

Online Laboratory Sessions in System Design with DSPs using the R-DSP Lab

<http://dx.doi.org/10.3991/ijoe.v10i4.3270>

A. Kalantzopoulos and E. Zigouris
University of Patras, Patras, Greece

Abstract—This paper presents the online conduction of the System Design with DSPs laboratory sessions through the R-DSP Lab (Remote Digital Signal Processor Laboratory). This interactive RL (Remote Laboratory) supports the verification of DSP applications which are written offline by the students in C and/or assembly programming language, through an internet accessible and user friendly control environment. The latest and most important feature of the R-DSP Lab which is also proposed in this paper, is the remote control of student's DSP applications through GUIs (Graphical User Interfaces) developed by them. In order to demonstrate the above feature, the implementation and verification processes of one laboratory session are presented. The assessment and evaluation results of both the System Design with DSPs laboratory sessions and the R-DSP Lab, are also discussed.

Index Terms—Digital signal processors (DSPs), Distance education, Embedded systems, Remote laboratory (RL).

I. INTRODUCTION

Recently there is a growing research interest in the area of embedded systems. This is originated from the fact that these systems are widely used in consumer-oriented devices. Nowadays, the embedded systems are realized using appropriately configured DSPs (Digital Signal Processors), FPGAs (Field Programmable Gate Arrays), and ASICs (Application Specific Integrated Circuits).

In most cases the embedded systems undertake the implementation of complex real-time digital signal processing algorithms which cover the needs of a wide range of applications from aerospace to multimedia and digital communications. These algorithms are quite demanding and their implementation requires relatively high computational load. As a result, in the most cases embedded systems are equipped with powerful DSPs.

Many universities worldwide have already organized courses on embedded systems which are based on DSPs or FPGAs [1]-[6]. These courses are usually divided in two parts, the lectures and the laboratory sessions. The laboratory sessions are very important for the students' education because they allow students to verify their theoretical knowledge in practice and they improve students' practical skills. This procedure is also very important for their future professional career.

Traditionally, the laboratory sessions take place in hands-on laboratories which allow students to perform experiments in real conditions through the interaction with instruments. However, the cost per workstation is quite high and it depends on the corresponding laboratory

equipment. Consequently, the number of the available workstations is usually limited. In addition, the conduction of any laboratory experiment, in the hands-on laboratories requires the physical presence of one or more instructors. As a result, the access to the students is limited in specific working hours per week.

In order to overcome the disadvantages of the hands-on laboratories, RLs (Remote Laboratories) were introduced by many universities and research groups. Nowadays, the RLs cover many scientific and engineering fields and allow students to access the laboratory equipment from a distance in order to perform their experiments [7]-[17]. These RLs are available twenty four hours per day and they do not require the physical presence of the students and the instructors. The total cost of the RLs is usually less than the corresponding hands-on laboratories because of the better exploitation of the laboratory equipment.

The initial purpose of this paper is to present the online conduction of the System Design with DSPs laboratory sessions utilizing the R-DSP Lab (Remote Digital Signal Processor Laboratory) which is an interactive RL. In addition, a totally new feature of the R-DSP Lab (Remote Digital Signal Processor Laboratory) which increases the sustainability of the presented interactive RL is proposed. According to this feature the R-DSP Lab users (students and instructors) are able to design and perform their own remote experiments by developing their own GUIs (Graphical User Interfaces). These GUIs undertake the remote control of the users' DSP applications which have been uploaded to the R-DSP Lab by the users and they are executed by the DSP development board of an available R-DSP Lab Workstation.

This paper is organized as follows: the organization and the laboratory sessions of the System Design with DSPs postgraduate course are briefly discussed in the second section, while the structure and the features of the R-DSP Lab are presented in the third section. The development stages of a real-time DSP application and its verification process utilizing the R-DSP Lab, from the student's point of view, are demonstrated through an example in the fourth section. The assessment and evaluation results of both the System Design with DSPs laboratory sessions and the R-DSP Lab are discussed in the fifth section.

II. LABORATORY SESSIONS IN SYSTEM DESIGN WITH DSPS

The System Design with DSPs is a postgraduate course which is delivered in the second semester curriculum of two Master degree programs in the Physics Department of the University of Patras, entitled "Electronics and

Communications” and “Electronics and Information Processing”. An average of twenty students attends this course in every academic year and most of them are expected to be familiar at least with the basic theoretical concepts of digital signal processing. They also have the required background in microcomputer architecture and C programming language, and this obtained through the appropriate courses within their undergraduate program curriculum.

The main objective of the System Design with DSPs course is to provide knowledge to the students according to the requirements of the DSP industry. Due to this, the students are trained in the design and implementation of real-time digital signal processing algorithms and applications on DSP based hardware. In order to serve the above targets, the System Design with DSPs course which is a three credit-hour course, is divided into three modules: the lectures, the laboratory sessions and the student projects.

The lectures of this course introduce students to concepts associated with the design and implementation of real-time DSP systems. They also provide students with the necessary theoretical background on a wide range of real-time DSP applications [18], [19]. The organization and the time schedule of the lectures are presented in Fig.1.

In parallel with the lectures, the students carry out a set of six laboratory sessions which are based on the TMS320C6713 DSP of TI (Texas Instruments). The initial purpose of these laboratory sessions is to become the students familiar with the TMS320C6713 DSP and the

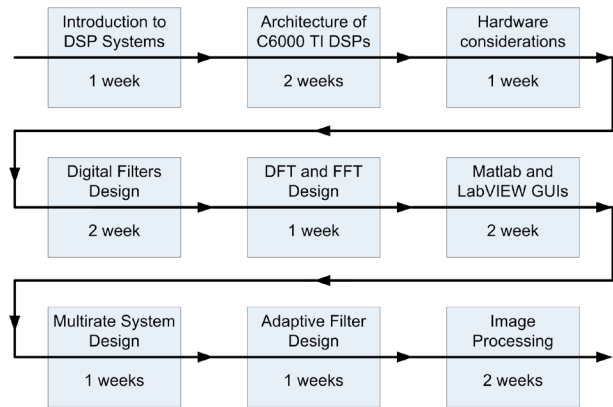


Figure 1. The organization of the System Design with DSPs lectures.

