doing we built remote control laboratory. This paper describes a Moodle based booking system for our remote laboratory. The web-based remote laboratory is presented through a RC oscillator experiment for learning of control design. In RC oscillator experiment, interactivity between the controller design parameters in Bode plot and Root Locus with real experiment has been implemented in order to support full visualisation of the controller design. An experiment for control of a mechatronic device, it is a DC motor, was implemented. Special safety measures built into the experiment are represented in the paper.**

I. INTRODUCTION

The Internet (Web) has become a widespread tool for teaching and learning. The Web enables more flexible delivery (anytime), distance education (anyplace), new visualisation possibilities (interactivity), and cost reduction. Virtual and remote web-based laboratories [1 - 4, 7] have been developed to date. Virtual laboratories are usually opened to the public. Remote web-based laboratories are based on sharing the same resources over the Internet [2, 3, 4]. Therefore, a booking system for reservation of the remote laboratory experiments is needed. And therefore remote web-based laboratories are usually available only to a limited number of assigned students. In the paper a Moodle based booking system and a remote web laboratory that supports the learning of control theory and control design are presented RC oscillator control and DC motor control web-based remote experiments are represented in greater detail.

A modification of Moodle booking system implemented with our remote web laboratory is represented in section II.

The objectives of our remote web laboratory are:

- to teach students control design,
- to minimize the gap between control theory and practice, by teaching control implementation [5],
- to show students how to learn by Web and how to use it and
- to support learning by doing.

When teaching control we should be aware of the gap between the theory and praxis [2, 5] and do our best to avoid this gap. More attention should be given to the control implementation details and their consequences in order to avoid the gap between the theory and praxis. Remote laboratory control experiments are the best means to consider control implementation details and their consequences using Web. An RC oscillator interactive experiment with control design visual support is presented in section III. For RC oscillator controller design, a visualisation of controller design parameters is implemented interactively with the remote control experiment.

With DC motor control experiment safety measures built into the implementation of the DC motor control are discussed. DC motor control experiment is also represented in section III. Section IV summarizes this article with some final thoughts.

II. MOODLE AND MOODLE BASED BOOKING SYSTEM

Moodle [8] is a widely adopted software package for producing internet-based courses and web sites. It's an ongoing development project designed to support a social constructionist framework of education. The word Moodle was originally an acronym for Modular Object-Oriented Dynamic Learning Environment, which is mostly useful to programmers and education theorists. It's also a verb that describes the process of lazily meandering through something, doing things as it occurs to you to do them, an enjoyable tinkering that often leads to insight and creativity. As such it applies both to the way Moodle was developed, and to the way a student or teacher might approach studying or teaching an online course [8]. Moodle has a very large user base with 12,165 registered sites in 155 countries with 4,021,531 users in 376,565 courses (May 30, 2006) [10].

Moodle is provided freely as Open Source software (under the GNU Public License). Basically this means Moodle is copyrighted, but that users have additional freedoms. Users are allowed to copy, use and modify Moodle provided that they agree to: provide the source to others; not modify or remove the original license and copyrights, and apply this same license to any derivative work [8]. Moodle runs without modification on UNIX, Linux, FreeBSD, Windows, Mac OS X, NetWare and any other systems that support PHP. Data is stored in a single database: MySQL is best supported, but it can also be used with commercial databases.

Moodle booking system for online labs [9] assumes that remote experiments and Moodle course management system run on the same computer. While this is usually the case in the

1-4244-0755-9/07/$20.00 ©2007 IEEE

virtual laboratories, in majority of remote laboratories, experiments and CMS operate on separate computers.

A. Original Booking System

In original booking system [9], the following information's must be provided by an experiment administrator during the booking creation process: booking name, booking description, the maximum number of students that can work on experiment at the same time, experiment URL address and the time interval when the booking link will be available (Fig. 1). Actual experiment URL address is generated dynamically at runtime and it is created randomly based on URL address entered in the *URL base (fixed string)* field. This field should contain the experiment Web address with the *[[ACCESSKEY]]* string near the end of the line.

