Development of Remotely Accessible Matlab/Simulink Based Electrical Drive Experiments

Seta Bogosyan
s.bogosyan@uaf.edu
University of Alaska Fairbanks, Electrical &Computer Engineering Dept, Duckering Building, Fairbanks, AK 99775, USA

Metin Gokasan
gokasan@itu.edu.tr
Istanbul Technical University, Electrical & Electronics Eng. Faculty, Maslak, Istanbul, 34469, Turkey

Ali Turan
ftat@uaf.edu
University of Alaska Fairbanks, Electrical &Computer Engineering Dept, Duckering Building, Fairbanks, AK 99775, USA

Richard W. Wies
ffrww@uaf.edu
University of Alaska Fairbanks, Electrical &Computer Engineering Dept, Duckering Building, Fairbanks, AK 99775, USA

Abstract: This paper describes the development of remotely accessible electrical drive sets for hands-on education in electrical machinery, electrical drive and control courses as well as certain aspects of motion control and power electronics. The remote access is provided via the developed client-server communication method using the TCP/IP protocol to run the Matlab/Simulink compatible electrical drive sets operating on the remote server side. The developed method allows access to the robotics and electrical drive test-beds, namele permanent magnet DC (PMDC), permanent magnet synchronous (PMSM), and induction motor (IM) motor-load sets, developed for both on-site and remote use at the Control Laboratory in the University of Alaska Fairbanks (UAF). The client-server communication is developed in C/C++ using wxWidgets to communicate with the Matlab/Simulink downloadable DS1104, which is the control unit used in all the electrical drive systems in the lab. The UAF remote laboratory web site can be accessed at: http://www.uaf.edu/ece/remote-lab.

Keywords: Remotely accessible electrical drive experiments, client-server communication, Matlab/Simulink.

I-Introduction

Remotely accessible laboratories are gaining increasing popularity all over the web due to the significant sharing of resources they promise for costly equipment and the flexibility they provide in terms of time and variety.

Web experiments are mostly based on existing software platforms, such as LABView or Matlab. Among several such examples in the literature and on the web, [1] presents an application on electrical machinery and electrical drive systems. In [2] a remote power electronics lab is discussed with no practical application examples. [3] presents remote DC-DC converter experiments for power electronics education. [4] presents the real-time control of the AdeptCobra 600, a 4-axis SCARA robot, via the internet In this paper, the LABView software is used to establish communication between client and server and AdeptWindows software through TCP/IP; [5] and [6] present web-based laboratory experiments using the Wincon/Simulink package via the TCP/IP protocol; [7] discusses different installations of robotics systems and philosophy of remote robotic laboratories; [8] presents a remotely controllable 4-axis robot using LABView; [9] describes a project that enables students to work on a 5 degree-of-freedom (DOF) robot manipulator from a remote site and [15] presents a remotely accessible hardware-in-the-loop simulator (HILS) lab. In [2], which is another application for remote labs, power-electronic converter experiments are developed through the internet. The user can remotely control the experiments and get graphical results of the measurements. More recently, [10] presents the development of an off-line and a real-time system for remote experimentation.

This paper describes the development of remotely accessible electrical drive experiments and several case studies performed using the system. The developed client-server communication method allows the remote user to run Matlab/Simulink based experiments on the DS1104 board, which runs the electrical drive systems. The setups could be used for hands-on education in electrical machines, electrical drives, power electronics, and in control systems courses.

The client-server communication is established with the RemoteHIL_Lab software developed using TCP/IP with sockets. The main program is written in C/C++ with the use of wxWidgets. The Matlab Engine is used to establish a link between Matlab and the C/C++ code. For data capturing, a code is written in Python, which is compatible with the ControlDesk, used for signal monitoring and tuning.

The paper is organized as follows; Section I gives an introduction to the study; Study II Section III Section IV. Finally, Section V gives the conclusion of the study.

