Int. J. Engng Ed. Vol. 23, No. 6, pp. 1210–1218, 2007
Printed in Great Britain.

0949-149X/91 $3.00+0.00

Distance Practical Education for Power Electronics*

P. BAUER
Delft University of Technology, Faculty EWI, EPP Group, Delft, The Netherlands.
E-mail: p.bauer@ewi.tudelft.nl

J. DUDAK and D. MAGA
Faculty of Mechatronics, Alexander Dubcek University of Trencin, Trencín, Slovakia

V. HAJEK
Department of Electrical Machines, Brno University of Technology, Brno, Czech Republic

Virtual and distance laboratories extend the application area of the web. This leads to an openly integrated environment which facilitates the sharing of not only educational material but also hardware and software resources. This paper investigates distance learning with particular attention to experimental work. DelftWebLab provides the user with a practical experience in Power Electronic education. It was designed based on cutting-edge ideas and had clear targets.

Keywords: distance education; power electronics

INTRODUCTION

DISTANCE LEARNING is promoted across the entire education sector due to the increasing number of people who educate themselves out of their working hours, or as part of their professional development [1–3]. Furthermore, universities and high schools already make extensive use of the internet for communication with their students. This, combined with recent developments with regard to the internet and information technology in general, has seen the need for web-based teaching grow rapidly.

Distance learning via the internet focuses on the delivery of information to the student, typically via web pages that are greatly enhanced by multimedia content. The student, sitting at home in front of the computer, receives lessons in a given subject while maintaining contact with the other students as well as with the teacher via e-mail, chat-rooms, on-line tests, etc. The use of multimedia such as video-clips, audio material or 'slide shows', whether within the classroom or at a distance (via the internet), is an important issue in education. Advanced educational material frequently makes use of interactive programs, in the form of experiments performed via a simulator, for instance. Alternatively, engineering problems can be solved in combination with text explaining the theory.

Rapid changes in both society and technology have created a demand for engineers that are more flexible and that possess more than just a high level of technical or scientific specialization. The drawback of a purely theoretical approach in an undergraduate electrical engineering (EE) curriculum is that less attention is paid to the phenomena that emerge from laboratory experiments and the exploration of system components. The result of this, aided by the rapid development of computer applications, is that hands-on laboratory experience is vanishing and computer simulations are receiving increasingly more attention.

However, it is crucial for students to gain this practical experience. Physical experiments give the student a feeling for practical testing, which can allow them to see the influence of second/higher-order, real-time and even parasitic effects that are often difficult or even impossible to simulate perfectly. The reason for this is that simulations are always based on approximate or simplified models. This makes it important for the student also to have real-world practical experience. However, building an experiment is expensive, making it impossible for an educational institute to cover the complete range of experiments. Hardware experiments should therefore be adapted in such a way that they can also be accessed via the web. In this way, advances in ICT will be combined with the real world. The virtual (distance) laboratory proposed here is not a web-based simulation. It is a real electro-technical experiment conducted in the laboratory, but it is remotely controlled and monitored by web-based tools. It is even possible to visualize the measuring instrument, the electronic components and many more factors, such as lay-out, for example, and this facility is useful in fulfilling today's requirements for teaching over the internet.

Experiments conducted in this way should not only be orientated to analysis (to measure and see the results) but also to synthesis, so at least one

* Accepted 18 January 2007.

design aspect should be included. Therefore, a project should be based on a cutting-edge idea and should have clear targets. We have applied this technology to teaching practical power electronics. First of all, the technology of such an integrated approach is described here, and thereafter guidelines for achieving distance interactive practical education are defined.

PEDAGOGICAL APPROACH

In the pedagogical approach, learners and instructors interact with one another via three subsystems [1]. These three subsystems will be discussed in this section.

Learning subsystem

The main function of DelftWebLab is to provide a web-based remote control for power electronic experiments. The learning process includes several, specially designed, experimental tasks. However, for safety reasons no one will be allowed to perform any experiment until he or she has shown adequate knowledge of the experiment. Entering wrong input parameters due to insufficient knowledge of the experiment may also lead to improper operation of the experiment. Therefore, a learning schedule is designed for students to gain the requisite knowledge before attempting the experiment.

After the completion of an online experiment, learners are given the opportunity to complete a simple questionnaire or alternatively to submit their report through the available feedback subsystem for its final evaluation. This depends on the instructor's requirements. All learning procedures are recorded for future reference and analysis.

Evaluation subsystem

To use DelftWebLab to achieve the desired learning effect, the system first has to assess the learner's prerequisite knowledge of the experiments. The evaluation subsystem assesses this before it permits the student to access the online experiments. Several types of evaluation can be used in this system [2]. The simplest method is to use a questionnaire that only contains 'true or false' questions, single questions, and multiple-choice questions. Instructors may also ask learners to submit simulation results or reports on simulation tasks via e-mail and then evaluate the results manually. Another possible method of online evaluation that is currently being considered is a peer-review method. An experienced learner who has been trained can be assigned as a teaching assistant (TA) for a specific experiment. The TA can then talk to, or correspond with, anyone who requests permission to do that experiment. Once the TA believes that the new learner has adequate knowledge of the experiment, he or she can grant the student access to that experiment. In this way the instructor's workload can be significantly reduced.

