



APPLYING LTI PROTOCOL TO EXERCISE SOLVING IN A MOODLE PLATFORM

J.Cosp-Vilella^{*1}, R. Meseguer-Pallares², H. Martínez-García¹

¹*Escola Universitàri ad'Enginyeria Tècnica Industrial de Barcelona (EUETIB); Consorci de l'Escola Industrial de Barcelona (CEIB); Departament d'Enginyeria Electrònica; Universitat Politècnica de Catalunya (UPC). BarcelonaTech; c/ Comte d'Urgell, 187. 08036. Barcelona, Spain.*

* Corresponding author: Email: jordi.cosp@upc.edu; Phone: + 34 934137492

²*Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels (EETAC); Departament d'Arquitectura de Computadors; Universitat Politècnica de Catalunya (UPC). BarcelonaTech; c/ Esteve Terradas, 7 - 08860 Castelldefels, Barcelona, Spain.*

Received: 2015-7-2; Accepted: 2016-2-1

Abstract

This paper presents the development of an application to propose complex exercises to students on a Moodle platform by connecting to an external server through LTI (Learning Tools Interoperability) protocol. The system has been applied to automatic grading of hardware descriptions made by students using the high level hardware description language VHDL. In this application the student has to upload his or her description in a Moodle platform and this description is simulated in another server with the appropriate simulation software. Thus, an immediate evaluation of the exercise is obtained with the need of additional software in the platform that hosts the Moodle platform and the student database, being another computer managed by teachers that performs simulations. The system is easily scalable to other fields different from digital electronics as Pspice.

Keywords

IMS LTI, LMS, TSUGI, Moodle, e-learning, elearning, interoperability, VHDL, automatic assessment





1. Introduction

Since the 2009-2010 academic year, new Bachelor's Degrees are being conducted within the framework of the European Higher Education Area (EHEA) at Escola Universitària d'Enginyeria Tècnica Industrial de Barcelona (EUETIB) of Universitat Politècnica de Catalunya – BarcelonaTech (UPC–BarcelonaTech). Specifically, the offered curricula are Biomedical Engineering, Energy Engineering, Electric Engineering, Industrial Electronics and Automation Engineering, Mechanics Engineering and Chemical Engineering (UPC, 2015).

The change, besides eliminating the traditional technical and superior engineering studies and establishing a new curriculum divided in Bachelor's and Master's Degrees, has led to a different way of teaching the studies where the practical part and the conduction of tasks has become more important over previous curricula and has reduced the weight of theoretical classroom tests. This model, beyond the advantages that may have on student education, implies a substantial increase in effort by teaching staff.

For this reason, it is necessary to develop new techniques and tools that allow lecturers to keep track of student work without incurring an appreciable increase in their workload. Learning Content Manager Systems (LCMS) as Moodle are good tools that incorporate multiple functionalities and enable reaching all students massively. Usually, student registration is totally transparent to teachers who have a virtual platform with all data of students already entered at the beginning of each course. The main drawback that may appear is that, as this registration is not managed by the teacher, important limitations may exist. For this reason, it would be desirable to create connectors between the LCMS server that stores student registration data and another server managed by the lecturer that would store tasks to be performed by students, and thus, all the desired software can be installed.

In this work, we present a system installed in a remote server that connects via LTI (Learning Tools Interoperability) protocol developed by IMS Global (Severance et al, 2010) that takes advantage of student database in a Moodle server and then, automatically evaluates the proposed exercises to students using another server managed by the teacher. Specifically, exercises consists of writing high level descriptions of digital systems using



VHDL (Very-high speed integrated circuit Hardware Description Language) in a Digital Electronics and Microprocessors course in the Industrial Electronics and Automation Engineering Bachelor's Degree.

2. Budget constraints concurrent with the implementation of the plan

EHEA implementation at EUETIB began during 2009-2010 academic year and it has replaced plans implemented in 1995 and its reform in 2002.