CCS (Code Composer Studio) IDE (Integrated Development Environment) of TI. In addition, the available set of laboratory sessions includes the design and implementation of real-time digital signal processing applications using the DSK (DSP Starter Kit) C6713 development platform of Spectrum Digital. The learning outcomes of the supported laboratory sessions are presented in detail in Table I.

Following the successful completion of these laboratory sessions the students have acquired a significant experience in the development of real-time DSP applications in C and/or assembly programming language. They have also become familiar with the development of GUIs based on Matlab and LabVIEW for the control and verification of their DSP applications. This experience is

TABLE I.
THE LEARNING OUTCOMES OF THE LABORATORY SESSIONS

| Laboratory Sessions | Learning Outcomes | Duration |
|--|--|----------|
| Lab.1: Introduction to the DSPs Architecture (TMS320C6713) and Familiarization with CCS IDE. | <ul style="list-style-type: none"> Familiarization with the features and the hardware components of the DSK C6713. Development of simple DSP applications with CCS using C and/or Assembly. Learn how to use the laboratory equipment. | 2 weeks |
| Lab.2: Design and Implementation of FIR (Finite Impulse Response) Digital Filters. | <ul style="list-style-type: none"> Design of FIR digital filters for given specifications using Matlab. Development of real-time FIR digital filters in C and/or Assembly using the CCS. Verification of the developed real-time FIR digital filters using the laboratory instruments | 2 weeks |
| Lab.3: Design and Implementation of IIR (Infinite Impulse Response) Digital Filters. | <ul style="list-style-type: none"> Design of IIR digital filters for given specifications using Matlab. Development of real-time IIR digital filters in C and/or Assembly using the CCS. Understand the differences between FIR and IIR digital filters. Verification of the developed real-time IIR digital filters using the laboratory equipment. | 2 weeks |
| Lab.4: Implementation of FFT (Fast Fourier Transform). | <ul style="list-style-type: none"> Design and simulation of FFT of 512 points using Matlab. Implementation of a real-time FFT in C using CCS IDE. Comparison of the results between the Matlab simulation and the real-time implementation. | 1 week |
| Lab.5: Design and Implementation of a Ten Band Graphical Equalizer using a GUI Developed with Matlab. | <ul style="list-style-type: none"> Design and simulation of the ten band graphical equalizer's IIR digital filters using Matlab. Implementation of the real-time application with the CCS IDE. Development of a GUI with Matlab for the control of the graphical equalizer. Verification of the graphical equalizer using the laboratory equipment. | 2 weeks |
| Lab.6: Design and Implementation of a DTMF (Dual Tone Multi Frequency) Encoder/ Decoder using a GUI Developed with LabVIEW. | <ul style="list-style-type: none"> Simulation of a DTMF encoder/ decoder using Matlab. Implementation of a real-time DTMF encoder/ decoder using C and the CCS IDE. Development of a GUI with LabVIEW for the control of the DTMF encoder/decoder. Understanding basic aspects of the RTDX (Real-Time Data eXchange) technology. | 2 weeks |

valuable for their professional careers because they will be able to verify advanced DSP applications by developing their own GUIs.

In order to accomplish successfully the requirements of this course, groups of two or three students undertake projects. These projects which are different for each student group, include, but they are not limited to, the implementation of audio effects, adaptive filters, signal modulation/demodulation, image processing etc. This process has a great impact into the students' confidence because they realize how to approach and solve a practical problem.

Traditionally, the laboratory sessions and student projects are performed in the hands-on laboratory which is equipped with ten workstations. This hands-on laboratory is available to students for up to three hours per week and two instructors undertake to support the students. In the past, many students in order to complete in time their laboratory sessions and projects, requested to have access to the hands-on laboratory for more hours per week. In order to response to this major request an interactive RL, called R-DSP Lab, was designed and developed. Through this RL, the students are able to perform their laboratory sessions and projects without place and time barriers. The architecture and the operation modes of the R-DSP Lab are described in the following section.

III. THE R-DSP LAB

The R-DSP Lab is an interactive RL, designed and developed at the Physics Department of the University of Patras, which can be accessed at <http://rdsplab.physics.upatras.gr>. It has a flexible and upgradable structure which is based in two basic structural elements, the Main Web Server and the Workstations [20]. The Main Web Server communicates with one or more Workstations over Internet or Ethernet through a MySQL database. The number of Workstations that will be used depends on the occasional needs and it can be modified without influencing the operation of the R-DSP Lab. The architecture of the R-DSP Lab is presented in Fig. 2.

The Main Web Server [20] undertakes the reception and the authentication of visitors. The home page of the Main Web Server provides information to the visitors about the R-DSP Lab, the equipment and the laboratory sessions of the System Design with DSPs course. After a successful login, the authorized user who wishes to

perform a remote experiment is redirected to experiments web page of Main Web Server in order to select the type and the parameters of the remote experiment which will be performed. Subsequently, the Main Web Server checks the availability of the R-DSP Lab Workstations by retrieving the corresponding data from the MySQL database and reserves the Workstation which serves the fewer users. The user's selections are stored in the database by the Main Web Server and the authorized user is redirected to the reserved Workstation. In addition, the Main Web Server stores statistical information about the usage of the R-DSP Lab in the MySQL database. The above functions of the Main Web Server were developed with PHP. The HTML (HyperText Markup Language) web pages of the Main Web Server are hosted by the Apache web server.

