Peers' evaluation of a reconfigurable IEEE1451.0-compliant and FPGA-based weblab

Ricardo J. Costa[12], Gustavo R. Alves[1], Mário Zenha-Rela[2],
Danilo Garbi Zutin[3], Johan Zackrisson[4], Willian Rochadel[5], Unai Hernández-Jayo[6]

ISEP/CIETI/LABORIS[1], FCTUC/CISUC[2], CUAS[3], BTH[4], UFSC[5], UD[6]
rjc@isep.ipp.pt; gca@isep.ipp.pt; mzrela@dei.uc.pt;
d.garbizutin@fh-kaernten.at, johan.zackrisson@bth.se, willian.rochadel@ufsc.br, unai.hernandez@deusto.es

***Abstract* – It is already more than 10 years that weblabs are seen as important resources to provide the experimental work required in engineering education. Several weblabs have been applied in engineering courses, but there are still unsolved problems related to the development of their infrastructures. For solving some of those problems, it was implemented a weblab with a reconfigurable infrastructure compliant with the IEEE1451.0 Std. and supported by Field Programmable Gate Array (FPGA) technology. This paper presents the referred weblab, and provides and analyses a set of researchers' opinions about the implemented infrastructure, and the adopted methodology for the conduction of real experiments.**

Keywords - Weblabs, Remote labs, Remote experimentation, IEEE1451.0 Std., validation, verification.

I. Introduction

Currently, weblabs are seen as important resources for engineering education [1][2]. However, the lack of standardization for their development and access, encouraged the implementation of an infrastructure compliant with the IEEE1451.0 Std. [3] that describes the so-called smart transducers and the way these can be network-interfaced. To provide the same facilities of traditional laboratories, where users (students and teachers) can setup experiments by selecting different instruments, it was also decided to provide a reconfiguration capability to the weblab infrastructure, taking the advantage of the reconfiguration nature of FPGAs that enable embedding hardware circuits in their cores. The involved research and developments of the implemented weblab was already presented by the authors through several publications [4][5][6]. The focus of this paper is the validation & verification process of the implemented architecture and underlying infrastructure, according to several researchers' opinions involved in the development of weblabs.

Section II of this paper provides an overview of the architecture and underlying infrastructure. Section III presents the objectives and the strategy followed in the validation & verification process. Section IV describes the selected experiments, and section V the implemented methodology. Before conclusions, section VI presents and analyses the opinions reported by the researchers.

II. Implemented Weblab Architecture and the Underlying Infrastructure

The weblab infrastructure is part of an architecture that supports the selection, reconfiguration and standard access to different weblab modules used to control and monitor a specific experiment. These weblab modules (e.g. oscilloscopes, multimeters, switches, or others) are able to be remotely bond to the infrastructure in the same way students and teachers may select a specific instrument in a traditional laboratory to attach to an Experiment Under Test (EUT). Furthermore, those modules are accessed using standard commands provided by the IEEE1451.0 Std. HTTP API, and they can be easily replicated and embedded, through a reconfiguration process, into the weblab infrastructure.

As represented in figure 1, the weblab architecture comprises an underlying infrastructure and a Weblab Server. The infrastructure was designed according to the reference model of the IEEE1451.0 Std.. It integrates a Network Capable Application Processor (NCAP), implemented by a micro web server connected to a Transducer Interface Module (TIM), this implemented by an FPGA-based board. The NCAP provides the IEEE1451.0 HTTP API that enables the remote access to the TIM, which internally embed an IEEE1451.0-compliant module bonding the weblab modules through Transducers Channels (TCs). These modules are described through hardware description files and their features are defined using one or more Transducers Electronic Data Sheets (TEDSs), as defined by the IEEE1451.0 Std..

There are two distinct connections between the NCAP and the TIM. The first is the access connection used to control the operation of the weblab modules. The second is a reconfiguration connection that allows reconfiguring the TIM for bonding those weblab modules. The remote control is made by a direct access to the NCAP using the API, while the reconfiguration requires the use of a Reconfiguration Tool (RecTool) provided by the Weblab Server. This RecTool, whose part of the interface is illustrated in figure 2, is accessible through the web. It allows remote users to upload files defining the weblab modules, and configuration files containing all the instructions required to redefine the IEEE1451.0-compliant module so it can bond those modules.