Feedback subsystem

A feedback subsystem plays an important role in improving the performance of the learners and the use of DelftWebLab. Feedback to learners often includes the evaluation results and suggestions on learning, while feedback to instructors and supervisors often includes problem reports on the DelftWebLab and questions during the learning process. Peer–student or student–instructor interactions are both significant in this feedback subsystem. In DelftWebLab the authors have developed several feedback mechanisms. Feedback to learners may be provided instantly from predefined functions, or from an instructor or administrator with a certain time delay. E-mail is one of the easiest ways for learners to communicate with instructors. Discussion forums or online chatrooms also provide environments for feedback.

SYSTEM ADMINISTRATION

Experiment administration

Every experiment has its own server, because of the limited control ports available in an ordinary PC and also for easy maintenance. Remote users first log onto a main server, after which they will be directed to the specific server for actually performing the experiment. Before an experiment becomes available online, it should be tested to verify the correctness of the experiment results, as well as the stability of the experimental set-up. Access to the experiments is available 24 hours a day. Supervisors routinely check the status of each experiment to make sure that they are functionally correct and available for use.

Server site administration

The remote experiments offered on the web can also be done locally (i.e. using the control programs on the server without connecting to the internet).

Client site administration

Several clients can connect to DelftWebLab simultaneously. However, internet bandwidth becomes extremely limited when too many remote users request to use the system. Several concurrent remote users via an internet connection are allowed for each experiment. However, each experiment in DelftWebLab can only be operated by a single remote user at a time. The system thus considers each experiment as a 'resource', and remote users who wish to operate a specific experiment should first get permission. Once the resource is in use, other remote users cannot access that resource, because it is then marked as 'locked'. All remote users without access permission can only see the online, real-time video of that experiment.

THE ARCHITECTURE OF THE SYSTEM

The system includes a number of subparts. The principle functionality is showed in Fig. 1. There are several parts to the system. The main part is the 'Measuring and web server'.

Communication with the measurement instrument, control of the power supplies of all the components of DelftWebLab and displaying of the web pages with measured values are all provided by the computer. The measuring instruments that are connected to the server will be described later. The users are divided into groups: authorized users and guests. Only authorized users can control the measurements. Guests can observe measured values but they cannot control any part of the system.

WEB AND MEASURING SERVER

The main part of the system is the web server, which is responsible for all the web services, web pages and the correct functionality of the user interface. The web server also communicates with other applications (Fig. 2) that can access the other parts, such as the measured data. The parts that cooperate with the web server are described below.

Measurement application

Most similar applications use LabView [13]. In our case, the measurement application communicates with measurement instruments via a GPIB interface. This application was written in a Matlab [14] environment and finally compiled/built as an executable application. Matlab has to be installed, together with its 'Instrument Control Toolbox', to provide communication via the GPIB interface. This toolbox is a general programming interface controlling all instruments equipped with a GPIB interface.

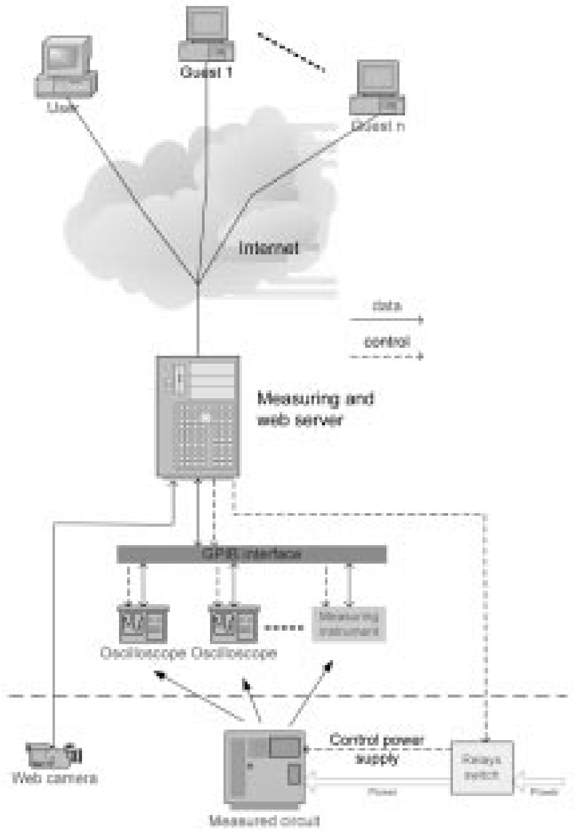

Fig. 1. Structure of distance laboratory.

Control of power part

An application programmed in C++ language controls the on and off switching of the power supply to the measuring instruments and the measuring board. The data bits of a parallel port are used as a control signal. A built-in remote controller controls the power switches by means of radio waves (see Fig. 3).