II-Electrical Drive Experimental Setups

At the UAF Power and Control Lab, there are currently three electrical drive control sets [11], which have been designed specifically for educational experiments on AC and DC motors; namely, permanent-magnet synchronous motors (PMSMs) and induction motors (IMs) and permanent-magnet DC machines (PMDC). The drive boards include two independent three-phase PWM inverters to supply for AC or DC motors and their mechanical loads using 42V DC bus voltage, digital PWM input channels for real time digital control, and complete digital/analog interface with the dSPACE board. A PMDC generator is used for the mechanical load.

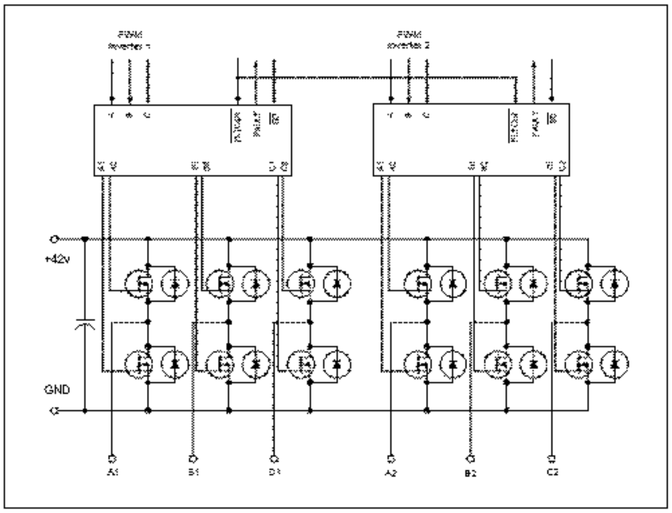

Fig.1 Electrical drive experimental set drive board

Power MOSFETs are used as power switching devices in each of the three-phase inverters. The three-phase outputs of the first inverter are marked A1, B1, and C1 and those of the second inverter are marked A2, B2, and C2 as given in Fig. 1. A signal supply of ± 12V is required to isolate the analog signal outputs from the drive board. This is obtained from an isolated power supply.

For voltage measurements, test points are provided to observe the inverter output voltages. For current measurements, LEM sensors are used to measure the output current of the inverters. The MOSFETS are driven by three-phase bridge drivers (IR2133) in the inverters. The PWM inputs are isolated before being fed to the drivers. PWM and other digital signals for the board are supplied to the 37-pin DSUB connector.

The Drive Boards contain over-current protection for each inverter.

III-Development of Client-Server Communication for Matlab/Simulink Based System, [12], [13]

The code to establish communication between the client and server is written in C/C++ with the use of wxWidgets, which is an open-source toolkit facilitating graphical user interface (GUI) applications in C/C++. Two programs are developed for communication; -one which runs on the server and one, which runs on the client computer- with wxWidgets sockets establishing the communication between the two programs. The protocol used in this study is TCP/IP for the sequential and error-free transmission of data.

The developed server program binds to the local address via the TCP port.Then, the server sets up an event handler, which is informed when a client connects to the server. The server waits in listening status, which means that the server is ready to accept connections [14]. Only one client can connect to the server at a time.

Remote users have three main experiment choices: PMDC, PMSM or IM experiments. On each of those test-beds, users are given options for two experiment types:

a) Designing the full control system and electrical drive, provided they own a Matlab/Simulink license. The user is provided with a template prepared in Simulink, which he/she must download from the RemoteLab library including controller and PWM producing blocks. Next, the remote user places the control algorithm that he/she has developed in Matlab/Simulink into the designated blocks in the provided template file. In the next step, the user makes the proper experiment choice from the Experiments menu, selects the modified template file from the file selection menu and uploads it to the server.

b) Experimentation by changing controller parameters, reference position or frequency, in which case the user only experiments with the provided controls.

In the codes developed for communication in this study, the files and data are directly written to the buffer and sent to the client computer. When a file is transferred from the client to the server, at first a flag is set on both sides. When this flag is set, no I/O will be returned unless all the data has been written or read. Next, the whole file is read into a buffer on the client side and the file is written from the buffer to the socket, which is then received by the server and read by a buffer on the server side.

After receiving the DC or AC motor type information, as well as controller, reference trajectory and parameter data, the Server program calls the Matlab Engine to invoke the Matlab. The Matlab Engine consists of a set of routines which invoke the Matlab from C or Fortran code, so any Matlab command can be used in the C code by using the Matlab Engine.