978-1-4799-2741-8/13/$31.00 ©2013 IEEE

To validate & verify the relevance of the presented weblab for teaching and learning processes in the engineering education, it was decided get opinions of some researchers. Next sections describe the implemented strategy, the involved resources (experiments), the adopted methodology and the obtained results.

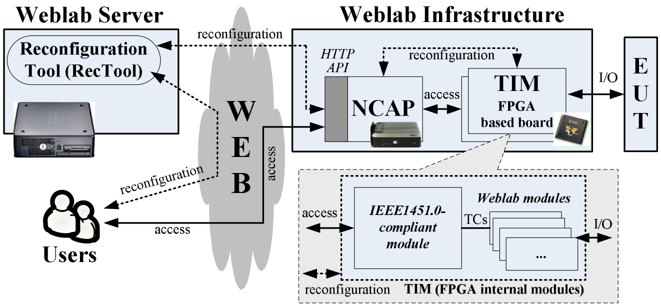

Figure 1. Architecture for the IEEE1451.0-compliant FPGA-based weblab.

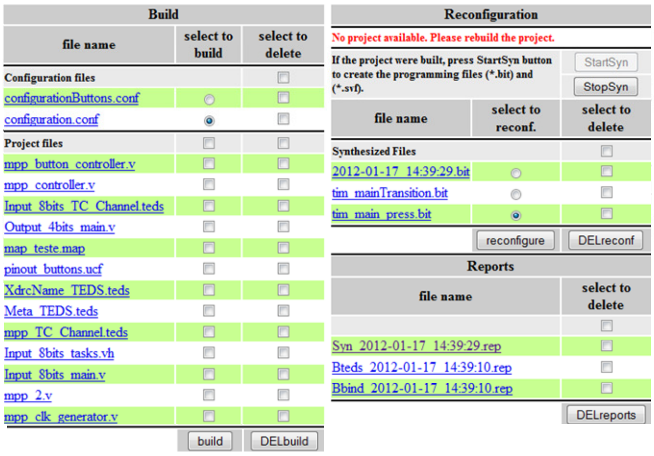

Figure 2. Screenshoot of part of the weblab RecTool interface.

III. THE OBJECTIVES AND THE STRATEGY

The weblab validation & verification process comprehended three main objectives:

1. validate the importance of the weblab for designing and conducting remote experiments;
2. verify if the associated infrastructure runs correctly;
3. get new suggestions to improve the weblab and, new ideas for possible users´ scenarios.

For this purpose, a set of researchers were able to remotely interact with the entire architecture (Weblab Server and infrastructure). They had the possibility to reconfigure the infrastructure with different weblab modules using the RecTool, and to control those same modules using commands provided by the IEEE1451.0 HTTP API. In order to get reliable opinions about the implemented weblab, all researchers were selected with proven expertise in the area of weblabs. A relevant factor for their choice was supported in their past and current research activity concerning the participation in the development and maintenance of weblabs, and in their skills as engineering teachers. Due to the inherent difficulties of obtaining contributions from all the involved researchers in the area, 4 of them were invited to participate in the process. They were invited from different countries with implemented and well disseminated weblabs, and with several publications in the area. The options were the following: i) Unai Hernández from the WebLab-Deusto Research Group, Spain, ii) Danilo Z. Zutin from Carinthia University of Applied Sciences, Austria, iii) Willian Rochadell from the rexlab at the Federal University of Santa Catarina, Brazil, and iv) Johan Zackrisson that is involved in the VISIR project from the Blekinge Institute of Technology, Karlskrona, Sweden. All of them participated in the entire process, excluding Unai, from Spain, that focused his contribution in more generic aspects about the relevance of the implemented weblab in engineering education (he did not interact with the weblab infrastructure). Despite the distinct level of involvement in this process, to guarantee the privacy of their participation, the results are presented as a whole, which means that no particular answers are provided.