At first, in spite of the increase of teaching workload in the new system, it had to be implemented with the same teaching resources available in the previous model, which implied a number of severe limitations. However, the actual situation has been different. Due to the economic crisis experienced in Southern Europe, higher education budgets not only have been frozen but have been reduced considerably. Specifically, at Universitat Politècnica de Catalunya (AFS, 2015), and without any correction for inflation or increase of cost due to staff aging, staff budget has been reduced from year 2010 to year 2015 from 206 to 177 million Euros, approximately. This means that staff budget was 16% higher in 2010 when the new model was to begin to be implemented than 2015 when the model was fully implemented. This figures, drawn from UPC webpage and shown in figure 1, include teaching and research staff as well as administrative and technical support staff but ratios between them have not varied significantly in these last five years and it is useful to get an idea of the evolution of available resources.

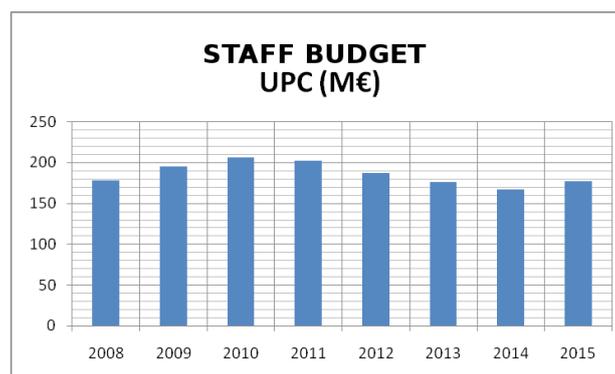


Figure 1. Staff budget evolution at UPC in M€ for the last 8 years



3. The course in which the automatic correction tool has been applied

3.1 Environment of the course within the curriculum

Despite the adverse economic environment, curricula adapted to EHEA have been implemented with successful effort. Among the different courses taught in the new curricula, this work focuses on course "Digital Electronics and Microprocessors (EDM)" (Cosp and Gutiérrez, 2015) which belongs to fifth semester of Bachelor's Degree in Industrial Electronics and Automation Engineering. The course is simultaneous with another course on Electronics Technology (TE) and immediately follows a course on Electronic Systems (STI) that provides an introduction to Electronics.

Before enrolling to the EDM course, students should have acquired the specific competence "Basic knowledge on digital electronics" and during this course should develop other specific competences as "Foundation and application knowledge on digital electronics and microprocessors" and part of "Ability to design analog, digital and power electronic systems". Also, students should develop generic competence "Effective spoken and written communication skills", which is not treated in this work.

Students begin the course once they have followed and passed the STI course, which is transversal to all Engineering degrees taught a EUETIB. In this previous course, basic aspects on electronics are shown and among them, basic digital electronics. In this way, in EDM course, apart from going further on the design of more complex digital systems at gate and block levels, it also exists an important part focused on applications in which students develop the ability to write down digital circuit descriptions using the high-level hardware description language VHDL (IEEE, 2009). VHDL language is a language similar to the programming language ADA but its application is not the description of algorithms to be run on a digital automaton (microprocessor) but the description of digital systems to be modeled, simulated and/or synthesized. By means of these tools, the engineer is able to describe her or his system to, then, have it implemented on a Programmable Logic Device (PLD) or a Specific Application Integrated Circuit (ASIC). Using this technique, circuit design and implementation time is reduced to a large degree compared to traditional implementation techniques where standard TTL circuits are used. In addition to this, this



process implies an important cost reduction and the possibility of implementing moderately complex circuits, which is not feasible in a reasonable amount of time when traditional manual techniques are used. As an example of a VHDL program, figure 2 shows the description of a simple NOT gate.

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;

entity inverter is
  port(
    A: in std_logic;
    X: out std_logic);
end inverter;