At the runtime, the *[[ACCESSKEY]]* string is replaced with randomly generated 32-character access key. When the remote user selects the booking link, a 7-day calendar (timetable) appears (Fig. 2). The calendar page shows the local time (at the user's time zone), which may of course be different from the server time. The time conversion takes place automatically, so that users at any time zone see the free/booked time slots market according to their local time zones. Each column in calendar is divided into 24 time slots (cells). By clicking on the book icon ( ) inside individual calendar cell, the remote user is able to book the selected one-hour time slot. Any reserved slot may be released only by its owner, by clicking on the unbook icon ( )[9].

When the reserved time slot becomes the current one-hour slot, a red arrow icon ( ) appears inside the cell near the booking name (Fig. 2). This icon represents a valid link to the remote experiment. When the remote user selects the red arrow icon, access key is firstly added to the experiment file name, and afterwards the user is redirected to the newly created web page.

B. Modified Booking System

In addition to the original booking system, modified solution enables easy booking creation for remote experiments that are developed with LabVIEW Web publishing tool and during their operation use LabVIEW Remote Panels technology. Remote experiments can execute on the same computer as Moodle CMS or on the separate one. Although, Remote Panels includes an algorithm, to prevent hardware access conflicts by different remote users, current solution does not provide booking possibility.

LabVIEW Web publishing tool and Remote Panels technology

Using LabVIEW Web publishing tool, LabVIEW front panels can be quickly and effortlessly publish on the Web. Once published, anyone on the Web with the proper permissions can access and control the experiment without installing LabVIEW. LabVIEW Web publishing tool automatically creates HTML document for selected LabVIEW virtual instrument (VI) and store it, by default, to the

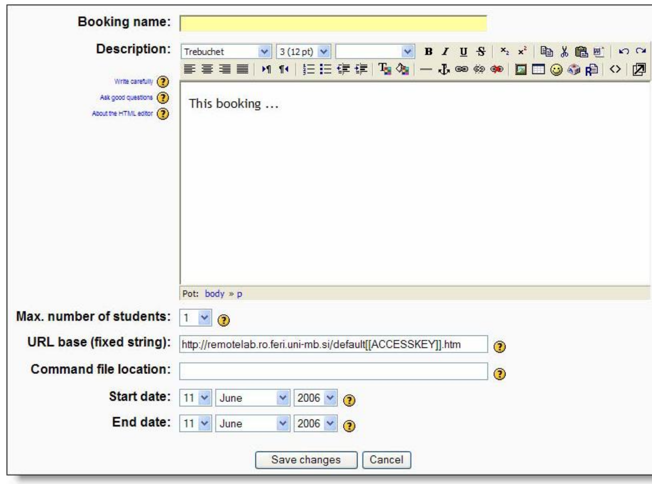


Fig. 1. Booking creation window.

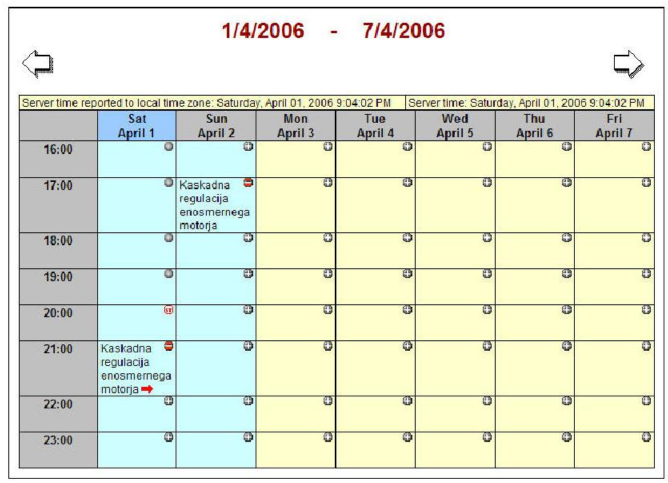


Fig. 2. Experiment booking timetable.