During the process, the weblab was introduced to the researchers. They verified if the infrastructure were able to be reconfigured using two distinct layouts, and if the weblab modules adopted in each configuration were correctly accessed using the IEEE1451.0 HTTP API. The entire interaction with the weblab was supported by a webpage that, among an introduction to the architecture, selected experiments and involved technologies, guided researchers to the entire process. In order to get opinions about the weblab, at the end this same webpage, a set of questionnaires were provided to validate its relevance for the experimental work.

IV. SELECTED EXPERIMENTS

Since the two issues under analysis focused on the reconfiguration and control, it was prepared two different configurations for the infrastructure using three weblab modules compatible with the IEEE1451.0-compliant module. In each configuration, researchers were able to run two experiments, namely: the control of a hardware loop and of a bipolar step-motor. In both, a set of weblab modules were reconfigured in the infrastructure using two layouts. The objective was to prove the reconfiguration capability that enables remotely changing and replicating weblab modules without physically modifying the infrastructure. Accessed using single TCs, it was adopted three weblab modules:

1. 8-Bit Input Module - monitors 8 input digital lines;
2. 6-Bit Output Module - controls 6 output digital lines;
3. Step Motor Controller Module (SMCM) - is a more complex module used to control a bipolar step-motor (speed, number of steps, the direction of the rotation, etc.), according to parameters defined in a Manufacturer Defined TEDS (MD-TEDS).

The first configuration, illustrated in figure 3a), adopted the three weblab modules. The hardware loop was controlled using the two I/O digital modules accessed using TC 1 and 2, while the step-motor was controlled using the SMCM accessed by TC 3.

The second configuration, illustrated in figure 3b), adopted the same experiments, i.e. the hardware loop and the step-motor control. In this configuration, the SMCM was not re-configured in the infrastructure, being replaced by a replication of the 6-Bit Output Module. The external wires, that established the hardware loop, were not changed, but the I/O modules were rearranged in the TIM, namely the adopted TCs for their control. In this configuration, the I/O modules adopted for the hardware loop were controlled by TC 2 and 3. The step-motor was controlled using the 6-Bit Output Module accessed by TC 1, which required remote users to send digital sequences to control its rotation.

Due to the specificity of the implemented weblab infrastructure, that involves several technologies and a particular architecture, researchers follow a specific methodology for interacting with the weblab.

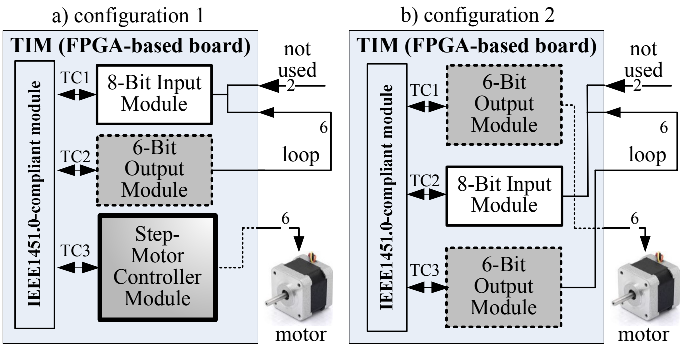

Figure 3. Weblab configurations.

V. IMPLEMENTED METHODOLOGY

The methodology adopted to validate & verify the implemented weblab, was mainly supported by a webpage that presents the weblab architecture and involved technologies, the stages researchers should follow to interact with the weblab, and provides a web link to a set of questionnaires to get researchers' opinions. To facilitate their interaction, it was also provided videos exemplifying the involved stages[1].