For all experiment types, the Simulink file is built and downloaded into the DS1104. The simulations can now be started for both options.

Data Capturing

Variables measured and calculated during the simulation process should first be saved in the server in order to send the client for further processing. The ControlDesk, which is again a dSpace product, is used for data tuning and monitoring. For the remote utilization of the chosen experimental setup, the ControlDesk access must be automated. All the library files are developed to be compatible with the Python programming language by dSpace; thus, in this study the data capturing code is written in Python, which is an open source scripting language.

The Python code, at first initializes the ControlDesk, and then checks the current working directory. After that, Simulink file that is created at the end of the build process is loaded. In the next step, ControlDesk layout is prepared, and then data capturing is activated. The Python code is developed to have the ControlDesk plot and capture the data in the server. A selected signal triggers the data capturing process with its raising or falling edge that is determined in the Python code.

The data capturing continues for a duration determined in the Python code. The procedure is finalized at the end of that period, upon which ControlDesk layout is removed.

For automated use, the library files are prepared to be compatible with Python 2.2 by dSPACE company. The scripts prepared with later versions of Python can be transformed to the executable file format (.exe) by py2.exe software, however py2exe does not support the Python 2.2. Thus a batch file is written to invoke the Python script. To start data capturing, the batch file is invoked in the Matlab command line, which in turn, invokes the Python script. All the captured data is saved in a .mat file as a structure data type. Structures are Matlab arrays, fields of which can contain any kind of data.

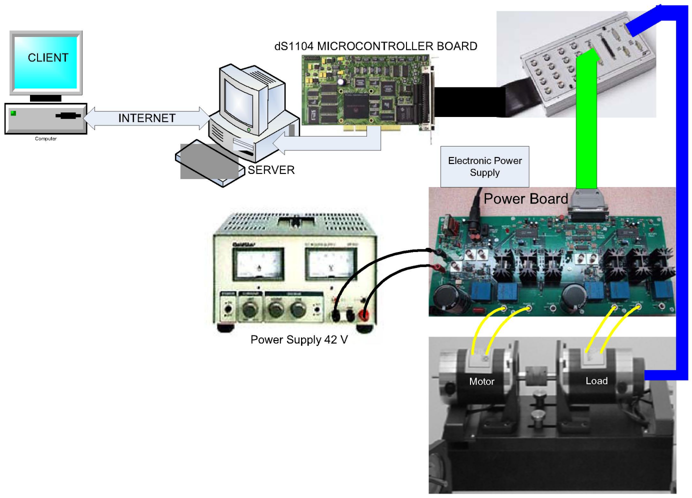

Fig. 2 General overview of the experimental setups

IV. CASE STUDIES

In this section, case studies are presented for both experiment types, allowing the user a capacity limited to controller parameter changes, as well as full capacity in which the user designs and downloads the full control of the system.

a) Remote Control Experiments with PMDC Drive

The first case study is related to the PID control of the PMDC drive, for which the remote user enters the desired trajectory and the PID coefficients. Figure 3 depicts the developed Simulink blocks in the RemoteLab library for the PID control of the PMDC motor-load pair.

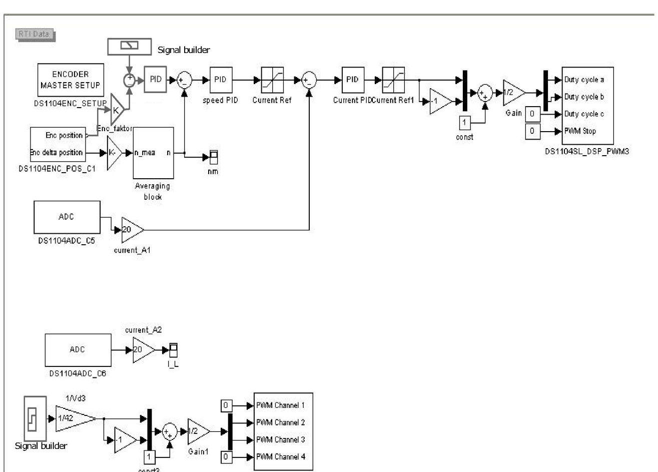

Fig. 3 Simulink blocks for the position control of DC motor&load

The reference input of the DC motor is designed using the Signal Builder block, as can be seen in Fig. 3. The torque input for the DDM is also applied from the Signal Builder block. These signals, which are reference position and load torque, are shown in Fig. 4 and Fig. 5. Also to eliminate high frequency noise, filters are used for the speed.