architecture LOGIC of inverter is
begin
  X<=not A;
end LOGIC;
```

Figure 2. Elementary VHDL description example. NOT gate.

3.2 Course development and planning

This course is divided in three main parts. The first one is lectures, in which basic concepts are presented. Another one is lab sessions in which students test their designs on a commercial development board (Digilent, 2015) based in an FPGA (Field-Programmable Gate-Array). The last part is homework in which they write down and simulate descriptions that they will test afterward at the lab. This part of the plan is very important as there is not enough time at the lab for the complete description design, simulation and implementation processes, thus, students at the lab must have their design in a well advanced development stage to finish the exercise in time.

Student descriptions are individually reviewed by the instructor as well as at the lab where student receives a feed-back and once the exercise is finished to be revised and evaluated. This process is essential but also enormously tedious. When reviewing the description, the instructor must take into account a great variety of characteristics. Some of them can not be easily automated and must be done by the instructor as the description global assessment, the use of specific required techniques (as structural or behavioral descriptions) and the most important of them, student feedback and personal interaction. However, there



other characteristics that can be easily automatically assessed as correct description syntax and its correct functionality. A tool that could implement these assessments would be a great help in the review process, allowing the instructor to focus on the first aspects that must essentially be carried out by him and can not be left to the machine.

4 Automatic correction tools

The development of tools for automatic correction is an important research field since their great interest in technical education due to exercise complexity. Computer programming and hardware description are not an exemption. Numerous examples of these applications exist. Some examples are given in (Baruque and Herrero, 2015) in which an automatic correction tool is applied to Java programs developed by students and they obtain a preliminary assessment to their tasks previous to their final delivery to their instructor. Also, in (Gutiérrez et al, 2010) it is shown a developed module for Moodle that can automatically verify the uploaded descriptions by students relieving the instructor from continuously supervising student tasks.

When comparing this tools, it is necessary to take into account the following characteristics (Caiza and del Álamo, 2013): The range of programming and/or description hardware languages supported; the programming language used for development that facilitates the deployment and maintenance of the tool, the logical architecture that will set its modularity, scalability and flexibility; the required hardware; if the tool requires other modules or not; technologies used by the tool (standards, protocols, libraries, etc.) and the metrics that it can evaluate as well as the ability to calculate a score for the task.

A good review of existing applications can be found in (Caiza and delÁlamo, 2013) and (Ihantol et al, 2010).

5 Moodle and LTI protocol

A Moodle based LCMS (Atenea (ICE-UPC, 2015a)) is available at UPC and this is the tool used for the automatic correction application in the course presented in this paper. This platform, among other options, gives us the possibility to pose tests with closed answers or solving mathematical exercises using other tools as WIRIS quizzes (ICE-UPC, 2015b) but, obviously, it does not exist any specific embedded module for all disciplines as compilers



