80C537 Microcontroller Remote Lab for E-Learning Teaching

M. Gilibert[1]; J. Picazo[1]; M. Auer[2], A. Pester[2]; J. Cusidó[1]; J. A. Ortega[1]
[1]Technical University of Catalonia, [2]Carinthia University of Applied Sciences

***Abstract*—A remote microcontroller lab based on an 80C537 mock-up is presented. This paper presents a new way to interact with these kinds of systems via the Internet, giving the possibility for complete interaction. The Citrix Application Server is the platform which manages the debugging microcontroller software, and National Instruments LabVIEW 8 is used to develop the applications which allow the interaction between the system and the remote user. By means of LabVIEW web server, full remote interaction has been achieved. Our results suggest a new way to practise in laboratories without being present, yet retaining the feel of an actual laboratory environment.**

***Index Terms*— Remote labs, e-learning, microcontroller courses.**

I. Introduction

A microcontroller is one of the basic subjects in all electronic engineering disciplines. The best way to introduce the content of this topic to undergraduate students is by means of practical exercises, avoiding great theoretical explanations on the blackboard.

The students need to work with teaching kits in order to assimilate the basics of programming microprocessor systems [1]. The usual method implies that the student has to be present at the laboratory, where the microcontroller mock-up is physically located. We call this part of the course where the student and lecturer are actually present onsite the presence learning phase.

The trend in higher education is to facilitate the work to the students avoiding displacements as much as possible, transforming the traditional teaching in a semi-present or totally non-physical teaching environment [2]. Today this is well-solved for the subjects that don't need lab work, using internet applications [3], but additional work is necessary in areas such as microcontrollers [4] [5].

On the other hand, the tendency in higher educational programs is to move from traditional teaching methodologies to new ones, such as Problem Based Learning (PBL). This methodology proposes a practical problem to students and sets up a learning environment to help them to find the proper problem solution on there's own initiative including the necessary theory. It has been proved that this kind of methodology increase the interest of students in the learning process [6].

To achieve a complete e-learning teaching experience and for satisfactory implementation of PBL methodologies, a remote lab becomes mandatory. Students need to practice, test and do exercises. In this paper, the development of such a microcontroller remote lab is presented.

II. Presence learning work bench

The microprocessor remote lab will be developed using as starting point an existing work bench for presence learning and teaching. This is divided in two main parts; hardware which is composed by the microcontroller board µdee537, and the Keil µVision2 as a software development, running on a PC. Figure 1 shows the block diagram of the presence teaching work bench.

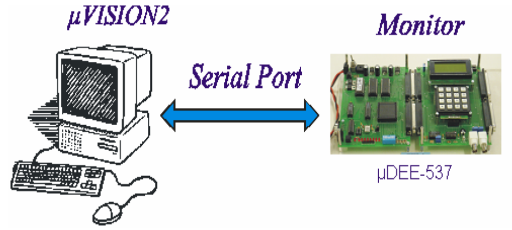

Figure 1. Microprocessor teaching work bench

The microcontroller hardware mock-up is composed for two boards. The first one contains the microcontroller (80C537 belongs to 8051 family), the memories and elementary peripherals (switches and leds) that work as basic digital I/O. The second one is an expansion board devoted to complex external peripherals, such as a LCD display (4 x 16), a hexadecimal keyboard, analog I/O lines and sensors such as temperature and angular position.

The development software is µVision2 [7]. It is a microcontroller development environment of Keil. It is very popular among microcontroller programmers and gives the user the possibility to edit, assemble and debug an application for several families of microcontrollers, specially indicated for 8051 family. This software interchanges debug information with the microcontroller mock-up via RS-232 serial link. Communication management on the side of the microcontroller is performed by means of a monitoring software.

III. Remote laboratory approach

In order to convert the presence learning environment to a successful remote experience it is needed to allow users to interact in two different levels. The first one is the debugging level, which permits the users to download their own code, to monitor and control the registers, the memory contents and the execution. The second level is the interaction between the students and the board at a physical level, such as to press a key on the keyboard, to turn on or off different switches and to read the display output.

To achieve the debugging level, the microcontroller mock-up is connected to a server by means of a RS232 serial link. To download programmes to the microcontroller system and debug microcontroller user applications, the Keil µVision2 is needed. To facilitate the remote use of this software, the Citrix Application Server software is installed. This is a remote access/application publishing product, built on the Independent Computing Architecture (ICA) [8]. The Citrix software provides secure access to applications and programmes installed on the server to a wide number of clients. Therefore the Keil µVision2 is available and ready to be used remotely without the necessity to install any extra software on the client computer. Allowing the remote execution of the µVision2, the possibility to interchange debugging information with the system is assured.