To demonstrate the effect of the load, the experiment is started initially with no load. The results of this experiment are shown in Figs. 6, 7, 8, and 9, which depict the angular position, error in the angular position, speed variation, current variation of the PMDC and detailed current variation of the PMDC, respectively.

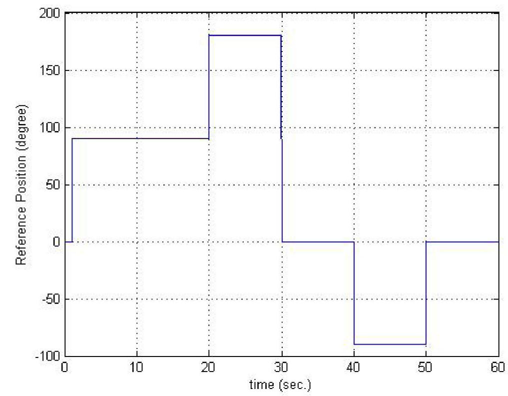

Fig. 4 Reference position trajectory of the DC motor

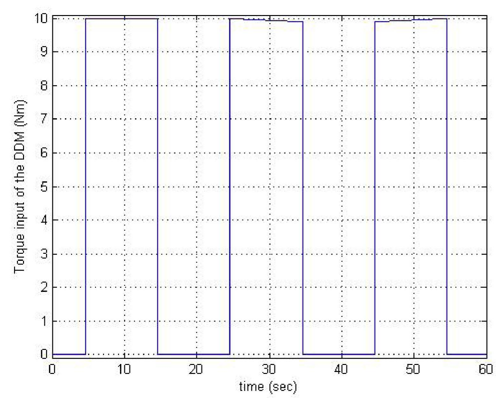

Fig. 5 Torque reference of the DDM

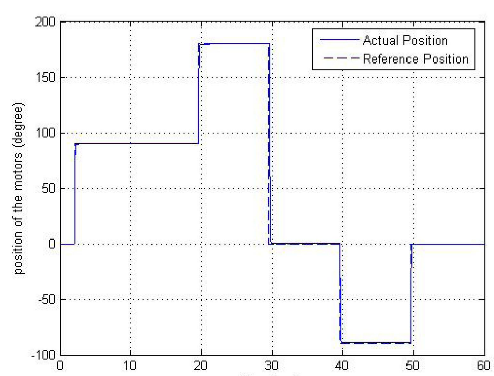

Fig. 6 DC motor: reference and actual angular position variations

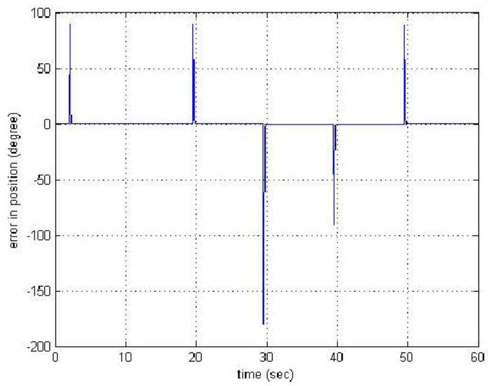

Fig. 7 DC motor: variation of position error

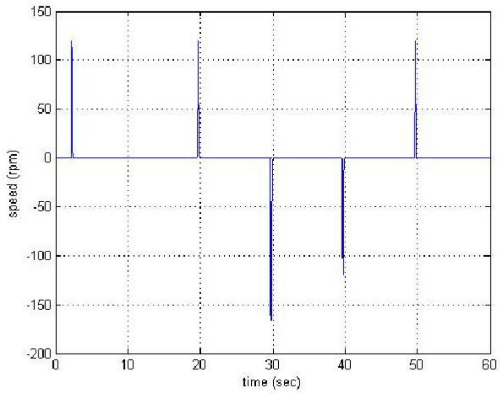

Fig. 8 DC motor: speed variation

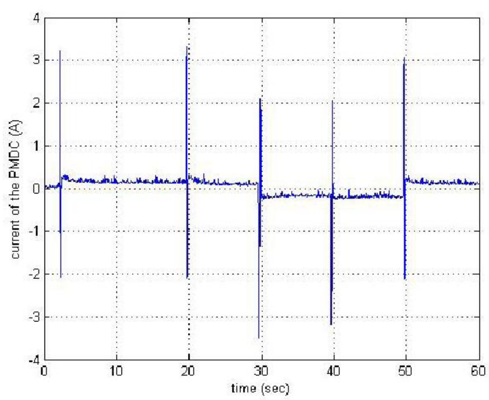

Fig. 9 DC motor: current variation

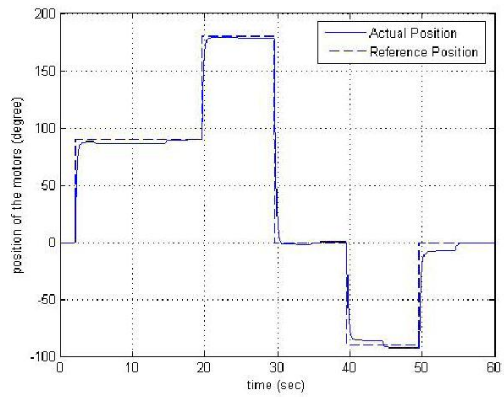

Fig. 10 DC motor: reference and actual angular position variations