and computer simulators. This prevents proposing exercises to students on the development of high-level hardware descriptions as VHDL. Nevertheless, there exists a module that may connect platform Atenea with an external server using LTI protocol (Severance et al, 2010). By using this protocol (figure 3) a task can be generated at Atenea server that is linked to the external server. When the student performs de task, the server sends the necessary information about student to the external server and opens a window on the LCMS with the contents generated by the external application. From this point, the student interacts with these contents but within the environment of the LCMS.

On the other hand, to facilitate the deployment of the protocol and the development of applications using LTI, libraries exist that embed the most common tasks required to establish a successful link between two servers by means of the LTI protocol.

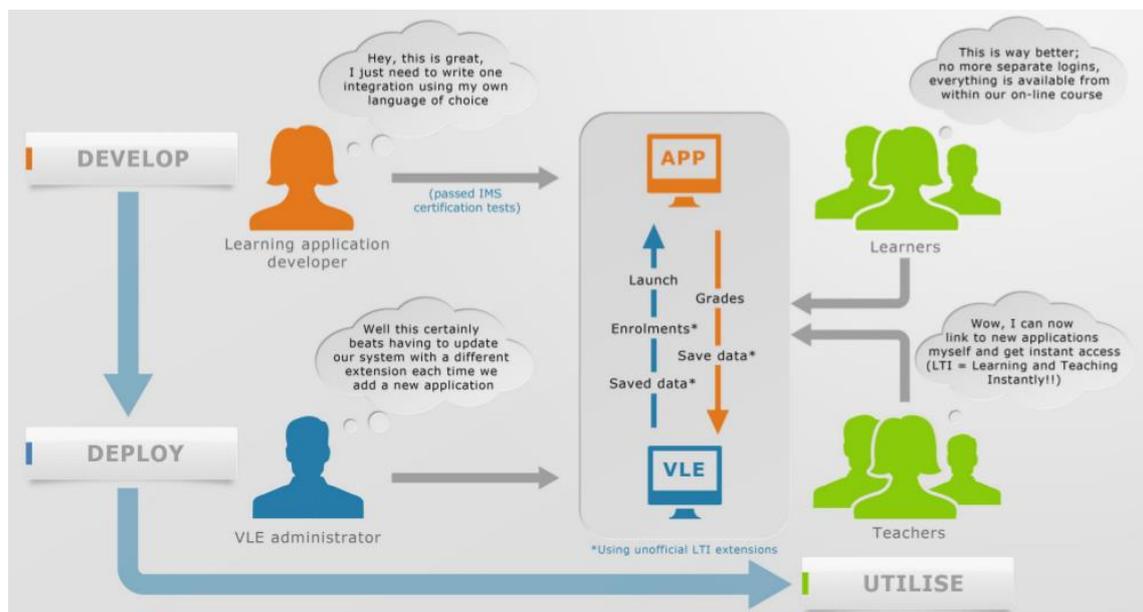


Figure 3.Diagram of a LCMS + LTI system.(Source: www.celtic-project.org)

6 Tool description.

In order to automatically assess exercises consisting in VHDL descriptions, a PHP based application running on a server managed by course instructors has been developed. This application is linked using LTI protocol to a Moodle server (Atenea version from

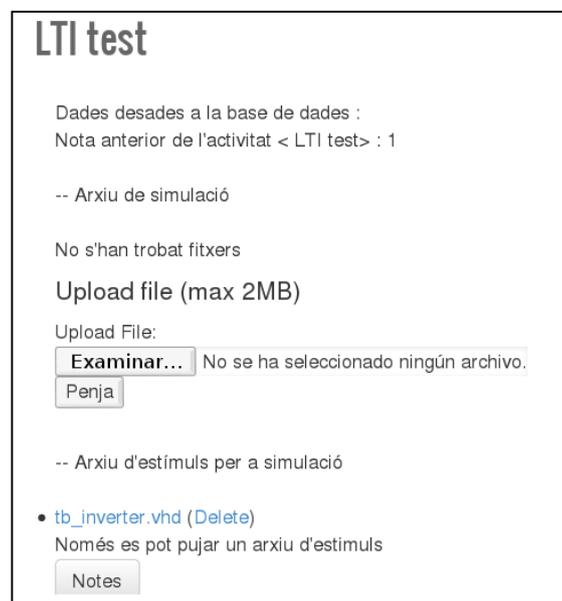


Universitat Politècnica de Catalunya) using the standard TSUGI (Galanis et al., 2014) libraries.

An external task is created in Moodle that is linked to our server where the student can upload the description he has created. Then, once the file has been uploaded, our server proceeds to its simulation by means of the open source VHDL simulator ghdl (Gingold, 2005). Previously, the instructor has uploaded in the same task the VHDL testbench corresponding to the required description to test it. Compilation results (i.e. if there have been any syntax error) and simulation results (i.e. if the student design behavior behaves as expected by the testbench designed by the instructor) is saved in a spreadsheet located at the server managed by instructors.

At this point, as the Moodle implementation at UPC is limited to LTI v.1.1 and v.2.0 is not available yet, it is not possible to return the results and grades to Moodle server, they have to be managed by the external server.

Figure 4 shows a sample window generated by the external server and embedded in Moodle as an external task, specifically a window to upload VHDL files to be simulated (note that this is an instructor's version as it also requires the stimuli file or testbench).



The screenshot shows a web interface titled "LTI test". It contains the following elements:

- Text: "Dades desades a la base de dades : Nota anterior de l'activitat < LTI test > : 1"
- Text: "-- Arxiu de simulació"
- Text: "No s'han trobat fitxers"
- Section: "Upload file (max 2MB)"
- Text: "Upload File:"
- Form: A file selection button labeled "Examinar..." with a message "No se ha seleccionado ningún archivo." and a "Penja" button below it.
- Text: "-- Arxiu d'estímuls per a simulació"
- List: A bullet point for a file named "tb_inverter.vhd (Delete)".
- Text: "Només es pot pujar un arxiu d'estímuls"
- Form: A "Notes" button at the bottom.

Figure 4. Embedded window at LCMS to upload files to be tested. This is the instructor version that includes a stimuli file or testbench.



7 Conclusions