A. *Supporting webpage*

The supporting webpage [7] introduced the weblab architecture, the underlying infrastructure, and the involved technology selected for its implementation. Introductory texts were provided about the research work and the contribution it intends to give to the experimental work, namely to the development of weblabs. Web links to previous publications and a detailed description about the implemented infrastructure were also presented and explained. So researchers may understand their expected contribution in this process, the 3 main objectives, already defined in the previous section, and the two main issues to explore were clearly defined. These issues focus on i) reconfiguring the infrastructure with different modules using the RecTool, and on ii) controlling those same modules using a set of commands provided by the IEEE1451.0 HTTP API. Since all invited researchers were more familiar with software architectures, a set of texts about the IEEE1451.0 Std. and FPGAs were suggested to be consulted before proceeding with the verification & validation process. It was explained how researchers can issue commands through a set of buttons, how can the error reply messages be monitored through status registers, and the way to restart the infrastructure in case it enters in a deadlock state. All these cases required issuing IEEE1451.0 commands according to a specific HTTP header.

Before providing links to the questionnaires, to guide researchers during the interaction with the weblab, the same supporting webpage presented the EUTs, the available weblab modules, and the detailed methodology that involves stages to validate and verify the weblab. This way, researchers did not need to understand all the details of the IEEE1451.0 Std., which could make the process more complex.

B. *IEEE1451.0 commands*

Since the objective focused on getting the researchers' opinions involved in the development of weblabs, it was decided to select and pre-define all commands to send to the NCAP. For this purpose, all types of IEEE1451.0 commands were issued using interfaces similar to the one illustrated in figure 4. These interfaces provides 4 windows: i) a window with a set of buttons that when pressed issue a particular command, ii) a command window that presents the command HTTP header, iii) a reply window that presents all commands' replies in a XML format, whose schema is in accordance with the IEEE1451.0 Std., and iv) a button to clear the reply window.

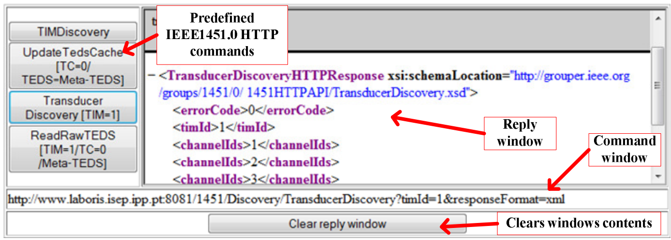

Figure 4. Tipical interface for issuing predefined IEEE1451.0 commands.

C. *Interaction stages*

It was defined three stages to guide users during the interaction with the weblab. Those stages comprehend the sequence that enables: i) configure the weblab infrastructure according to the described layouts in each configuration; ii) verify those configurations and; iii) control the weblab modules.

During the 1st stage researchers used the RecTool interface. They selected a set of predefined files, that defines each weblab module, and the files required to reconfigure the infrastructure. Using the RecTool, those files were uploaded to the Weblab Server defining a new layout for the weblab infrastructure. To verify if the configuration process was running correctly, during this stage users were able to consult a set of reports automatically generated by the RecTool.

[1] The videos can be seen through the following web link: http://www.youtube.com/channel/UCHuj2wC3glXwa0Uls2FXzlA/videos

In the 2nd stage users issued a set of IEEE1451.0 commands to the weblab, in order to verify if the configuration made in the previous stage was really successfully applied to the infrastructure. For this purpose, users issued `ReadRawTEDS` commands to read a set of TEDS of each configuration. The associated replies retrieved all data of the selected TEDS, so researchers may understand that a specific configuration was available in the weblab. All commands' headers were predefined with a set of parameters that specify the target TC, the TEDS, the XML format reply, and others; reducing this way the inherent complexity involved in this process.

The 3rd stage comprehended the access/control of the weblab modules, and consequently of the experiments. This section was divided in two. The first to the hardware loop control and the second to the step-motor control. The hardware loop control used write and read commands, namely the IEEE1451.0 `WR/RDdataSetSegment` commands, since both configurations used the same I/O weblab modules. The difference was the interface with the IEEE1451.0-compliant module, since those modules were bond through different TCs, just to verify the reconfiguration capability of the infrastructure. The step-motor control was made in a distinct way for each configuration. Due to the specificity of this type of experiment, users were able to visualize its axis through an image available in the supporting webpage.