The reference and actual angular position variations are given in Fig.10, while Fig.11 depicts the position error. Finally, the speed variation is given in Fig. 12, while the current, which demonstrates a proportional variation to that of the load torque, is depicted in Fig. 13.

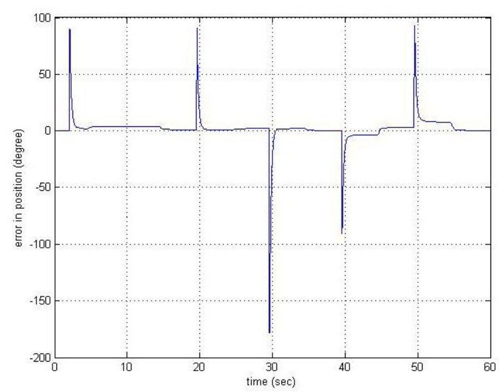

Fig. 11 DC motor: variation of position error

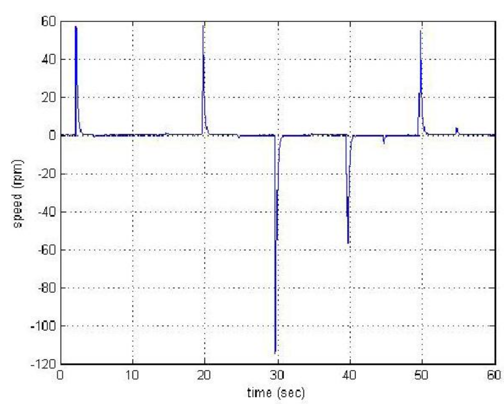

Fig. 12 DC motor: speed variation

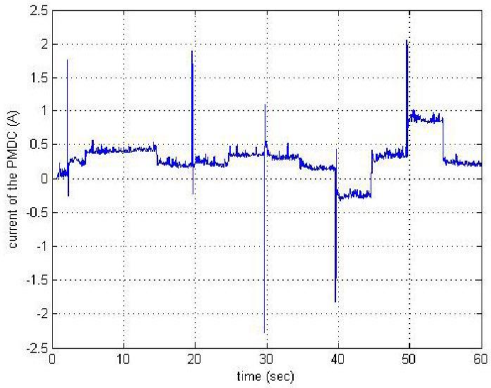

Fig. 13 DC motor: current variation

B) Remote Control Experiments with Induction Motor Drive

This case study involves the remote control of the induction motor drive, for which the full control system is developed using template Simulink block diagram given in Fig. 14. For the bi-directional open loop control of velocity, a V/f control is implemented with a trapezoidal frequency variation. Fig. 15 depicts the phase currents, phase voltage reference values applied to the PWM generator, and the corresponding variation of velocity. Fig.16 gives the line-to-line voltage applied to the motor terminals at 60 Hz, which is derived by transferring the digital oscilloscope output to the server.