Compiler of the source code

The source-code compiler for the XC866 microcontroller is also included in this system. Use of the compiler will be explained later. The selected microcontroller is an XC866 from Infineon [15]. This eight-bit microcontroller features a system on chip that has been developed for all kinds of motion control using PWM-based schemes. The microcontroller used in this system is commercially available in an evaluation kit. The PC is connected to this starter kit via the communications port, COM1. After the user sends the source code to the microcontroller, it is compiled and translated to a hexadecimal-executable file and then loaded in the microcontroller. The programming language used for the microcontroller is C. Users can use all the

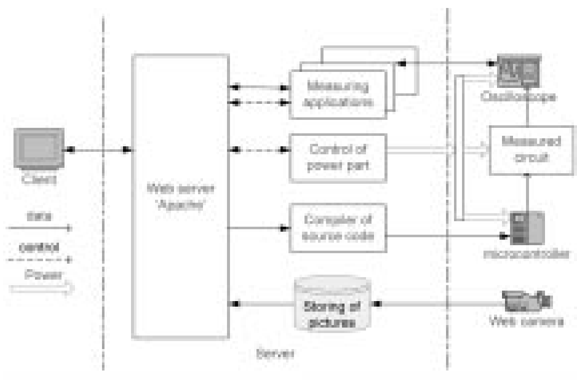

Fig. 2. Server functionality.

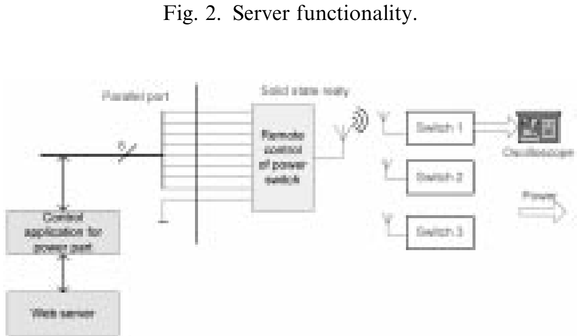

Fig. 3. Power controlling.

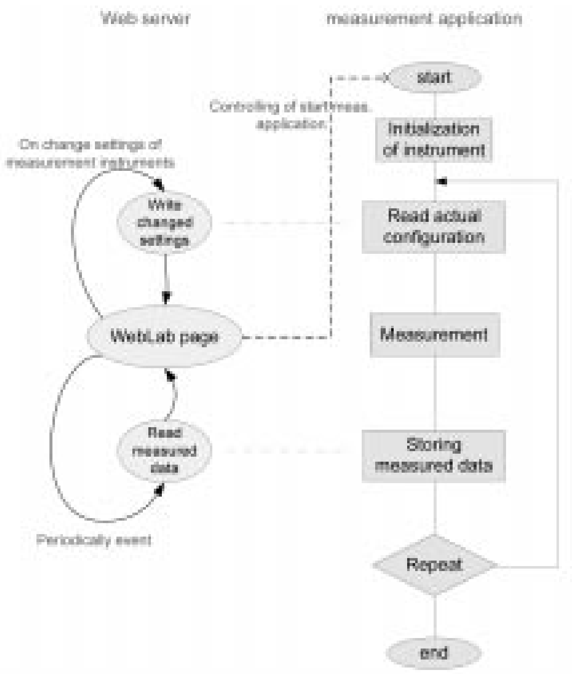

Fig. 4. Cooperation between web server and measuring application.

available features of C language and in addition they can also use the special registers as variables.

Software

The main functional part is the connection between the web server and the measuring applications, which in turn communicate with the measuring devices themselves. In Fig. 4 (left), a simple diagram of the web page is shown. The 'DelftWebLab page' bubble represents the web page with measured data. From this page the user can run the measuring application (dash line). The two arrows pointing away from the 'DelftWebLab page' bubble represent some events which might occur. The first event, namely the 'On change settings of the measurement instruments', occurs when some parameters are changed (e.g. the vertical scale of the scope, etc.). The second event occurs periodically, refreshing the web page to show the latest measured data. On the right side of this figure there is a block diagram of a measurement cycle. The grey lines represent the cooperation between the functionality of the web page and the measuring application. When the measuring application is launched, the program periodically controls the measuring instrument, reads the measured data and stores the measured data to a hard disk.

STRUCTURE OF WEB PAGES

The web pages are divided into three parts (Fig. 5): the set-up page, the measuring page and the viewing page (also measuring page) for guests.

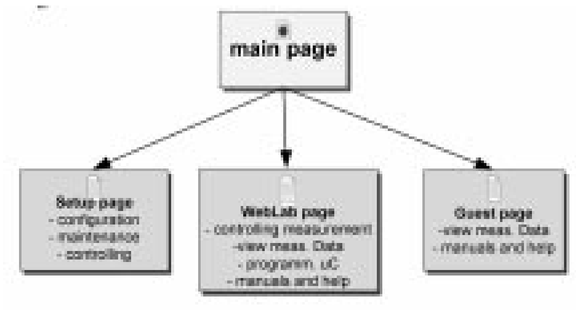

Fig. 5. Structure of the web pages.

Set-up page

The user can set the parameters of the measurement, manage authorized users and change the measured circuit (Fig. 6) on this page. The user needs to have an access code to enter this page.