LabVIEW WWW folder. When a remote viewer enters appropriate URL address, the LabVIEW front panel appears in the Web browser. Once connected to the remote experiment, the client connection is automatically in a monitor state. If another client is controlling the remote experiment, the remote client is able to monitor the actions of the controlling client. To request control of the program, right click on the front panel and selection of Request Control is required. If another client has control, the controlling client is notified that control time has now become limited. Once the timeout occurs or the controlling client released control, application control is automatically switched to the requesting client. Once the user granted control, GUI controls become active and running the LabVIEW application is like running the application from the local environment.

Booking system for remote experiments developed with LabVIEW Web publishing tool

Modified booking system preserves the same booking creation principle as original one. The only difference appears in *URL base (fixed string)* field. In contrast to the original booking system, which expects *[[ACCESSKEY]]* string in this field, modified solution expects *[[REMOTELAB]]* string. This string informs the booking system that this booking was created for experiment that is based on LabVIEW Remote Panels technology.

When the remote user selects an experiment link (➡) at the runtime, booking system checks experiment URL address (1 on Fig. 3). If *[[REMOTELAB]]* string is found, booking system creates 32-character accesses key and, along with other experiment parameters, sends it to the lab PC (2). Based on this key, program running on lab PC (*RemoteLabDataServer.llb*) creates a copy of experiment HTML file (previously created by Web publishing tool) with retrieved access key as name. After successful response from the lab PC (3), booking system redirects remote user to the newly created experiment web page (4 and 5). After this, the direct connection between remote user and remote experiment is established. Every full hour, remote users are automatically disconnected from experiments and, at the same time, all copies of original experiments web pages are deleted.

In described operation, the lab PC remains original experiment web page. Upon each request from the booking system, program running on the lab PC just creates a copy of experiment web page with the access key as a name. Using this way, original experiment web page remains hidden to the remote users and consecutively there is a low probability to access these web pages by passing the booking system.

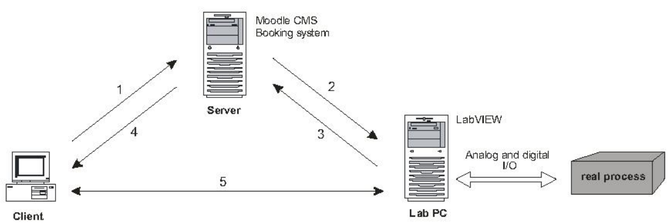


Fig. 3. Modified Moodle based booking process.

III. Remote Lab and Control Issues

A remote lab at Faculty of Electrical Engineering and Computer Science, University of Maribor was established in order to support the learning of control. This remote lab provides various remote experiments (various DC motor control, RC oscillator). Remote lab experiments are available on: http://remotelab.ro.feri.uni-mb.si/eng/experiments.asp.

A. Remote Laboratory System Architecture

Remote laboratory is temporary based on a DSP-2 learning module (Fig.4),. The DSP-2 learning module is an ‘embedded’, light and small in volume DSP-based control system. More demanding mechatronic and robotic experiments could be based also on DSP2 Robotic Controller (Fig.5). Both systems were developed at Faculty of Electrical Engineering and Computer Science, University of Maribor and are presented in detail on the address: http://www.ro.feri.uni-mb.si/projekti/dsp2.

Fig. 4. DSP2 - learning module.

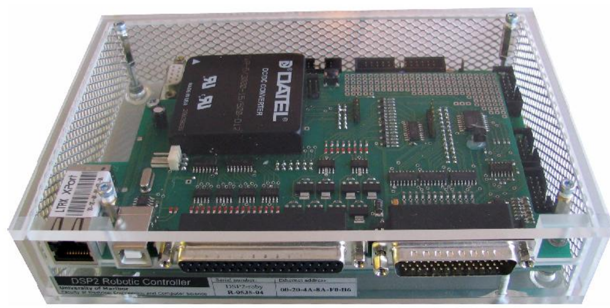


Fig. 5. DSP2 – robotic controller.