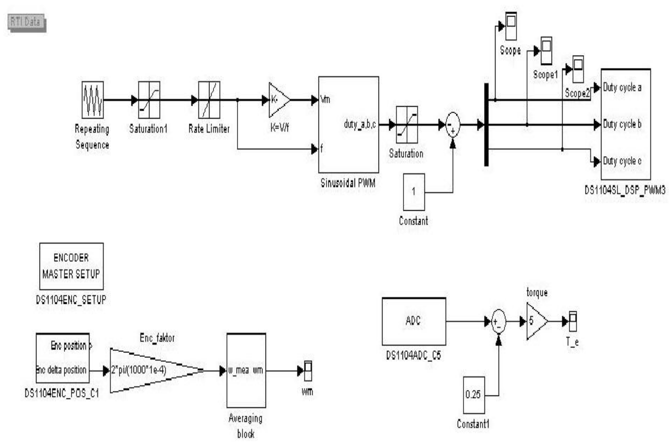

Fig 14 Simulink block diagram for IM control

V. Conclusion

The motor/drive systems originally designed for on-site education as described in [14] include AC and DC motors, namely a combination of PMDC, PMSM and induction motors and DC load generators. The design also allows a flexible use, in that the test-beds can be reversed and motors and generators can be switched. The motors are run by inverters and DC-DC converters, controlled by MATLAB/SIMULINK compatible, high performance floating-point DSP motion control units. The UAF Power and Control Lab currently has three such sets which are used in undergraduate electrical machinery and power electronics courses for experiments related to the control of electrical drives, the derivation of load/no-load characteristics of the AC/ DC motors, for power electronics circuits experiments, generation and control of PWM circuits and also as hardware-in-the-loop (HIL) simulator for a variety of electromechanical systems, i.e, robots etc.,with the generation of various mechanical load types on the generator side.

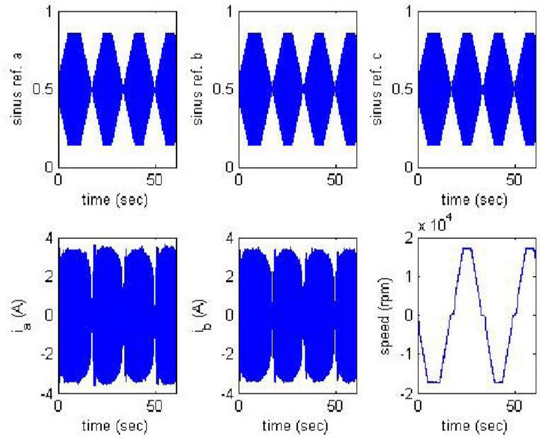

Fig 15 Sinusoidal voltage references for the derivation of PWM waveform, currents of *a* and *b* phases, and variation of velocity.

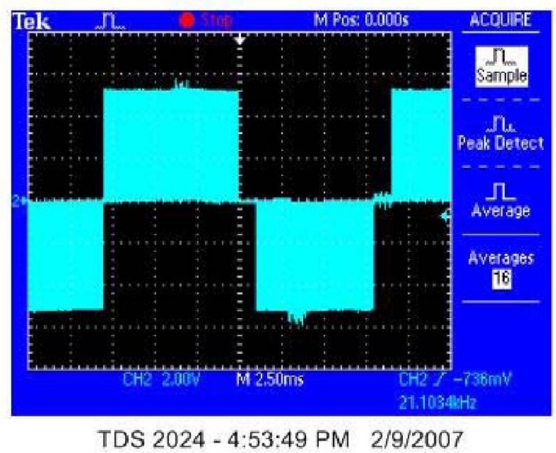

Fig 16 Voltage waveform applied to motor phases

This flexible group of electrical drive test-beds is currently available also for distance education at UAF. The developed internet based client-server communication system provides access to the Matlab/Simulink controlled test-beds, allowing remote users a capacity limited to controller parameter changes, as well as an increased intervention capacity, in which the user designs and downloads the full control of the system.