Measured page

The user can control, in real-time, the measurement instruments as well as program the microcontroller on this page. To avoid any misuse or failure, only one user can be logged on to this page at a time (an access code is required).

Guest page

This page is similar to the measured page, but the user (a guest) cannot control the measuring instruments (the control elements are hidden).

Set-up of the system

In the set-up part, the user can configure some parameters of the measurement itself (time for the measurement and the instruments to be used, for example) as well as control the status of the measurement workstation. There is the option to toggle the hardware part of the measurement on or off (measured circuit, microcontroller, oscilloscope) as well as run the measurement application, as shown in Fig. 6.

There are two types of authorization in the main page: user authorization and administrator authorization. The difference reveals itself on the set-up page. The administrator has full control over all the possibilities (as shown in Fig. 6—the set-up categories are listed on the left side). The rights of the user are restricted; for example (Fig. 7), users do not see the part of the page that is marked with a red ellipse, because they do not have the right to add or delete new users.

In the 'configuration' page there is an option to change the following parameters:

- The time of the measurement. This corresponds to the duration for which the measurement application should run. After this time has elapsed, the application will be closed.
- The type of scope to use. Several types of oscilloscopes can be connected in the measurement. Those supported are Tektronix TDS 3034, 224 and 210 models, for example.

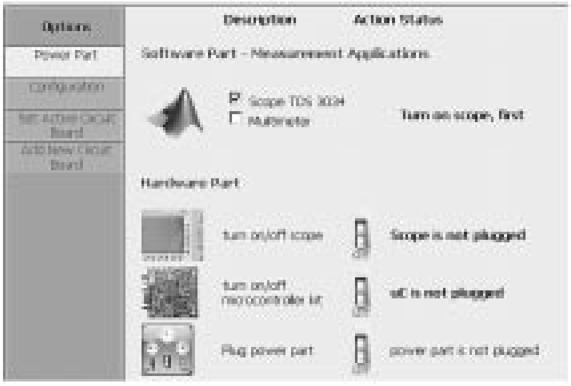

Fig. 6. Set-up of DelftWebLab.

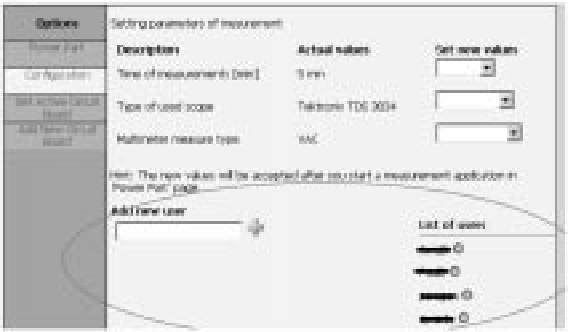

Fig. 7. Set-up of DelftWebLab—configuration.

- The type of multimeter to use. If a multimeter uses a measurement, the user can select the type of measurement (voltage ac/dc, current ac/dc, frequency, etc.).
- Add a new user: only the administrator can add a new user.

In the parts 'set active circuit board' and 'add new circuit board' (only the administrator can make these changes), it is possible to choose one of the prepared measuring boards and to add a new circuit board into the DelftWebLab system.

WebLab page

The WebLab page (Fig. 8) is the user interface of the DelftWebLab system. On this page one can find all the control components required to perform a measurement. These include the status information of the system, a form for sending the programming code to the microcontroller, the circuit diagram with predefined circuit points, control components of the oscilloscope and multimeter as well as measured values and curves.

1. Status information

The status information is displayed at the top of the main page because the user is physically at a different place and is therefore unable to see the real status of the measuring instruments, the measured circuit, etc. Information regarding the power supply of the circuit board under test, the microcontroller, the measuring instrument (oscilloscope) and the status of the measuring application itself (the application is running or not, or it's starting) is provided. This status information (Fig. 9) corresponds to the changes made in the 'power part' in the set-up element of the DelftWebLab (Fig. 6.)

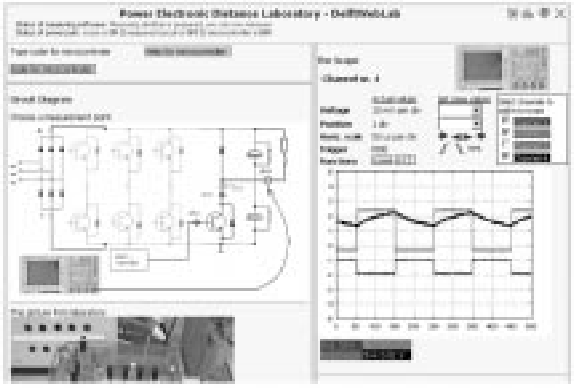

Fig. 8. DelftWebLab system.

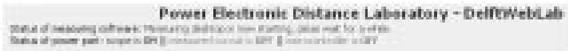

Fig. 9. The status information about the system.

Several states are possible. The power part can only be ON or OFF. The manner in which the power parts are connected is described in the next section. The measuring software has three possible states: 'Measuring desktop is prepared, you can now measure', 'Measuring desktop is now starting, please wait for a while' and 'Measuring desktop is not prepared, first run measurement application'. The second of these is only available because the start of the measuring application is not instantaneous: some time is required to start the application.