In this paper we have presented the development of an application to automatically assess high-level hardware descriptions, specifically written using VHDL, developed by students following the course "Digital Electronics and Microprocessors" in the Bachelor's Degree in Industrial Electronics and Automation at Universitat Politècnica de Catalunya.

The application is based on PHP language and it is running on a server that can connect to the Learning Content Manager System at UPC (Atenea, which is based in Moodle) by means of LTI protocol. In this application, students, after logging in to the course from the LCMS, are redirected to an external server where their VHDL descriptions are compiled, simulated and its behavior verified by means of a testbench created by the instructor. Once they have finished the proposed exercise, results of each student are exported to a spreadsheet for the instructor to assess the task.

At this moment, the code is fully developed and its functionality verified and it is going to be tested with students during the next semester.

8. References

UPC – Universitat Politècnica de Catalunya (2015). Escola Universitària d'Enginyeria Tècnica Industrial de Barcelona (EUETIB), <http://www.upc.edu/aprender/centros-docentes/euetib>

Severance C., Hanss T., Hardin J. (2010). IMS Learning Tools Interoperability: Enabling a Mash-up Approach to Teaching and Learning Tools. *Technology, Instruction, Cognition & Learning*. 7 (3/4). pp. 245-262

AFS - Accounts and Finance Service at Universitat Politècnica de Catalunya (2015). UPC Budget, <http://www.upc.edu/la-upc/la-institucio/fets-i-xifres/pressupost/pressupost-2015>

Cosp J., Gutiérrez F. (2015). Digital Electronics and Microprocessors Course Guide (2015) <http://www.euetib.upc.edu/els-estudis/estudis-de-grau/grau-en-enginyeria-electronica->





industrial-i-automatica/guiadocent-obtenir-pdf?codi=820224&idioma=en&grup=1

IEEE - Institute of Electric and Electronic Engineers. (2009). IEEE Standard VHDL Language Reference Manual IEEE Std 1076-2008 (Revision of IEEE Std 1076-2002). Institute of Electric and Electronic Engineers.

Digilent Inc. (2015) Basys™2 Spartan-3E FPGA Board. Digilent Inc. <http://www.digilentinc.com/basys2/>

Baruque B., Herrero Á. (2015). Self-Assessment Web Tool for Java Programming. International Joint Conference, Advances in Intelligent Systems and Computing. Á. Herrero et al. (eds.), 369, pp.583-592.

Gutiérrez E., Trenas M. A., Ramos J., Corbera F., Romero S. (2010). A New "Moodle" Module Supporting Automatic Verification of VHDL-Based Assignments. Computers & Education. Ed. Elsevier. 54 (2). pp. 562-577

Caiza J. C., del Álamo Ramiro J. M. (2013). Programming assignments automatic grading: review of tools and implementations. Informe interno E.T.S.I. Telecomunicación (UPM). <http://oa.upm.es/25765/1/INVE\MEM\2013\160449.pdf> (acceso 7 de junio de 2015)

Ihantol P., Ahoniemi T., Karavirta V., Seppälä O. (2010). Review of recent systems for automatic assessment of programming assignments. Proceedings of the 10th Koli Calling International Conference on Computing Education Research - Koli Calling '10. ACM Press. pp. 86-93

ICE-UPC - Institute of Education Sciences (2015a). Atenea Service. Introduction. <http://www.upc.edu/atenea/servicio-atenea>

ICE-UPC - Institute of Education Sciences (2015b) Atenea Service. WIRIS quizzes. <http://www.upc.edu/atenea/servicio-atenea/aplicaciones-y-recursos-en-atenea/wiris-quizzes>



Galanis N., Alier M. Casany M. J., Mayol E., Severance C. (2014). TSUGI. Proceedings of the Second International Conference on Technological Ecosystems for Enhancing Multiculturality - TEEM '14. ACM Press. pp. 409-413

Gingold T. (2005) GHDL Home Page. <http://home.gna.org/ghdl/>