The Workstations are the most critical part of the above architecture because they allow the interaction of the authorized users with the laboratory equipment through an internet connection. At this time, in order to reduce the cost per R-DSP Lab workstation, the equipment of the corresponding hands-on laboratory which includes a Windows based PC (Personal Computer), the TG 2000 function generator of TTI (Thurlby Thandar Instruments) and the TDS1012 oscilloscope of Tektronix, was exploited. Each Workstation is also equipped with a web camera and this increases the sense of reality to the R-DSP Lab users due to the fact that the current status of the available laboratory instruments is displayed.

Every R-DSP Lab Workstation executes three same applications which were developed with LabVIEW and they are called WS_Applications [20]. Each one of these applications was built as an executable file (.exe) and it includes an embedded web server which is provided by LabVIEW. Due to this, the installation of LabVIEW in the Workstation's PC is not required. The main purpose of each WS_Application is to allow only one authorized user to control the above laboratory instruments through a user-friendly control environment which is accessible through a common web browser such as Internet Explorer, Mozilla Firefox and Google Chrome utilizing the remote panels technology of NI (National Instrument). Due to this, the run time engine of LabVIEW which is freely provided by the NI, must be installed in the user's PC.

Consequently, every R-DSP Lab Workstation is able to serve up to three simultaneous authorized users who perform different remote experiments, according to its mode of operation. In the case that the R-DSP Lab Workstation serves more than one authorized users the activated WS_Applications are able to access the available laboratory equipment according to a queue priority logic.

The great advantage of the R-DSP Lab control environment is the accurate representation of the real instruments (oscilloscope and function generator). This environment was designed to support most of the real instruments' features and to allow the users to be experienced with the operation and the handling of the corresponding instruments. Furthermore, there is no risk for the hardware due to the fact that all the necessary hardware connections between the real instruments and the DSP based development platform were implemented by the R-DSP Lab administrators. This fact does not limit the flexibility of the R-DSP Lab because the desired remote experiment is determined by the corresponding executable file which is loaded into the DSP by the user.

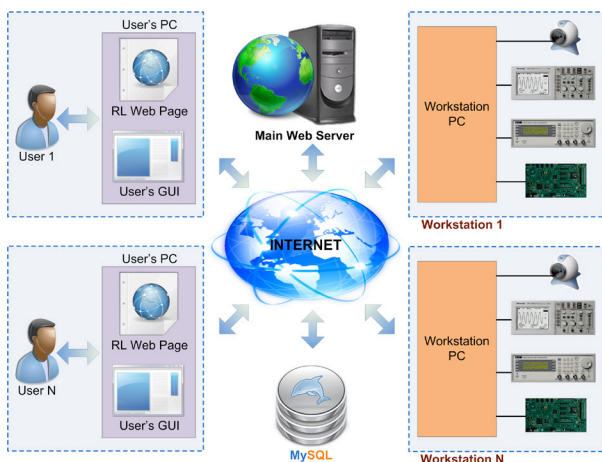


Figure 2. The R-DSP Lab architecture.

From the other hand the supported remote experiments are not time consuming, therefore the users are able to repeat any remote experiment in a few seconds. As a result, the experimental data is not necessary to be saved in the database.

The high security level of the R-DSP Lab is ensured by several unique procedures which are built in both the Main Web Server and the WS_Applications. Initially, the user is identified by the Main Web Server using the user's credentials (username and password) which are stored in the database of the R-DSP Lab. At this time, the above credentials are used only into the R-DSP Lab. When an authorized user requests the execution of a remote experiment, the Main Web Server records the current IP (Internet Protocol) address of the user's PC and redirects the user to a WS_Application of an available Workstation. During the redirection procedure, the corresponding WS_Application retrieves from the database the user's IP address and activates its embedded web server only for the retrieved IP address. This procedure ensures that only the authenticated user has access to the reserved WS_Application. In addition, the WS_Application checks periodically if the authenticated user is still connected. It also monitors if a non-authenticated user attempts to access the WS_Application from the same IP address. In this case it prohibits the access to the non-authorized user. If the authorized user is disconnected, the embedded web server of the corresponding WS_Application is deactivated. Finally, the WS_Application updates the database in order to inform the Main Web Server that it is available to serve a new authorized user.

The communication between the Main Web Server and the Workstations is achieved through the database which is installed into the Main Web Server. Furthermore, the security of the database is very important. Using a firewall the access to the database is permitted only for the IP addresses of the R-DSP Lab Workstations. In addition as a second security level, the access to the database is secured by a password.

The R-DSP Lab, from user's point of view, operates in two different modes, the verification of a DSP executable code and the control of a DSP application through GUIs developed by the students (GUI mode) as it is presented in Fig. 3. In both cases the user should login into the R-DSP Lab through the Main Web Server home page. The mode of operation is selected by the user through the experiment selection web page.

A. Verification of a DSP executable code

In this mode of operation, the authorized student uploads to the database of the Main Web Server the executable code of her/his DSP application. This code, written in C and/or assembly programming language, was produced offline by the user using the CCS IDE. Subsequently, the user is redirected to the web page of an available WS_Application. Through the control environment web page the user is able to control the laboratory equipment in order to verify the operation of her/his DSP application. When the user requests the control of the laboratory instruments the WS_Application checks if the laboratory equipment is reserved by another WS_Application. If the instruments are available, the WS_Application configures the instruments according to the user's selections and the executable code of user's DSP application is downloaded to the DSK C6713 of the