The DSP-2 learning module represents an open framework for rapid control prototyping (RCP) and rapid remote control experiment development. Fig.6 presents the block scheme of the DSP-2 learning module-based remote laboratory. A DSP-2 learning module, connected to a lab PC through the serial port, implements a control algorithm developed using Simulink [6], and through the analog and digital I/O signals, drive the real plant. LabVIEW virtual instrument and the LabVIEW server run on the same lab PC for the purpose of enabling remote control of the real plant. LabVIEW VI performs communication between the lab PC and the DSP-2 learning module, and enables DSP-2 data visualization and parameter tuning, while LabVIEW server enables remote operation of the LabVIEW VI. Remote users, connected to the server through the Internet, must have a ‘LabVIEW Run-Time Engine’ installed on their personal computer in order to perform remote experiments. During remote experimentation, the remote user can adjust the controller parameters and send experimental results via email.

The DSP-2 learning module-based RCP system is based on two commercially available software packages i.e. MATLAB/Simulink and LabVIEW, and custom-made hardware i.e. DSP-2 learning module. MATLAB, Simulink and Real-Time Workshop (RTW) are used for control algorithm development, simulation, offline analysis and rapid executable code generation [6], while the LabVIEW provides on-the-fly data visualization and parameter tuning tasks. LabVIEW virtual instrument (VI) is automatically generated during the binary code generation process, from Simulink model, where the user front end of created VI depends on special DSP-2 blocks used in the Simulink model. Using Remote Panels (LabVIEW add-on toolkit), generated VI's can be easily viewed and controlled over the Internet. LabVIEW VI's can be published on the Internet with no additional programming and can be remotely observed or controlled by using only the standard web browser and LabVIEW Run Time Engine.

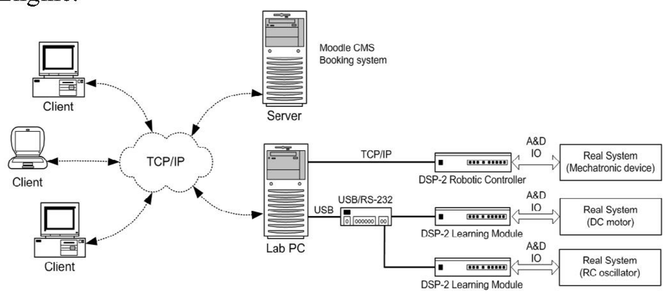


Fig. .6. DSP-2 based remote laboratory block scheme.

In the following two different remote lab experiments are briefly described.

B. RC Oscillator

The RC oscillator is a web-based version of interactive controller design experiment (WICDE). The RC oscillator experiment is used in an introductory control course at our faculty, as one of homework tasks.

Unfortunately, the policy of minimum student's software requirements in order to use Remote lab sets an undesirable limitation on the implementation of 'Learning through doing'. This limitation means that students could not build their own experiment but could only vary the parameters of the already prepared experiment. Such a limitation is widely accepted for web-based experiments. Only a web-based experiment presented in [3] assumes the MATLAB/Simulink software environment to be possessed by the students. Therefore, in the case of the web-based experiment in [3], students could build their own experiment from home.

RC oscillator experiment is implemented using a DSP-2 learning module, and a breadboard with an RC circuit (Fig. 7). The resistor and capacitor values of the RC circuit are:

$$R1 = R2 = R3 = R = 47\ \mathrm{k\Omega},$$
$$C1 = C2 = C3 = C = 1\ \mathrm{\mu F}.$$

A control loop of the RC oscillator is shown in Fig. 8. Mathematical model (transfer function) of the RC circuit is given by equation (1).

The MATLAB/Simulink scheme implemented for the WICDE RC oscillator is shown in Fig. 9. The open-loop control of RC circuit and feedback control of RC circuit are combined in one Simulink block scheme. Switch S1 is used to