2. Code for microcontroller

One of the measurements requires that pulses be created with the microcontroller. For this reason, an option to post code to the microcontroller has been included. The CX866 microcontroller that was used can be programmed in the C programming language. The microcontroller programming guide is provided within the help function of the program. A Keil compiler (uVision) is used for programming the microcontroller, although users do not get to see its window interface. In the system (Fig. 8) only one form (Fig. 10) is available to upload the source code to the microcontroller. When the uploaded code is correct (i.e. there are no errors in the source code), the code is loaded into the microcontroller.

3. Controlling the scope

Measured curves from the scope as well as the control components of the scope are placed on the right side of the WebLab page. The curve to display is chosen by clicking on a marked measured point on the circuit diagram. The other

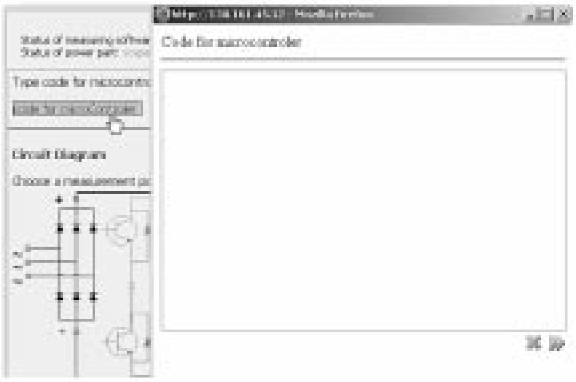

Fig. 10. Form for sending code to the microcontroller.

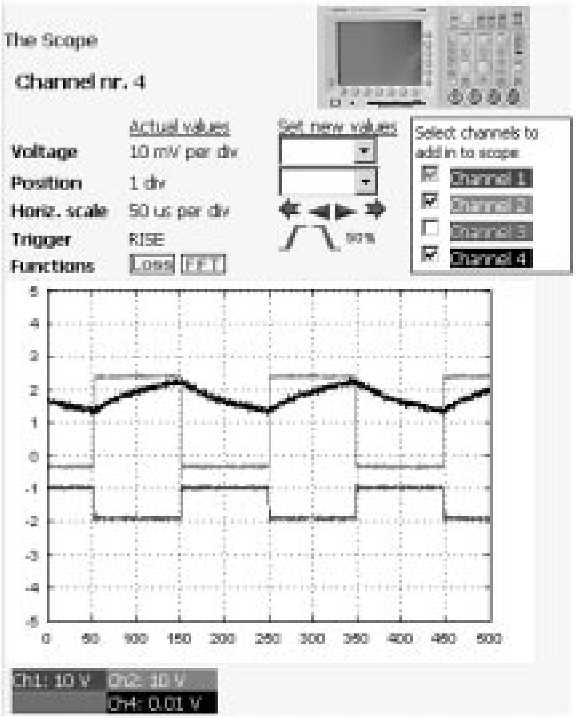

Fig. 11. Controlling the components of the oscilloscope.

option is to click on the color legend above the displayed curves.

The scope page (Fig. 11) allows the user to control the settings of the oscilloscope: the vertical scale, vertical position, horizontal scale, and the trigger. Making changes to these parameters is very easy—it can be done by changing values in combo boxes, or by clicking on the blue arrows. There are two ways of changing the horizontal scale: a big step and a small step in the time scale. The next possibility is the trigger, which can be set up to trigger on the rising edge, the falling edge or on 50% of the measured trace. There is also an option to display the losses (voltage x current) and FFT (Fourier transform). These curves are computed independently of the oscilloscope using Matlab [13]. Fig. 12 illustrates the coherence between the measured curve (sine wave) and its FFT.

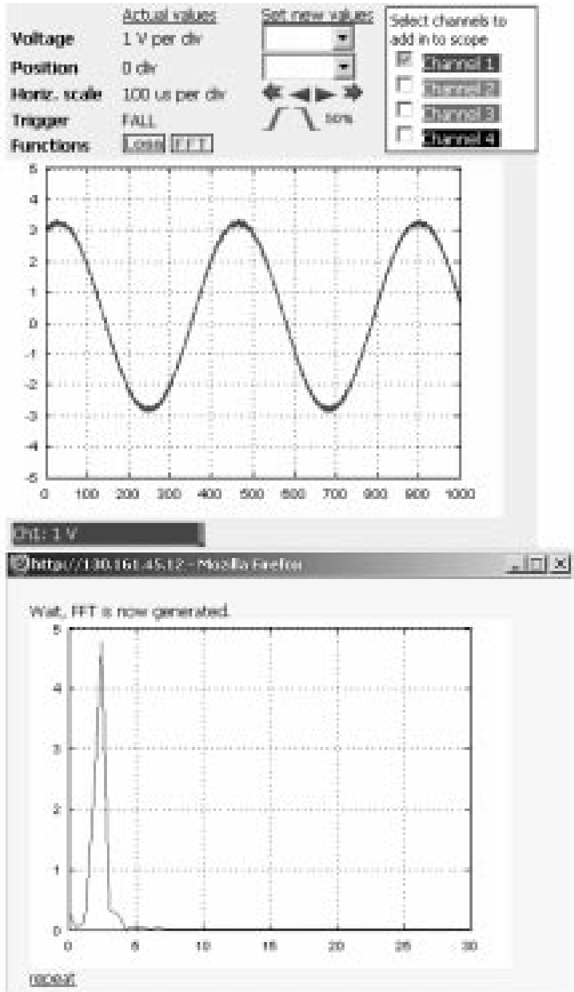

Fig. 12. Sine wave and its FFT.