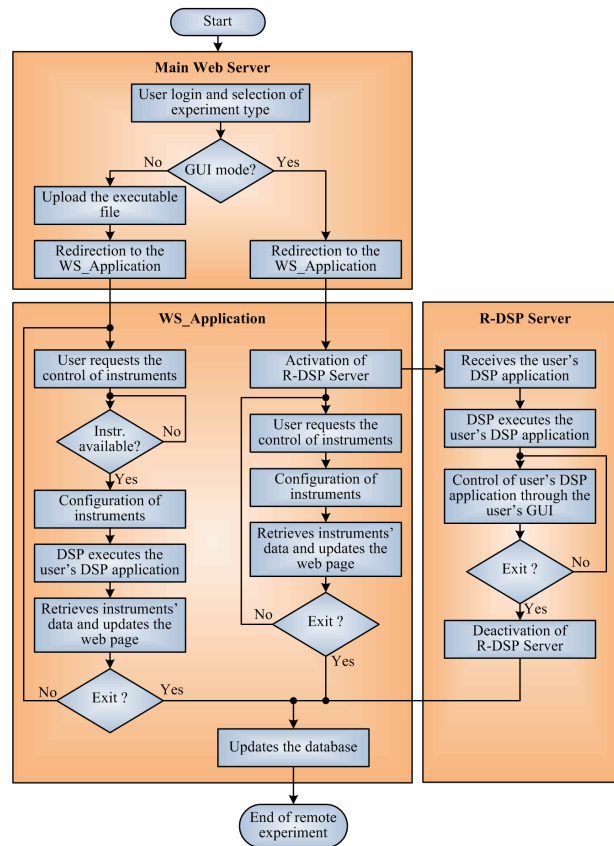


Figure 3. The operation of R-DSP Lab from the user's point of view.

corresponding Workstation. Subsequently, the WS_Application retrieves the instrument data and updates the control environment web page. If the user exits, the WS_Application updates the database in order to inform the Main Web Server that it is available to serve a new authorized user.

This mode of operation was designed to serve the first four laboratory sessions of the System Design with DSPs course according to Table I. The most important advantage of this mode of operation is that each Workstation can simultaneously serve up to three users who are able to perform different experiments using the same laboratory equipment. This feature is based on a queue priority logic which allows only one WS_Application per time to have access to the available laboratory equipment. Therefore, the cost per Workstation is significantly reduced without any influence in its operation.

B. Remote control of a DSP application through GUIs

In the second mode of operation of the R-DSP Lab only one WS_Application of the corresponding Workstation is activated. After the redirection of the authorized user to the available Workstation, the activated WS_Application enables a dedicated server which is called R-DSP Server and it is installed in every Workstation. The R-DSP Server allows the control of the DSK C6713 and the user's DSP application through a GUI developed by the user. This GUI which communicates with the R-DSP Server through TCP/IP (Transmission Control Protocol/Internet Protocol) messages, is running in the user's PC. When the communication between the R-DSP Server and the user's GUI is established the user's GUI sends the executable

code of the user's DSP application which is written in C and/or assembly language. This code is automatically downloaded to the DSK C6713 of the corresponding Workstation and the execution of the DSP application is started. At this point the user through her/his GUI is able to control her/his DSP application. In addition the user utilizing the R-DSP Lab control environment of the corresponding WS_Application is able to control and observe the laboratory instruments (oscilloscope and function generator) in order to take the necessary measurements for the verification of her/his DSP application.

In this scenario the R-DSP Server acts as a slave and waits to receive commands from the user's GUI. When the R-DSP Server receives a command, executes the corresponding procedure. The supported procedures include the loading of the executable code in the DSK C6713 development platform and the control of both the DSP and the CCS. It also supports the communication with the DSP either by direct access of the DSP memory or by utilizing the RTDX (Real-Time Data eXchange) technology of TI DSPs. When each one of the above procedures is completed, the R-DSP Server sends to the GUI a reply TCP/IP message including data and information about the execution of the corresponding procedure.

The above TCP/IP messages are formatted according to the R-DSP Protocol which is based on the MODBUS master/slave protocol. Anyone of these TCP/IP messages is divided into different fields with various lengths which include binary data or information in ASCII (American Standard Code for Information Interchange) format. The encoding and decoding procedures of the above TCP/IP messages can be developed using any programming language. Therefore, the users are able to develop their GUIs with any programming language. More detailed descriptions of both the R-DSP Server and the R-DSP Protocol have been presented in [21].

This mode of operation was designed to support the conduction of the last two laboratory sessions (Table I) and student projects of the presented course. In addition, this totally new feature increases the sustainability of the R-DSP Lab because it allows with a small effort the development of new remote experiments from instructors and students. The disadvantage of the proposed approach is the increased development complexity of such GUIs. This is due to the fact that the users should write from scratch the appropriate encoding and decoding procedures. For educational purposes, the reduction of the above development complexity is quite critical. In order to support this target, two sets of functions for Matlab and LabVIEW users respectively were developed and proposed in [21]. These sets of functions are called R-DSP Matlab and R-DSP LabVIEW Toolkits and undertake the communication between the user's GUI and the R-DSP Server through TCP/IP messages. Using the functions of the above toolkits the development of GUIs either with Matlab or LabVIEW is simplified and accelerated, because it is not required the end users to know in depth the details of both the R-DSP protocol and the R-DSP server.

The operation of the R-DSP Lab in this mode is analytically described in the following section through the verification process of the presented example.

IV. DESIGN, IMPLEMENTATION AND VERIFICATION OF A REAL-TIME DSP APPLICATION

The design and implementation of a real-time DSP application requires several development and verification stages. In order to demonstrate this process, from the student's point of view, the online conduction of the Lab. 6 is described. According to this laboratory session, the students are asked to design and implement a real-time DTMF (Dual Tone Multi Frequency) encoder/decoder which is controlled through a simple LabVIEW based GUI developed by them. This laboratory exercise is related to the basic principles on the encoding and decoding of DTMF signals which are used for signaling in telecommunication systems. It also allows the familiarization of students with the RTDX technology of TI DSPs and the development of GUIs with LabVIEW.