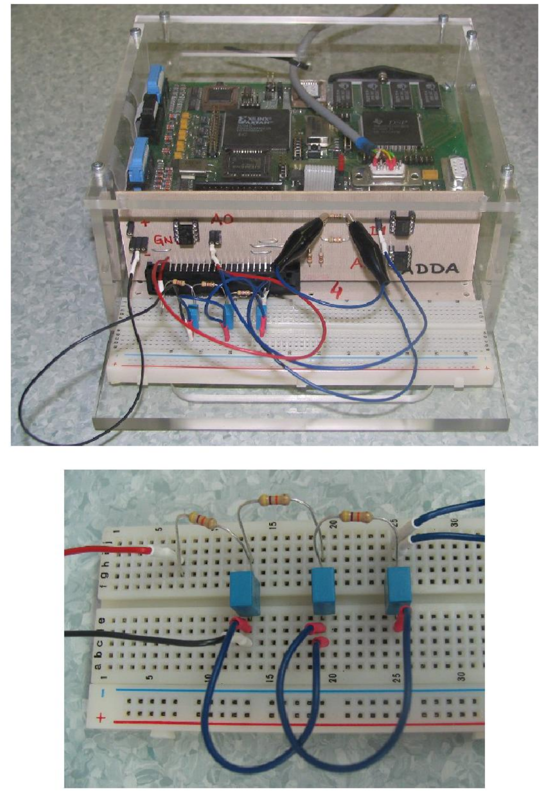


Fig. 7: DSP-2 learning module with RC circuit on the breadboard.

$$F_{RC}(s) = \frac{U_{out}(s)}{U_{in}(s)} = \\ = \frac{1}{(R\cdot C)^3\cdot s^3 + 5\cdot (R\cdot C)^2\cdot s^2 + 6\cdot (R\cdot C)\cdot s + 1} \qquad (1)$$

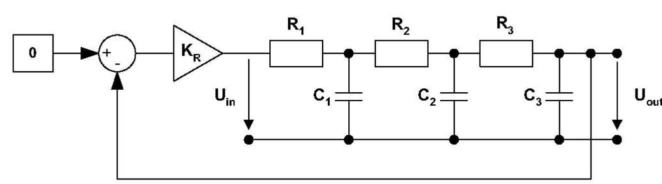


Fig. 8: RC oscillator control loop.

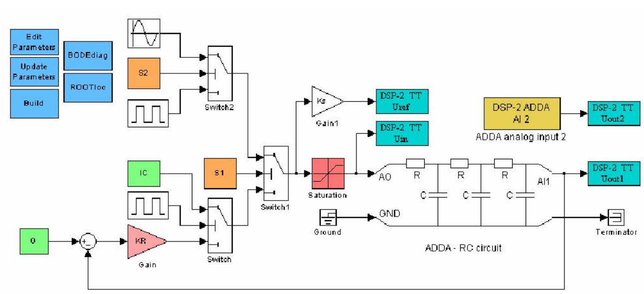
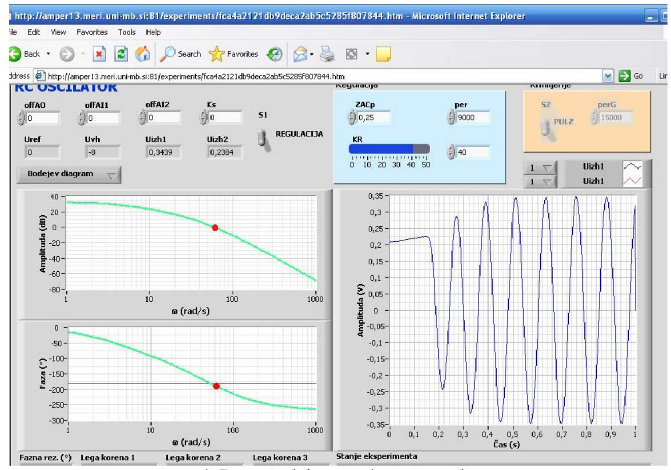


Fig. 9: Matlab/Simulink block scheme for RC oscillator.