While distance education can be delivered quite efficiently in most disciplines, engineering education remains hindered in this aspect, due to the strong need for hands-on experimentation to complement the somewhat rigorous and theoretical courses. Systems such as the above could address the need quite effectively, while also increasing student interest in the essential fields of electrical machinery, drives, control and power electronics.

References

[1] N. Ertugrul, Labview for electric circuits, machines, drives, and laboratories, Prentice Hall, 2002.

[2] Fernandes, R.M.; Rolim, L.G.B.; Suemitsu, W.I.; "Design and implementation of a power-electronic remote-laboratory (ELEPOT-rLab)", IEEE Industrial Electronics, 2003, ISIE '03. 2003 IEEE, 9-11 June 2003, pp. 307 – 311.

[3] A. Nogueiras, A. Lago, L. Eguizabal, J. Marcos, and C. Martinez-Pefialver, "Power Electronics Education: Lectures, Simulation and Laboratory Synergetic Approach" Computer as a Tool, 2005. EUROCON 2005.The International Conference, Volume 1, 21-24 Nov. 2005, pp. 827 – 830.

[4] T. Chang, P. Jaroonsiriphan, M. Bernhardt, P. Ludden, "Web-based command shaping of cobra 600 robot with a swinging load", IEEE Trans. Industrial Informatics, vol. 2, pp.59-69, 2006.

[5] T.F. Junge and C. Schmid, "Web-based Remote Experimentation Using a Laboratory Scale Optical Tracker," In Proc. American Control Conf., Chicago IL, June 28-30, 2000, vol. 4, pp. 2951-2954.

[6] C. Schmid and A. Ali, "Web-based System for Control Engineering Education," In Proc. American Control Conf., Chicago IL, June 28-30, 2000, vol. 5, pp. 3463-3467.

[7] A. Bicchi, A. Copelli, F. Quatro, L. Rizzo, F. Turchi, A. Balestrino, "Breaking the lab's walls. Tele-laboratories at the University of Pisa", Proceedings – IEEE Int. Conf. on Robotics and Automation, Seoul, Korea, May 21-26 2001, v2, p 1903-1908.

[8] E. Cooney and A. Shiver, "Remote Control of a Robot Using LabVIEW and World Wide Web," Proc. of the 2001 American Society for Eng. Education Annual Conference & Exposition, Session 2526.

[9] A.C. Reinaldo, C. Bianchi, A. La Neve, "The Influence of Final Course Projects on The Early Engineer's Professional Life," Int. Conf. on Engineering Education, Aug. 6-10, 2001, Oslo, Norway.

[10] P. E. Pace, B. H. Nishimura, W.H. Morris, R.E. Surratt, "Effectiveness calculations in captive-carry HIL missile simulator experiments", Aerospace & Electronics System, IEEE Transaction on Volume 34, Issue 1, Jan. 1998, pp. 124 – 136.

[11] "DSP Based Electric Drives Laboratory User Manual", Department of Electrical and Computer Engineering University of Minnesota Revised: July 27, 2004

[12] A. Turan, S. Bogosyan, M. Gokasan, "Development of a Client-Server Communication Method for Matlab/Simulink Based Remote Robotics Experiments," IEEE International Symposium of Industrial Electronics, ISIE 2006, Montreal, Canada, 9-13 July, 2006.

[13] T. Singaraju, A. Turan, M. Gokasan and S. Bogosyan, "Hardware-in-the-loop simulation of PUMA 560 via internet," The 32nd Annual Conference of the IEEE Industrial Electronics Society, Conservatoire National des Arts & Metiers, Paris, France, November 7-10, 2006.

[14] http://www.wxwidgets.org/manuals/2.6.3/wx_samples.html#samplesocket

[15] H. Temeltas, S. Bogosyan, M. Gokasan, "Hardware in the loop robot simulators for on-site and remote education in robotics", International Journal of Engineering Education, Vol. 22, No.4, 2006, pp. 815-828.