The design process begins by understanding the basic principles of the DTMF encoder/decoder in the classroom using the appropriate textbooks [18], [19]. According to the demands of this laboratory session, the specifications of the DTMF encoder/decoder as they defined by the ITU (International Telecommunication Union) in [22] should be fulfilled. The selected implementation of the DTMF decoder consists of a filter bank based on the modified Goertzel algorithm [18]. This algorithm is one of the most common and efficient methods for the detection of DTMF signals.

The implementation of the real-time DTMF encoder/decoder is divided in two parts, the DSP application and the GUI. The interaction of these parts and their operation are presented in the flow chart of Fig. 4. During the conduction of this laboratory session the students separately design and develop the above parts.

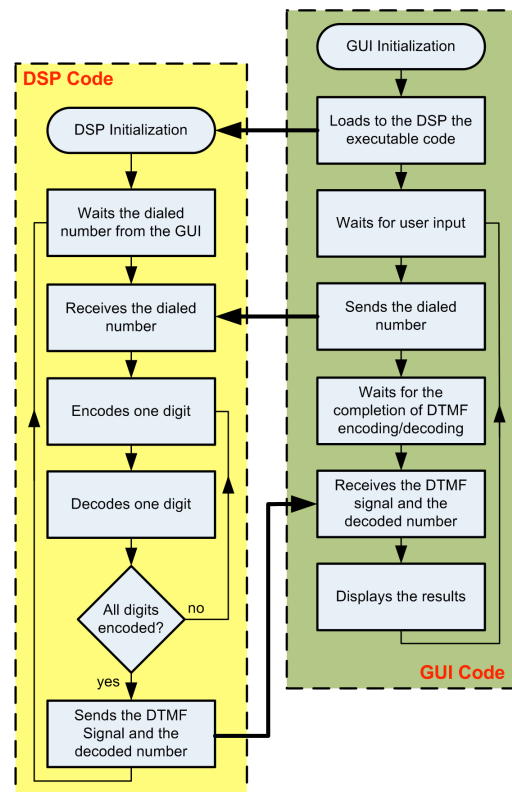


Figure 4. Detailed flow diagram of Lab.6 DTMF encoder/decoder.

The students start with the design and offline simulation of the DTMF encoder/decoder using Matlab. Through this process, they are able to understand the synthesis of two sinusoidal signals for the production of the DTMF signals and the operation of the Goertzel algorithm. The next development stage requires the implementation of the DTMF encoder/decoder in C programming language using the CCS IDE. The C code should also contain the necessary procedures for the communication between the DSP application and the GUI which will be developed by the students in the following development stage.

According to this laboratory session, the C code receives the dialed number from the student's GUI and returns the DTMF signal and the decoded number using the RTDX technology. The C code should also utilize the audio codec (coder/decoder) of the DSK C6713 for the input and output of the signals. In this case, the output signal which is the produced DTMF signal becomes the input signal of the decoding procedure using a loopback technique. The operation of the C code which will be loaded to the DSP is also presented in Fig. 4.

The development of such applications by the students is quite demanding and the students with low programming skills have difficulties to write their own code from scratch. In order to facilitate them, guidelines and supplementary material which includes example codes in C and/or assembly language are offered through the available laboratory handouts.

In the following development stage, the students have to design and implement a quite simple GUI which will control and communicate with the above real-time DTMF application. This GUI (Fig. 5) represents a 4x4 telephone keypad which allows the students to dial the desired number and to observe the results of their DSP code. The communication and the control of the DSP application through the GUI is achieved utilizing the features of both the R-DSP Server and the R-DSP Lab. Due to this, anyone of the students should integrate in her/his GUI the appropriate functions of the R-DSP LabVIEW Toolkit

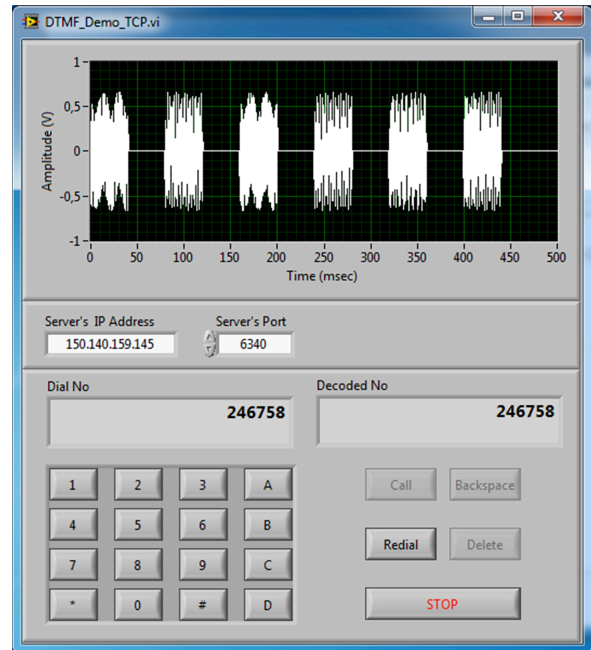


Figure 5. The GUI of Lab.6 DTMF encoder/decoder.

which are explained in the available laboratory materials. The operation of the GUI is described in the flow chart of Fig. 4.

In order to verify the DTMF encoder/decoder using the R-DSP Lab, the student should login and select the appropriate mode of operation. Through this process the reservation of the corresponding Workstation is achieved and the student is able to start the execution of the selected experiment.