The last option is for the user to hide or reveal the measured traces, by clicking on the checkbox in the colored legend on the right side of the page (Fig. 11). Only channels that have been checked will be displayed in the graph.

POWER ELECTRONIC EXPERIMENTS

A dc-dc converter and the demonstration of a three-phase inverter with vector modulation were selected as the two systems to be measured using this method.

The objectives are summarized by the following:

- Simulate a typical design process of a converter
- Show the physical layout and construction of a modern converter
- Demonstrate the switching effects of power semiconductor switches (e.g. switching on/off and reverse recovery)
- Demonstrate the real-time effects, delays in the drivers, etc.
- Compare the simulated and measured waveforms; show the influence of the parasitic elements

Design process

The main idea behind this experiment is to simulate a typical design process of a power converter. The design aspect has already been identified as very important part, and is in contrast with the traditional practical test, where the objective is only to observe the various phenomena.

The design process can be characterized by the following steps:

- Simulation of the desired system
- Programming of a microcontroller to generate pulses for the power converter
- Assembling of the breadboard using the available building blocks
- Measuring the real signals of the system and running the breadboard with a self-built (programmed) modulator.
- Comparison of simulated and measured waveforms concerning delays and real-time effects, etc.

Simulation and animation

The aim of the simulation is to get the student acquainted with the system and the working of the converter in preparation for the practical experiment. The component values for the simulated system are the same as will be used later for the breadboard. There is an option to connect the remote simulation tools shown in Fig. 13 to DelftWebLab. A vnc server web interface was used as the connecting application. After clicking on the remote simulation icon, a new page will be displayed prompting the user to enter a password. This password is needed in order to provide a secure connection. (To use this connection, the Java support must be installed on the user's computer.) After connecting to the remote computer, a simulation program can be run. Simulation of the converter system is performed with the simulation program labelled Caspoc [16]. The user-friendly interface allows one to simulate simple examples such as the buck converter. The use of a scope feature is in this case similar to a real scope (Fig. 13).

Performing simulations is an excellent preparation for the measurement itself. There is an option to change the power circuit and its configuration by dragging and dropping the respective components. One of the interesting features of Caspoc is its animation capability: next to the oscilloscope waveforms it is possible to see whether the switch is on or off, or to follow the flow of the current. A change in the control strategy or any of the circuit parameters influences the power flow. This is in spite of the fact that the simulated waveforms are somewhat idealized. In fact, it can be considered an advantage that the simulated waveforms are measured without parasitic and measurement noise hindrance, for comparison with the real-time effects will be made later during the actual measurement.

Animation can also be prepared in DelftWebLab. The animation page is prepared by clicking on the question mark located in the navigation panel (Fig. 8). Several case studies have been prepared.

The prepared measurements in DelftWebLab system

In the DelftWebLab system, three complex measurements have been prepared (see Fig. 14). These are the three basic conversion options, namely ac-dc, dc-dc and dc-ac.

Only the first application measurements, namely with the dc-dc converter, are described here. The dc-dc buck converter was selected (step down chopper) with a resistive load which will later be extended to a dc machine (Fig. 15). This converter topology is selected because it forms a basic building block for a voltage source inverter. This practical set-up of a buck converter is therefore extended to a switching leg of an inverter in the final part.

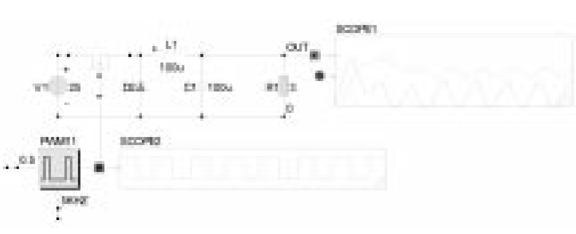

Fig. 13. Caspoc simulation scheme of a buck converter.