The interaction with the microcontroller system is achieved using a data acquisition board inside another server, to control the analog and digital inputs and getting information from the analog and digital outputs. To visualise the LCD display and the leds, an Internet Protocol (IP) camera is used.

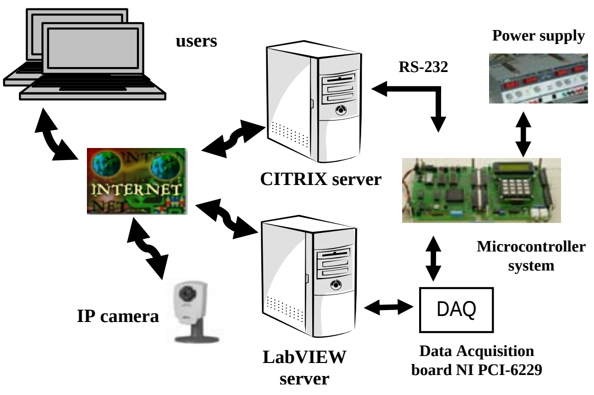

Figure 2. Remote lab diagram.

A remote LabVIEW application working as a Human Board Interface has been developed with the aim to control the data acquisition board and the IP camera, allowing remote users to send and receive information. The block diagram of the full system is shown in figure 2

IV. HUMAN BOARD INTERFACE

Apart of debug the microcontroller applications, it is mandatory to provide to the users the possibility to interact with the microcontroller mock-up in a similar way as if they were present in the laboratory. This means that the remote users have to be able to perform actions such us: change the state of the switches, press a key of the hexadecimal keyboard or read the displays prints.

To replicate the behaviour of the peripherals of the microcontroller mock-up, a human board interface has been developed with LabVIEW 8, which is a graphical development environment for creating tests, measurements and control applications. [9]. With LabVIEW, several Virtual Instruments (VIs) have been created to know and control the state of different peripherals of the microcontroller board. The VIs can be separated in two main parts, the front panel and the block diagram. The front panel is the part that the user interacts with the VI and the one which is published.

When the program is running, the user is able to control, change the inputs, look at the data updated and stop the peripheral VIs, all in real time. The block diagram contains the graphical source code of the VI.

The acquisition and the generation of the physical signals of the microcontroller have been performed by a National Instruments multifunction Data Acquisition (DAQ) board (NI PCI 6229) connected to the server. The Virtual Instruments created to interact with the board are shown in table I.

TABLE I.

VI	Function
Keyboard	Virtual keyboard to replicate the pressed key
Switches	Switch control to change the state of the switches
Leds	To monitor the state of the Leds
Function generator	To generate a signal for feed the analog input.
Oscilloscope	To visualise and measure the signals of the A/D converter.
Potentiometer	Value change of position and temperature sensor.