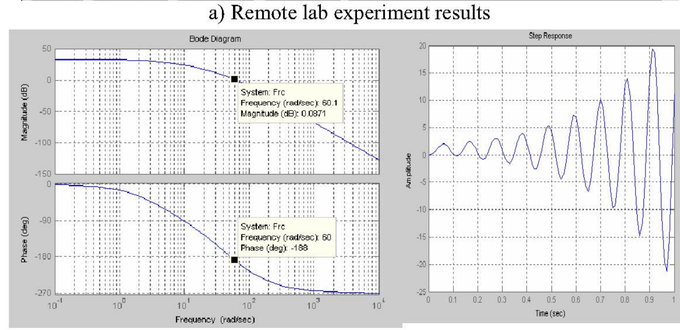
select between open loop or feedback control. When open-loop control is selected switch S2 is used to select the step or the sinusoidal input to the RC circuit. When feedback control is selected another switch takes action. This switch selects between the initial condition (IC) input and feedback with gain K_R for input into the RC circuit. Therefore two phases of the RC oscillator response are observed. The capacitors of the RC circuits are charging to the value IC when IC input is selected. This phase is called the 'charging phase'. When feedback with gain K_R is selected the RC oscillator response is observed. This phase is called the 'RC oscillator relaxation phase'. Signals Uref, Uin, Uout1 and Uout2 from the Simulink block scheme could be observed. Online tuning of the K_R (RC oscillator feedback gain), IC (the value to which capacitors should charge) and per the period of the pulse generator switching between the charging and the relaxation phase is possible. Due to delays appearing in Internet connections, delays also appear between the variation of the gain K_R and the interactive view of the RC oscillator response.

The RC oscillator Bode plot or Root Locus plot can be observed interactively with the RC oscillator response. Both mentioned plots are calculated by MATLAB on the basis of the RC circuit transfer function given in (1). Controller design parameters, such as phase margin and crossover frequency, are clearly marked in the Bode plot. Accordingly in Root Locus the actual roots of the control loop are clearly marked. Bode plot or Root Locus design is selected with the left button above the design diagrams in the LabVIEW front panel (Fig. 10a).

The objective of the RC oscillator remote experiment is reached when students found and try to explain the difference between the RC oscillator time response of the remote experiment and the time response of the RC oscillator that is calculated from model in Matlab (using Matlab Web Server) shown in Fig. 10a,b [11].



a) Remote lab experiment results



b) MATLAB - model based experiment

Fig. 10. a,b. Bode plot design and RC oscillator time response (K_R = 40).

C. DC Motor Control Experiment

DC motor is one of the simplest mechatronic devices. Control of DC motor is one of the applications very frequently met in the industry therefore it is a very frequent learning task for students. In order to explore demands of mechatronic remote laboratory DC motor control experiment was developed (Fig.12). In contrast to the RC oscillator experiment that is relatively undemanding for 24-hour operation this is not the case with the DC motor. As a mechatronic device DC motor is exposed to wear therefore any unsupervised and not intended rotation of the DC motor is not desirable. Special care should be taken in a design of a remote DC motor experiment to avoid such undesirable situations.

In the DC motor remote experiment it is necessary to build a model of DC motor current control loop and later on to design DC motor PI speed controller. In the remote experiment safety problems appear during measurement of the DC motor current loop step response. If during the remote experiment the desired current value is set to constant then the speed of a motor would go to maximal value and stay there. The same could happen if input signal from pulse generator does not have an average value equal to zero. Due to the wear of a DC motor this is not

Fig. 11. DC motor control setup.

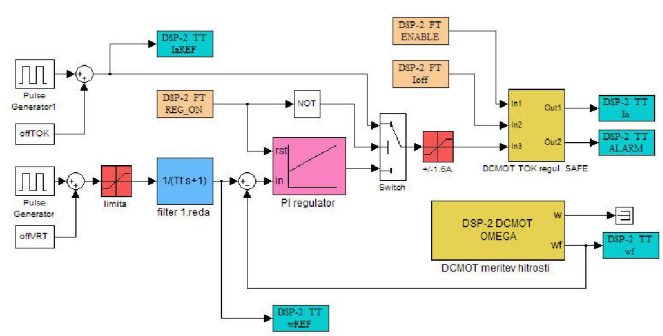


Fig. 12. Matlab/Simulink block scheme for DC motor control.

desired. In addition with our booking system a student gets control over the experiment for an hour. The student might not