In configuration 1 the step-motor was controlled using the SMCM, whose operation is defined through a MD-TEDS. Users were able to read and write specific fields of this TEDS, before starting the step-motor rotation. As represented in figure 5, this process involved sending a set of commands to verify the possibility of using the IEEE1451.0 Std. to control the step-motor.

During the step-motor control sequence, users were invited to use the `ReadTEDS` command whenever they want to read the current MD-TEDS configuration. Initially the SMCM, defined through its MD-TEDS, had a particular configuration that forces the step-motor to rotate continuously in a specific direction at a speed of 250 ms/step. Users were invited to start the rotation of the step-motor by sending a trigger signal to the SMCM using the `StartTrigger` command and to stop its rotation using the `StopTrigger` command. These commands were able to be issued any time users want to. After testing the continuous rotation of the step-motor, users updated the MD-TEDS's fields, using the `WriteTEDS` and `UpdateTEDS` commands, to a configuration that forces the motor to rotate one turn at a speed of 2.5 ms/step after each received `StartTrigger` command. All commands were applied and the replies monitored using interfaces similar to the one presented in figure 4.

In configuration 2 the step-motor control was made using the 6-Bit Output Module. This required introducing the basis of step-motor control through a small text and through an illustration. By consulting these resources, researchers (at least the ones not familiar with this type of control) were able to understand the reason for sending a set of code sequences to energize the coils of the step-motor, so it may rotate through half-steps. These sequences were sent using the `WRdataSetSegment` command sequentially, so the motor may rotate to the same direction.

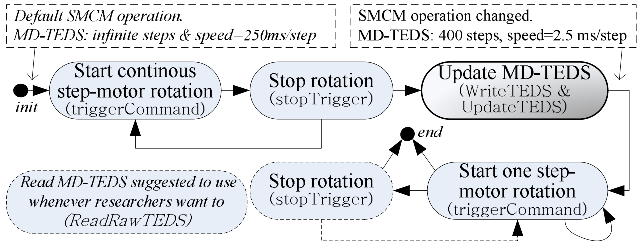

Figure 5. Sequence applied to control the step-motor in the 1st configuration.

D. *Questionnaire*

The questionnaire was provided in the 3rd section of the validation webpage. It was divided in 3 sections each focused on particular issues:

1. section 1- current weblab problems;
2. section 2- operation of the implemented weblab;
3. section 3- relevance of the proposed weblab solution.

VI. ANALYSIS OF REPORTED RESULTS

This section presents questionnaires' results and analyses comments provided by the researchers. Using an empiric method based on a Likert-type scale, all results illustrated in the graphs of this section indicate the average level of agreement, scaled from 1-(low) to 5-(high), researchers have with particular sentences.

A. *Current weblab problems*

The graph of figure 6 indicates the researchers' opinions about six pre-selected problems faced by weblabs. Observing the results, it is evident that there is a general agreement that most of the selected problems must be solved. However, most of the researchers do not consider the lack of a standard architecture important. Rather, supported on further responses, the lack of standard interfaces is a more relevant problem that should be solved briefly. One of the researchers even defend that it is perfectly normal the existence of a diversity of architectures, since different laboratories have different requirements and, therefore, distinct equipments. The common idea to all is that weblabs in the future should follow a plugged & shared approach using friendly interfaces supported by standard APIs, despite they may use different architectures.

Reusability, flexibility and scalability in the integration of weblabs are seen as fundamental issues to improve. This can be done using standard APIs to access, share and maintain weblab sessions independently of the implemented architecture. In this domain, a researcher indicated the current efforts on defining a standard description language so different systems (remote lab management systems, architectural repositories and other systems) may exchange information about their installations. Providing a more reliable hands-on interaction with the experiments, giving to the users the capability of managing connections like in traditional laboratories, was also pointed as an issue to investigate, since it has implications in the pedagogical aspect that still requires a special attention. The difficulty of sharing experiments among institutions was also pointed as an inherent problem caused by the lack of standard access to those weblabs.