Subsequently, she/he activates the implemented GUI which is running locally in the student's personal computer. When the connection between the GUI and R-DSP Server is established, the executable code is downloaded to the DSK C6713 development platform of

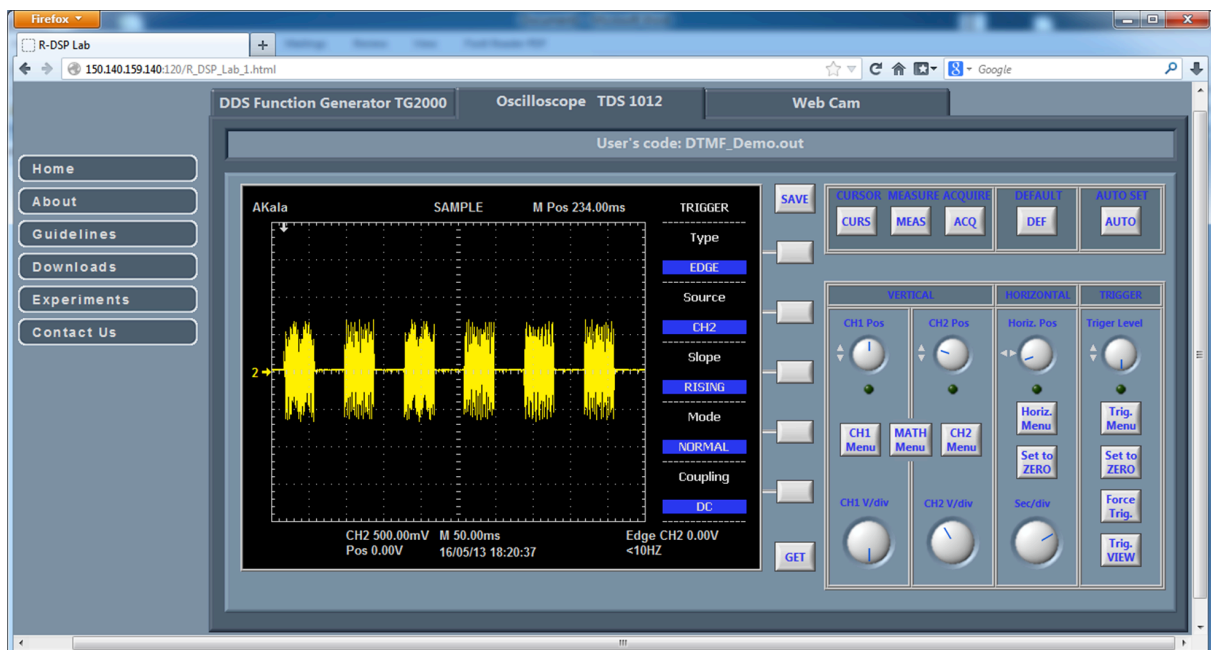


Figure 6. The oscilloscope of the R-DSP Lab control environment.

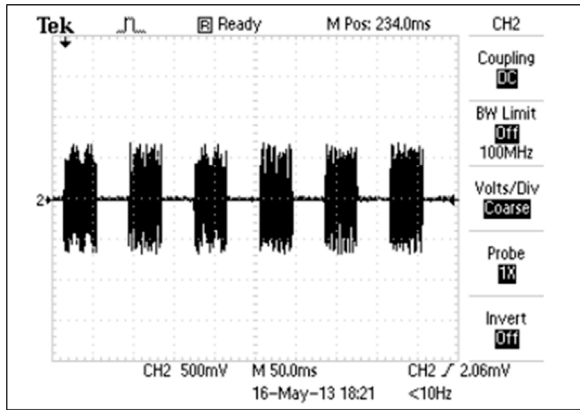


Figure 7. The DTMF signal as it is displayed by the real oscilloscope.

the reserved Workstation and the student remotely controls the DSP application through her/his GUI. In this phase the initialization procedures of both the GUI and the DSP application are completed.

The student using the GUI dials the desired number and observes both the produced DTMF signal and the decoded number. In order to take measurements, the student is able to control the oscilloscope using the R-DSP Lab control environment. This environment is presented in Fig. 6 and is accessible through a common web browser. The real oscilloscope is connected to the line-out of the DSK C6713 and displays the produced DTMF signal (Fig. 7).

V. ASSESSMENT AND EVALUATION

The presented laboratory sessions were integrated into the System Design with DSPs course in 2005 and the R-DSP Lab was available to students in 2009. At the end of each semester, the students fill out an anonymous brief questionnaire about the laboratory sessions and the R-DSP Lab. Subsequently, a discussion with the authors takes place in the classroom and the students are able to express their opinions, comments and suggestions. The evaluation results of the academic years 2009-2010, 2010-2011 and 2011-2012 include the answers of fifty students and they are presented in Table II.

The results of questions 2, 3, 4 and 5 indicate the value of the laboratory sessions from the student’s point of view. Most of the students answered St.A. (Strongly Agree) or M.A (Moderate Agree) in these questions. Due to this, the authors are convinced that the laboratory sessions of the System Design with DSPs course are well organized and enhance the students’ understanding of DSP theoretical concepts. They have also concluded that the students recognize the importance of the laboratory sessions for their education and their future professional careers. This perception is validated by the results of question 5 where the 84% of students answered St. A. or M.A.

In earlier evaluations the students asked for more access to the hands-on laboratory as the result of the required amount of work for the conduction of the laboratory sessions. In response to this major concern the authors designed and developed the R-DSP Lab. According to the presented evaluation, the 78% of students answered St. A. or M.A. in question 6. These results are significantly improved compared to previous evaluations which were approximately 35%. This fact indicates that the R-DSP

Lab allows a better exploitation of students’ time. In addition, an essential improvement of students’ performance has been recorded by the authors the last three academic years. According to the discussion with the students, the authors are deeply convinced that the time management of students has a great impact in their performance.

The evaluation of the R-DSP Lab is achieved through the questions 7, 8 and 9. As it is indicated by the results of question 7, most of the students have utilized the R-DSP Lab for the conduction of their laboratory sessions. The students’ response in question 8 confirms that the R-DSP Lab has a user friendly graphical environment and, consequently, the acceptance of the R-DSP Lab by the students is attributed to its graphical environment.