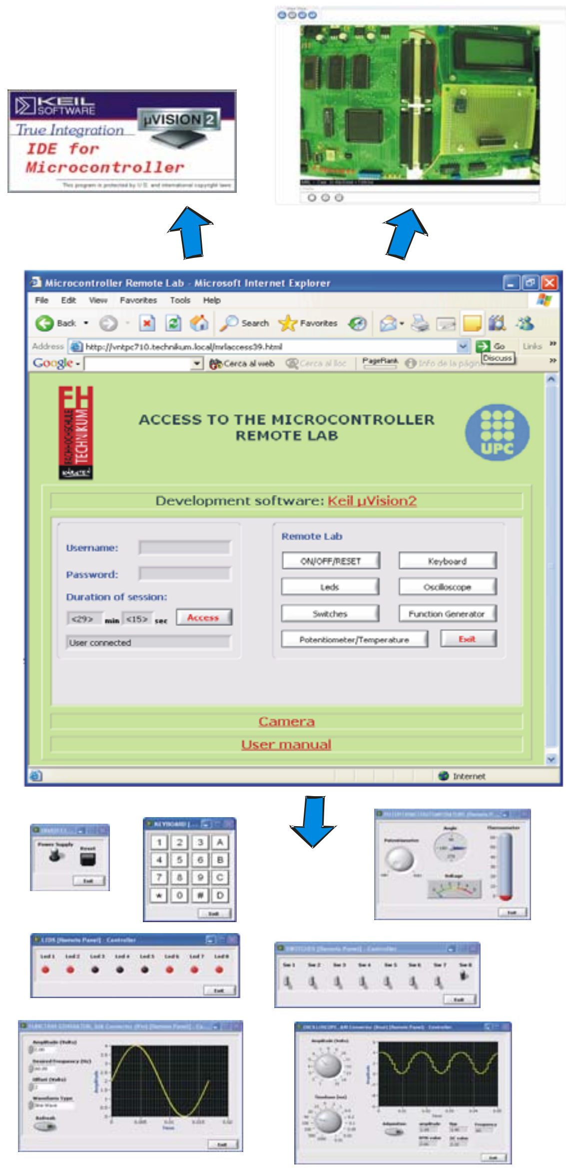

Figure 3. Human Board Interface.

In order to publish the developed LabVIEW application, the LabVIEW web server is utilised. It is enough enabling the LabVIEW web server in the server machine in order that the user can execute the applications remotely. Moreover, the VIs can be included in an HTML file, which can be created by means of the LabVIEW web publishing tool or manually.

Other important aspects to take into account are the scheduling process and the secure access. Regarding the scheduling, at the moment, there is only one physical system available for several potential users. Consequently, it is necessary to have a user access control to give the access to the physical system only to one student simultaneously and to regulate the time per session. On the other hand, this remote lab can not be an open system. However, the user access control assures that the access is only permitted to registered users.

V. CONCLUSIONS

The new self-made remote microprocessor work bench for 8051 family, based on the Citrix server software and remote LabVIEW applications has been presented, including successful practical results.

The remote lab for microprocessors gives the students and lecturers new chances in the learning process. It would be a useful tool in order to overcome the drawbacks related with availability of lab resources.

Students have at their disposal the microprocessor work bench in a remote way. This helps lecturers to implement new teaching methodologies, as PBL. With 24 hours of remote laboratory access, students can organize lab jobs in a satisfactory way, by minimizing displacement and maximizing learning performance.

REFERENCES

[1] Hedley, M.; Barrie, S.: “An Undergraduate Microcontroller Systems Laboratory”, IEEE Trans. on Education, pp. 345-353, 1998.

[2] Rego, H.; Moreira, T; Garcia, F; Morales, E; Barbosa, H; “Educational Technological Specifications to support Distance Education in an E Learning Platform”, ITHET 2005, July 7 – 9, 2005, Juan Dolio (Dominican Republic)

[3] de Assis, A.; Toledo, L.; Moreira, M.; Morelato, A. “Teaching Power Engineering Basics Using Advanced Web Technologies and Problem-Based Learning Environment” IEEE Transactions on Power Systems, Vol. 19, No. 1, February 2004

[4] Maskell, D.L.; Grabau, P.J. “A Multidisciplinary Cooperative Problem-Based Learning Approach to Embedded Systems Design”, IEEE Transactions on Education, pp. 101-103, 1998.

[5] Bruce, J.W.; Harden, J.C.; Reese, R. B. “Cooperative and Progressive Design Experience for Embedded Systems” IEEE Transactions on Education, Vol. 47, No. 1, February 2004

[6] Matthew, R.G; Hughes, D.C. “Getting at deep learning: a problem-based approach”, Engineering Science and Educational Journal, pp. 234-240, 1994.

[7] URL http://www.keil.com

[8] URL http://www.citrix.com

[9] “LabVIEW Basics I Course Manual”, National Instruments Corporation, Texas, 2000.

AUTHORS

Mireia Gilibert. Technical University of Catalonia. Department of Electronic Engineering, Colom 1. 08222 Terrassa. Spain; mgilibert@gmail.com

Jordi Picazo. Technical University of Catalonia. Department of Electronic Engineering, Campus Terrassa. Colom 1. 08222 Terrassa. Spain; jordi.picazo@gmail.com

Michael E. Auer. Carinthia University of Applied Sciences, School of Electronics Villach, Austria. Europastrasse 4. A-9524 Villach/St. Magdalen; M.Auer@IEEE.org

Andreas Pester. Carinthia University of Applied Sciences, School of Electronics Villach, Austria. Europastrasse 4. A-9524 Villach/St. Magdalen; A.Pester@fh-kaernten.at

Jordi Cusidó. Technical University of Catalonia. Department of Electronic Engineering, Campus Terrassa. Colom 1. 08222 Terrassa. Spain; jcusido@eel.upc.edu

Juan A. Ortega. Technical University of Catalonia. Department of Electronic Engineering, Campus Terrassa. Colom 1. 08222 Terrassa. Spain; ortegar@eel.upc.edu