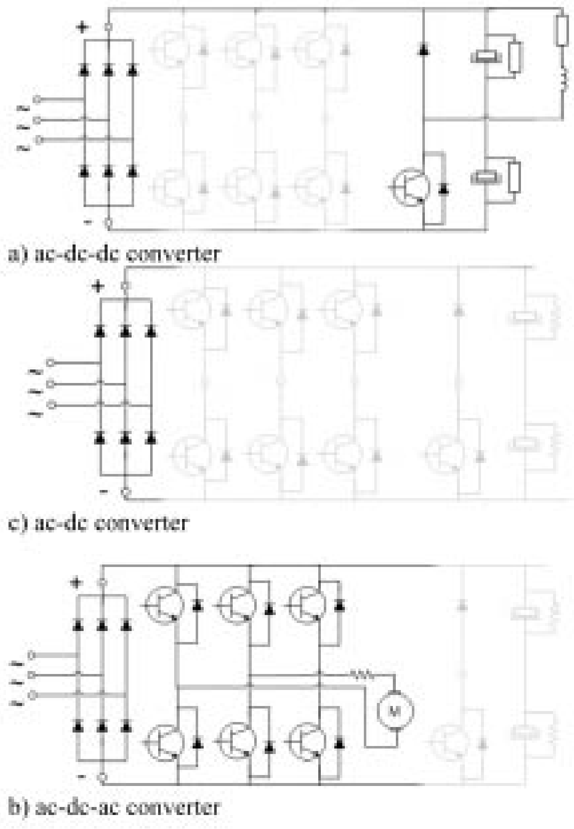

Fig. 14. The prepared measurements in DelftWebLab.

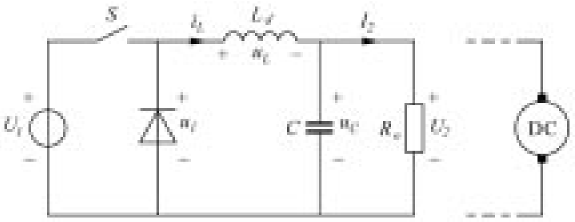

Fig. 15. Buck converter with R load and DC machine.

The measurement process has two parts:

a. Programming the microcontroller

To generate the pulses, the capture/compare unit (CCU) of the XC866 microcontroller is used. Every register is divided into two parts: the lower word and upper word (see Fig. 16), because the CCU works in a 16-bit mode. The program shown below generates pulses at a frequency of 5kHz.

b. Measurement of the buck converter

After the microcontroller has been successfully programmed, the measurement of the buck converter can commence. On the circuit diagram several measurement points (colored circles) have been prepared. The filled circles represent voltage values, and the clear circles represent current values.

```
CCU6_PAGE=1;
CCU6_TCTR0L    =  0x01;   // setting of timer, frequency, ...
CCU6_T12PRL    =  0x00;    // setting period value (0x0 ... 0xFFFF)
CCU6_T12PRH    =  0x0f;
CCU6_PAGE=0;
CCU6_CC60SRL   =  0x00;  // duty ratio
CCU6_CC60SRH   =  0x05;
CCU6_TCTR4L    =  0x42;  // load CCU6 timer control register 4 low
CCU6_TCTR4H    =  0x00;  // load CCU6 timer control register 4 high
CCU6_PAGE=2;
CCU6_MODCTRL   =  0x03;  // multi channel mode is disabled
CCU6_MODCTRH   =  0x00;
CCU6_T12MSELL  =  0x03;  // load CCU6 T12 camptura/compare mode
CCU6_T12MSELH  =  0x00;
PORT_PAGE=2;
P3_ALTSEL0      = 0x03;      // select pins 0 and 1 as output
P3_ALTSEL1      = 0x00;
PORT_PAGE=0;
P3_DIR          = 0xff;         // set P3 as OutPut
CCU6_TCTR4L   |=  0x02;         // start timer t2
```

Fig. 16. Listing of sample program.

The measured points of the dc-dc buck converter are shown in Fig. 8. They are the voltage over the switching device, over the gate driver, and the current through the load. The measured results are also shown in Fig. 8. Comparing the simulation and the real-time aspects measured on the system is an important stage. The delays in the control electronics, switching operation and other important issues can be studied in detail at this stage. Finally, the buck converter is loaded with a DC machine and the speed of this machine is controlled in an open-loop fashion using the calculated duty cycle.

CONCLUSIONS

Technology for establishing a distance laboratory is proposed here and applied to a practical power electronic experiment. It provides access to a practical laboratory experiment for a larger group of students. Furthermore, it allows the students to perform the experiments independently and safely. Official working hours are no longer a limitation for using the laboratory. The students can still experience the appearance of the measurement instrument and the electronic components themselves as well as many more factors, such as lay-out, for example. This facility is useful for fulfilling today's requirements for teaching over the internet. The design process followed during this power electronics practical gives the students confidence in their diploma work phase of study. It also provides insight into the standard design steps usually taken during the research of power electronics.

Acknowledgements—This work was performed within the project 'E-learning Distance Interactive Practical Education (EDIPE)'. The project was supported by the European Community within the framework of the Leonardo da Vinci II programme (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 do they involve any responsibility on its part. Research described in the paper was also financially supported by the Ministry of Education of the Czech Republic, under the project MSM 0021630516, and the project of the Grant Agency CR No. 102/06/1320.