TABLE II. EVALUATION RESULTS

| | |
|---|---|
| <p>Question 1</p> <p>The course’s organization into lectures, laboratory sessions and projects, is satisfactory.</p> | <p>Question 2</p> <p>The laboratory sessions are well organized and the objectives are well explained.</p> |
| <p>Question 3</p> <p>Handouts and reading assignments cover the requirements of the laboratory sessions.</p> | <p>Question 4</p> <p>The laboratory sessions provided me with a better understanding of DSP concepts learned in the class.</p> |
| <p>Question 5</p> <p>The gained experience is very valuable both for my education and future professional carrier.</p> | <p>Question 6</p> <p>The time schedule for the conduction of the laboratory sessions was satisfying.</p> |
| <p>Question 7</p> <p>The R-DSP Lab contributed to the conduction of the laboratory exercises.</p> | <p>Question 8</p> <p>The R-DSP Lab is user friendly</p> |
| <p>Question 9</p> <p>I did not experience any problems using the R-DSP Lab.</p> | <p>Answers</p> <ul style="list-style-type: none"> ■ St.A. (Strongly Agree) ■ M.A. (Moderately Agree) ■ Sl.A. (Slightly Agree) ■ Sl.D. (Slightly Disagree) ■ M.D. (Moderately Disagree) ■ St.D. (Strongly Disagree) |

Therefore, the graphical environment of a RL is very important because it renders the corresponding RL attractive to students. According to the students' response in question 9, most of them did not mention any problems during the use of the R-DSP Lab. Only a few problems which were mostly associated with the version of the operating system of students' PCs were reported and these were successfully treated by the administrators. In addition, some students mentioned that the time response of the R-DSP Lab is quite large but it is satisfactory for a web application. This could be improved through a replacement of the laboratory instruments but in this case the total cost per Workstation will be significantly increased.

From the above evaluation and assessment it can be concluded that the students are strongly satisfied with both the laboratory sessions and the R-DSP Lab. Besides the results of the above evaluation, the authors noticed a significant improvement of the students' performance compared to previous semesters. In the above period the 85% of students completed successfully the requirements of the System Design with DSPs course. In addition, the average final mark of the students, who passed this course, approached the 8.2 out of 10. According to the records of the R-DSP Lab database, during this period, almost 2400 remote experiments were performed by 44 students. The average time which was spent for the conduction of each remote experiment was about the 15 minutes. This time period is quite short due to the fact that the most time consuming procedures which include the implementation of the DSP applications and GUIs, were performed by the students offline. Consequently, the authors' decision to integrate the R-DSP Lab in the conduction of the laboratory sessions turned to be a great educational benefit for the students.

VI. CONCLUSIONS

The System Design with DSPs postgraduate course is totally focused on the design and development of real-time signal processing applications which are executed on hardware based on DSPs. The laboratory sessions of this course which are presented in this paper, are based on the DSK C6713 development platform. As opposed to the traditional laboratory sessions which are carried out in the hands-on laboratories, the students of this course are able to perform the corresponding experiments remotely using the R-DSP Lab.

This interactive RL allows the students to access the laboratory equipment from a distance twenty four hours per day without the physical presence of an instructor. The students are able to verify their DSP applications which are written by them in C and/or assembly programming language, through the control environment of the R-DSP Lab. This environment supports the remote control of both the function generator and the oscilloscope.

A totally new feature of the R-DSP Lab which is proposed in this paper provides the students with the ability to successfully carry out not only the laboratory sessions Lab.5 and Lab.6 but also their own projects. According to this feature the R-DSP users are able to design and develop their own GUIs in order to remote control their DSP applications which are executed by the DSK C6713 of an available R-DSP Lab Workstation. This feature is demonstrated in this paper by presenting

the implementation and verification processes of the DTMF encoder/decoder (Lab.6) from the student's point of view.

The assessment and evaluation of both the System Design with DSPs laboratory sessions and the R-DSP Lab are also thoroughly discussed in this paper. According to the evaluation results, these laboratory sessions have a great impact in students' education because they provide students with better understanding of the theoretical concepts. In addition, the contribution of the R-DSP Lab in the conduction of the laboratory sessions and projects allows a more efficient exploitation of students' time. As a result, an essential improvement of the students' total performance has been recorded.

Future plans include the integration of new laboratory sessions on digital image and video processing which will be supported by the R-DSP Lab. Due to this, the features of the R-DSP Lab will be expanded in order to support DSP development platforms such as the DSK C6416 and the TMDSVDP6437 which are designed for digital image and video processing applications respectively. In addition, the design and development of new RLs in different areas such as digital electronics and computer architecture seems to be a challenge.