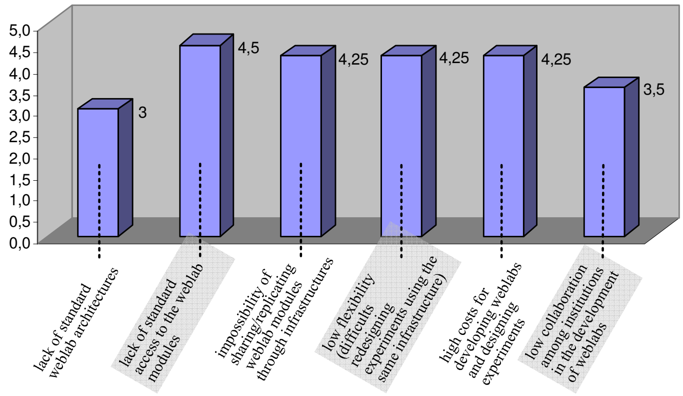

Figure 6. Agreement with six problems currently faced by weblabs.

B. *Operation of the implemented weblab*

With the contribution of 3 researchers, section 2 of the questionnaire focused on verifying the implemented weblab, namely the configuration and the control of the weblab modules adopted for each experiment. This section was divided in three subsections, according to the methodology described in section V of this document, namely: i) configurations, ii) verification of those configurations and iii) weblab modules' control. Graphs of figures 7, 8 and 9, provide the answers to each specific issue, using the same classification made for the graph of figure 6, which means that each answer was classified according to the agreement level of each researcher.

Observing the graph of figure 7, the configuration phase was moderately satisfactory. The RecTool interface was easily understood by all researchers. One of them indicated the simplicity of the design that may promote the users' adoption and experience, since all files and reports, generated during the reconfiguration process, are located in a single and well organized interface. Other indicated that the layout can be improved concerning usability issues, despite the added value it may bring for sharing resources, namely the weblab modules. Nevertheless, same issues were suggested to improve and some facilities do not run as expected, probably due to network firewalls that blocked uploading some configuration files. To overcome this difficulty, the lab administrator uploaded these files using the same RecTool interface by accessing it within the same network where the Weblab Server ran.

Uploading several files one by one to the Weblab Server, was also a suggestion to improve in future versions of the RecTool. Two of the researchers suggested that it will be more user-friendly if they could send several files at once, per example, by concatenating them into an archive. This originated that uploading files to the Weblab Server was considered the more difficult task during the reconfiguration process. Probably due to the inexperience of using the RecTool, some researchers also pointed as a difficult task the process of building the weblab project. They reported some lack of information during this process, and the long time spent (about 20 minutes).

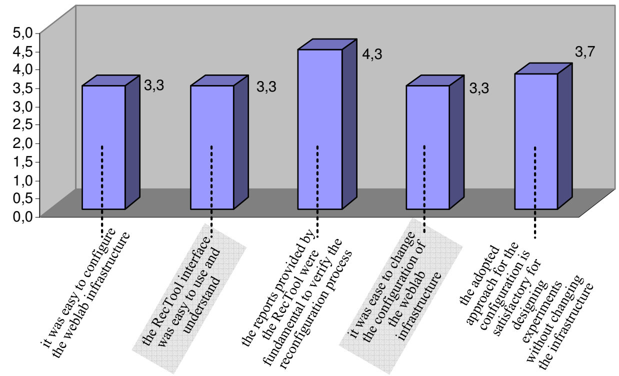

Figure 7. Agreement with the configuration phases.

Next step involved the verification of the configuration. Researchers sent several IEEE1451.0 commands to the weblab and observed the replies in a XML format. The graph of figure 8 presents their opinions during this phase, divided according to generic and particular issues of each configuration. The answers were satisfactory, which means all researchers were able to verify that the weblab infrastructure was reconfigured as expected, according to the replies retrieved from TEDSs read after each configuration. To interpret more easily the information, that was retrieved as a raw of data, it was suggested the development of an interface to provide the information more human readable. It was also suggested improvements to the NCAP-TIM communication, since the infrastructure retrieves some error reply messages after sending some commands, which justifies the average classification of 4 in both configurations.