REFERENCES

1. G. W. Chang, Z. M Yeh, H. M. Chang and S. Y. Pan, Teaching photonics laboratory using remote control Web technologies, *IEEE Trans. On Eduction,* **A247** (April 1955), pp. 529–551.
2. B. Szabados, Interactive outcome based assessment using multimedia, *International Journal of Engineering Education*, **20**(2) (2004), pp. 141–151.
3. P. Penfield and R. C. Larson, Education via advanced technologies, *IEEE Trans. On Eduction,* **39**(3) (April 1996), pp. 436–442.
4. C. Fernandez, O. Garcia, J. A. Cobos and J. Uceda, *Self-learning Laboratory Set up for Teaching Power Electronics Combining Simulations and Measurements*.
5. K. W. E. Cheng, C. L. Chan, N. C. Cheung and D. Sutanto, *Virtual Laboratory Development for Teaching Power Electronics*, EPE-PEMC, Riga, Latvia (2004).
6. Z. Yi, J. Jian-jun and F. S. Chun, A LabVIEW-based interactive virtual laboratory for electronic engineering education, *International Journal of Engineering Education*, **21**(1) (2005), pp. 94–103.
7. A. B. Buckman, VI-based introductory electrical engineering laboratory course, *International Journal of Engineering Education*, **16**(3) (2000), pp. 212–217.

8. C. S. Peek, O. D. Crisalle, S. Deapraz and D. Gillet, The virtual control laboratory paradigm: Architectural design requirements and realization through a DC-motor example, *International Journal of Engineering Education*, **21**(6) (2005), pp. 1134–1147.
9. C. Fernandez, M. A. Vicente and L. M. Jimenez, Virtual laboratories for control education: A combined methodology, *International Journal of Engineering Education*, **21**(6) (2005), pp. 1059–1067.
10. N. Ertugrul, New era in engineering experiments: An integrated and interactive teaching/learning approach, and real-time visualisations, *International Journal of Engineering Education*, **14**(5) (1998), pp. 344–355.]
11. V. G. Agelidis, A laboratory-supported power electronics and related technologies undergraduate curriculum for aerospace engineering students, *International Journal of Engineering Education*, **21**(6) (2005), pp. 1177–1188.
12. P. H. Gregson and T. A. Little, Designing contests for teaching electrical engineering design, *International Journal of Engineering Education*, **14**(5) (1998), pp. 367–374.
13. N. Ertugrul, Towards virtual laboratories: A survey of LabVIEW-based teaching/learning tools and future trends, *International Journal of Engineering Education*, **16**(3) (2000), pp. 171–180.
14. Matlab, www.mathwors.com.
15. Inineon, http://www.infineon.com.
16. Caspoc, www.caspoc.com.
17. V. Hrabovcová, P. Rafajdus, M. Franko and P. Hudák, *Measurement and Modelling of Electrical Machinery*, in Slovak, ŽU Žilina, Slovakia (2004).
18. R. Hartanský, *Electronics*, in Slovak, 1st edition, TnUAD, Trencín, Slovakia (2006).
19. P. Manas, Technical simulation in the Czech Engineer Corps, Proceeding of 1st International Scientific and Technological Conference (4. 5. 2006), Bratislava, Slovakia, pp. 173–176.

Pavol Bauer received his Masters in Electrical Engineering at the Technical University of Kosice (1985) and his Ph.D. from Delft University of Technology (1995). From 1985 to 1990 he was with TU Kosice. Since 1990 he has been with Delft University of Technology, Netherlands, teaching power electronics, electrical drives and related subjects. From 2002 to 2003 he was partly working at KEMA (Arnhem) on different projects. Mr. Bauer has published over 180 conference and journal papers in his field, he holds international patents and has organized several tutorials at international conferences. He has worked on many projects for industry concerning wind power, and power electronic applications for power systems such as Smarttrafo, etc., and participated in several Leonado da Vinci EU projects. He is a member of IEEE, EPE and also a member of international steering committees at different conferences.

Juraj Dudak was born on 29 January 29 1980 in Trencin, Slovak Republic. He received an M.Sc. degree in mechatronics from Alexander Dubcek University in Trencin (TnU), Slovak Republic, in 2003. He is currently working towards a Ph.D degree at the Slovak Technical University, Bratislava, Slovak Republic. Since 2003, he has been working at the department of informatics, Faculty of Mechatronics (TnU), as an assistant lecturer. His research interests include new methods in network communication.

Dusan Maga was born in 1968 in Nitra, Slovakia. His finished his university education in power electronics in Bratislava (M.Sc. 1991, Ph.D. 1998). He became an associate professor at the University of Zilina, Slovakia, in 2002. Since 1998 he has worked at Alexander Dubcek University of Trencin, Slovakia (from 2004 as a dean of Faculty of Mechatronics). His professional interests are numerical methods of field solutions (especially finite and boundary elements) and distance computer-based education in electric machinery. He has authored approximately 100 world-wide publications (including three monographs) with more than 75 references.

Vítězslav Hájek graduated in 1962 with an M.Sc. in heavy current electrotechnics—electrical machines and drives at TU in Brno. From 1962 to 1982 he was a scientist with Magneton Kroměříž. He has been working at TU Brno since 1982 (since 2001 as a professor at the Department of Electrical Machines and Apparatus). His research interests are in electric machines, apparatus and drives, automotive electrical machines, and automatic measurement of electrical machines.