REFERENCES

- [1] I. M. Abdel-Qader, B. J. Bazuin, H. S. Mousavinezhad and J. K. Patrik, "Real-Time Digital Signal Processing in the Undergraduate Curriculum", *IEEE Transactions on Education*, vol. 46, no. 1, pp. 95-101, February 2003, <http://dx.doi.org/10.1109/TE.2002.808273>
- [2] U. Meyer-Bäse, A. Vera, A. Meyer-Bäse, M. S. Pattichis and R. J. Perry, "An Undergraduate Course and Laboratory in Digital Signal Processing with Field Programmable Gate Arrays", *IEEE Transactions on Education*, vol. 53, no. 4, pp. 638-645, November 2010. <http://dx.doi.org/10.1109/TE.2009.2039216>
- [3] W. Gan, "Teaching and Learning the Hows and Whys of Real-Time Digital Signal Processing", *IEEE Transactions on Education*, vol. 45, no. 4, pp. 336-343, November 2002. <http://dx.doi.org/10.1109/TE.2002.804393>
- [4] E. W. Hansen and P. K. Manwaring, "Introducing Signal Processing Through an Advanced Digital Design Course", *14th IEEE Digital Signal Processing Workshop and 6th IEEE Signal Processing Education Workshop (DSP/SPE)*, Sedona, Arizona, USA, pp. 107-111, 4-7 January 2011.
- [5] B. D. Barkana, "Curriculum Development of an Audio Processing Laboratory Course", *Signal & Image Processing: An International Journal (SIPIJ)*, vol. 2, no. 1, pp. 1-13, March 2011.
- [6] A. Spanias, et al., "Interfacing Java-DSP with a TI DSK for Use in a Signal Processing Class", *Computers in Education Journal*, vol. 17, no. 3, pp 27-35, July 2007.
- [7] L. Gomes and S. Bogosyan, "Current Trends in Remote Laboratories", *IEEE Transactions on Industrial Electronics*, vol. 56, no. 12, pp. 4744-4756, December. 2009.
- [8] M. Tawfik, E. Sancristobal, S. Martin, G. Diaz, M. Castro, "State-of-the-Art Remote Laboratories for Industrial Electronics Applications", *10th Technologies Applied to Electronics Teaching (TAE)*, Vigo, Spain, pp. 359-364, 13-15 June, 2012.
- [9] M. Stefanovic, "The Objectives, Architectures and Effects of Distance Learning Laboratories for Industrial Engineering Education", *Computers and Education*, vol. 69, no 1, pp. 250-262, November 2013. <http://dx.doi.org/10.1016/j.compedu.2013.07.011>
- [10] D. Hercog, B.Gergic, S. Uran and K. Jezernik, "A DSP-based Remote Control Laboratory", *IEEE Transactions on Industrial Electronics*, vol. 54, no. 6, pp. 3057-3068, December 2007. <http://dx.doi.org/10.1109/TIE.2007.907009>
- [11] A. Tekin, F. Ata and M. Gökbulut, "Remote Control Laboratory for DSP-Controlled Induction Motor Drives", *Computer Applications in Engineering Education*, vol. 20, no. 4, pp. 702-712, December 2012. <http://dx.doi.org/10.1002/cae.20440>

- [12] I. Gustavsson, et al., "On Objectives of Instructional Laboratories, Individual Assessment and Use of Collaborative Remote Laboratories", *IEEE Transactions on Learning Technologies*, vol. 2, no. 4, pp. 263- 274, October 2009. <http://dx.doi.org/10.1109/TLT.2009.42>
- [13] J. G. Zubia and G. R. Alves, "Using Remote Labs in Education – Two Little Ducks in Remote Experimentation", University of Deusto, Bilbao, 2011.
- [14] B. Popović, N. Popović, D. Mijić, S. Stankovski and G. Ostojić, "Remote Control of Laboratory Equipment for Basic Electronics Courses: A LabVIEW-based Implementation", *Computer Applications in Engineering Education*, vol. 21, no S1, pp. E110-120, August 2013. <http://dx.doi.org/10.1002/cae.20531>
- [15] M. Stefanovic, V. Cvijetkovic, M. Matijevic and V. Simic, "A LabVIEW-based Remote Laboratory Experiments for Control Engineering Education", *Computer Applications in Engineering Education*, vol. 19, no. 3, pp. 538-549, September 2011. <http://dx.doi.org/10.1002/cae.20334>
- [16] S. Gröber, M. Vetter, B. Eckert and H. J. Jodl, "Experimenting from a Distance – Determination of Speed of Light by a Remotely Controlled Laboratory (RCL)", *European Journal of Physics*, vol. 31, no. 3, pp. 563-572, May 2010. <http://dx.doi.org/10.1088/0143-0807/31/3/013>
- [17] F. Schauer, M. Ozvoldova and F. Lustig, "Real Remote Physics Experiments Across Internet – Inherent Part of Integrated E-Learning", *International Journal of Online Engineering (iJOE)*, vol.4, no. 2, pp.54-57, May 2008.
- [18] S. M. Kuo, B. H. Lee and W. Tian, "Real-Time Digital Signal Processing – Implementations and Applications", 2nd ed., John Wiley & Sons, 2006. <http://dx.doi.org/10.1002/0470035528>
- [19] R. Chassaing and D. Reay, "Digital Signal Processing and Applications with the TSM320C6713 and TMS320C6416 DSK", 2nd ed., John Wiley & Sons, 2008. <http://dx.doi.org/10.1002/9780470238141>
- [20] A. Kalantzopoulos, D. Karageorgopoulos and E. Zigouris, "A LabVIEW based Remote DSP Laboratory", *International Journal of Online Engineering (iJOE)*, vol. 4, no. S11: REV 2008, pp. 36-44, July 2008.
- [21] A. Kalantzopoulos, and E. Zigouris, "Real-Time DSP Applications Remotely Controlled through Customized GUIs", 5th European DSP Education and Research Conference (EDERC 2012), Amsterdam, Netherlands, pp. 71-75, 13-14 September 2012. <http://dx.doi.org/10.1109/EDERC.2012.6532228>
- [22] International Telecommunication Union, *Multifrequency Push-Button Signal Reception, General Recommendations on Telephone Switching and Signalling*, ITU-T Recommendation Q.24, November 1988.

AUTHORS

A. Kalantzopoulos is with the Electronics Laboratory, Electronics and Computers Div., Department of Physics, University of Patras, Rio Patras, GR-26500 (e-mail: kalan@upatras.gr).

E. Zigouris is with the Electronics Laboratory, Electronics and Computers Div., Department of Physics, University of Patras, Rio Patras, GR-26500 (e-mail: ez@physics.upatras.gr).

Submitted, October, 19, 2013. Published as resubmitted by the authors on March, 28, 2014.