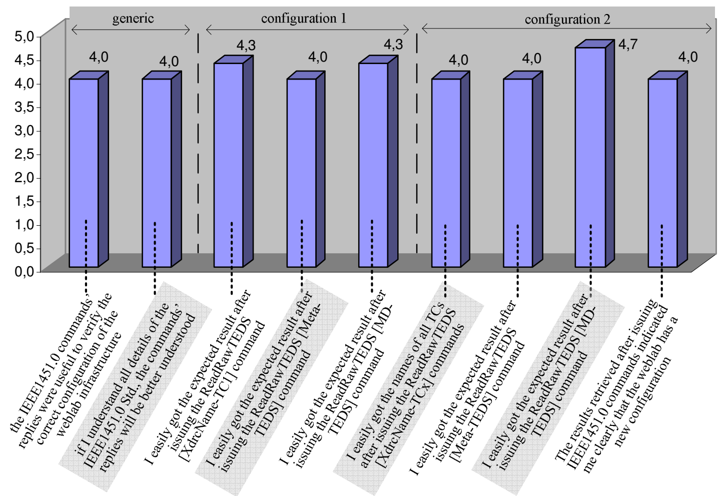

Figure 8. Agreement with the verify configuration phases.

Figure 9 reports the researchers' opinions about the interaction with the weblab modules using IEEE1451.0 commands. Despite the satisfactory answers regarding the control of each experiment in both configurations, some difficulties were reported. The control of the step-motor in configuration 1 was considered easier than in configuration 2 since a single trigger started the rotation of the motor. Some errors were pointed when sending write commands to the weblab, namely to the I/O modules, which justifies the lowest classification in the graph. This was evident regarding the step-motor control rotation in the second configuration, since it required sending several step-codes to the output module to rotate the motor through half-steps. Moreover, it was referred some difficulties observing the rotation of the motor using the axis image available in the supporting webpage, since each half-step corresponds to a very small rotation of 0.9º.

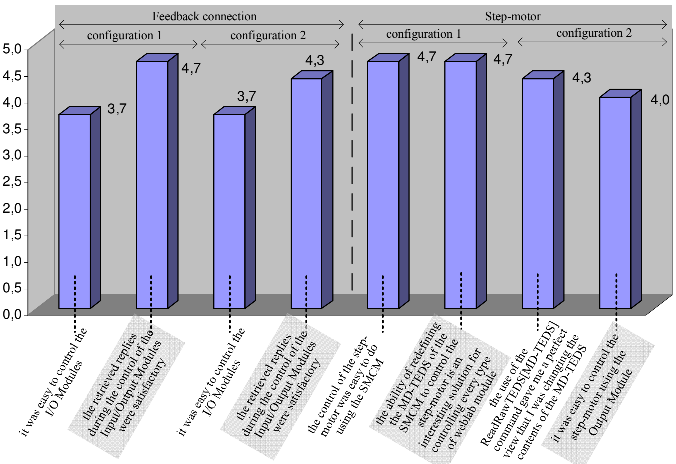

Figure 9. Agreement with the weblab modules' control.

C. Relevance of the proposed solution

Observing the graph of figure 10, it is obvious the possible contribution it may provide for the weblabs standardization. However, it indicates that researchers are not much interested in contributing for the development of new weblab modules and in adopting this type of infrastructures in their classes. This was justified in the remaining answers by the inherent complexity of the architecture and of the IEEE1451.0 Std..