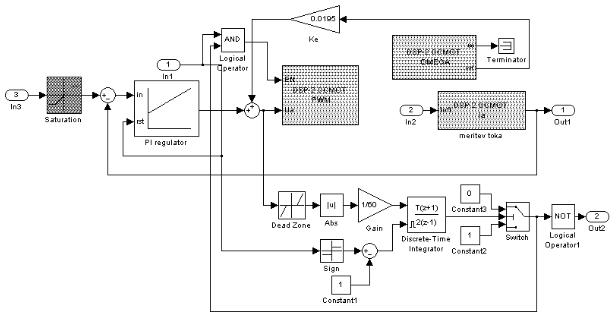


Fig. 13. Safe version of DC motor current control block scheme.

be aware what are the causes of his/hers action therefore in the worst case the motor might be left to rotate at max. speed for up to an hour. In order to avoid such situation a special safety version of current control shown in Fig.13 was designed. For safety reasons absolute value of DC motor armature voltage is integrated. If some predefined value is reached then the enable signal of DC motor power supply is turned off and the motor slowly stops. If desired by the remote operator enable signal to DC motor could be set again.

IV. Conclusion

To support learning by doing of our students at home remote laboratory was built. Our remote control laboratory is based on DSP2 learning module and two software packages: Matlab and LabVIEW. In the remote laboratory Matlab serves as a tool for control implementation and LabVIEW for on the fly data visualization and tuning of parameters. In the paper represented Moodle booking system is used to reserve remote experiments.

In RC oscillator experiment interactivity between the controller design parameters in Bode plot and Root Locus with real experiment has been established. The main purpose of the experiment is to make students aware of the gap between control theory and practice. The RC experiment was successfully used for student homework in two previous school years. When homework is given all students typically want to reserve and perform the remote experiment in the last evening and the last hour before the deadline for homework. But this is not possible in spite of the fact that the number off our students in the class is relatively low (around 15). Therefore deadline for the homework must be extended for a week. On the second attempt students successfully perform the remote experiment. Mechatronics part of the remote lab was started. A DC motor speed control was implemented and special safety was built into experiment to reduce unnecessary mechanical wear.

Acknowledgment

This work has been performed within the project „E-learning Distance Interactive Practical Education (EDIPE)". The project was supported by the European Community within framework of Leonardo da Vinci II program (project No CZ/06/B/F/PP-168022). The opinions expressed by the authors do not necessarily reflect the position of the European Community, nor does it involve any responsibility on its part.

References

[1] S. Dormido, "Control Learning: Present and Future", In Proc. 15th IFAC World Congres on Automatic Control,Barcelona, Spain, 2002.

[2] C. Schmid, "Internet-basiertes Lernen", *Automatisierungstechnik,* vol. 51, No. 11, 2003, pp. 485-493.

[3] M. Casini, et all, "The Automatic Control Telelab", *IEEE Control Systems Magazine*, vol. 24, No. 3, 2004, pp.36-44.

[4] H. Hoyer et all, "A multiuser Virtual-Reality Environment for a Tele-Operated Laboratory" ", *IEEE Trans. on Educ* , vol. 47, No. 1, 2004, pp. 121-126.

[5] G. Ellis, Control System design guide, A Practical Guide, Elsevier Academic Press, 2004.

[6] D. Hercog, K. Jezernik: "Rapid Control Prototyping using MATLAB/Simulink and DSP-based Motor Controller", *International Journal of Engineering Education (IJEE)*, Vol. 21, No. 4, 2005.

[7] A. Valera et all., "Virtual and Remote Control Laboratory Development," *IEEE Control Systems*, vol. 25, pp. 35–38, Feb. 2005.

[8] Moodle: the Moodle course management system, http://moodle.org.

[9] José M. Martins Ferreira, António M. Cardoso: A Moodle extension to book online labs, Remote Engineering and Virtual Instrumentation Symposium (REV 2005), Brasov, Romania.

[10] Wikipedia: http://en.wikipedia.org/wiki/Moodle.

[11] S.Uran et all., "Matlab Web Server and Web-based control design learning ," Proc. *IECON'2006 conf.*, pp.5420-5425, Paris, France, *Nov. 2006.*