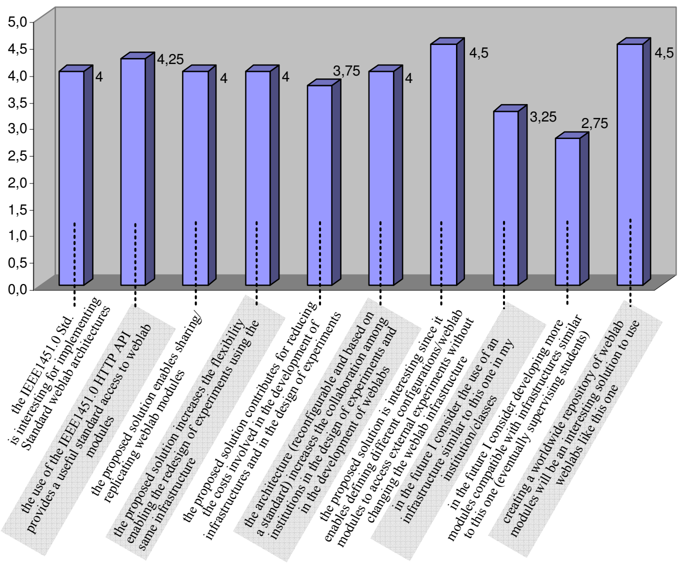

Figure 10. Agreement with the proposed weblab.

These observations are stressed in the remaining comments when they classify the solution as interesting but essentially a proof of concept. The development and replication of weblab modules and the reconfiguration capability were seen as an added value for future weblabs' implementations, since these are standard-based. No other scenarios were suggested, but the selected one was considered appropriated to validate and verify the capability provided by this type of weblabs.

VII. Conclusions

The results presented in this paper showed a generic interest on the implemented solution. Most of the researchers had some difficulties on understanding the IEEE1451.0 Std. details, since they did not have any previous knowledge on this standard, which was perceptible through some answers. This aspect did not influence the objective of this process, since a generic overview about the involved technology was previously presented. All answers showed that the configuration phase was the most problematic. This was perfectly normal, since this phase used a new approach for all of them, and they did not have any expertise in reconfiguring FPGAs. A concluding remark made by one of the researchers indicated that this type of weblabs can provide a significative advantage to developers, since they can use a standard to create their own weblabs. The possibility of replicating weblab installations in an easier way and the use of an IEEE standard was also pointed as advantages. The integration of this weblab architecture in a remote laboratory management system was referred as a potential solution to explore in the future. It was also evident that some improvements should be made to the infrastructure, since sporadic errors appeared when IEEE1451.0 commands were issued. This was a minor aspect that did not influence the main objective of this validation & verification that showed that this type of weblabs can be a valuable contribution to design and access remote experiments.

VIII. References

[1] Doru Popescu and Barry Odbert, 'The Advantages Of Remote Labs In Engineering Education', *Educator's Corner - Agilent Technologies - application note*, p. 11, Apr. 2011.

[2] Javier García Zubía and Gustavo R. Alves, Ed., *Using Remote Labs in Education: Two Little Ducks in Remote Experimentation*. Universidad de Deusto, 2012.

[3] IEEE Std. 1451.0^{TM}, 'IEEE Standard for a Smart Transducer Interface for Sensors and Actuators - Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats', *The Institute of Electrical and Electronics Engineers, Inc.*, p. 335, Sep. 2007.

[4] Ricardo J. Costa, Gustavo R. Alves and Mário Zenha-Rela, 'Reconfigurable weblabs based on the IEEE1451 Std.', *International Journal of Online Engineering (iJOE)*, vol. 6, no. 3, p. 8, Aug. 2010.

[5] Ricardo J. Costa, Gustavo R. Alves and Mário Zenha-Rela, 'Embedding Instruments & Modules into an IEEE1451-FPGA-Based Weblab Infrastructure', *International Journal of Online Engineering (iJOE)*, vol. 8, no. 3, p. 8, Aug. 2012.

[6] Ricardo J. Costa, Gustavo R. Alves and Mário Zenha-Rela, 'Reconfigurable IEEE1451-FPGA based weblab infrastructure', in *9th International Conference on Remote Engineering and Virtual Instrumentation (REV)*, University of Deusto, Bilbao, Spain, July 4 - July 6, 2012, pp. 1 –9.

[7] 'Validation & Verification - supporting webpage -', 2013. [Online]. Available: http://www.laboris.isep.ipp.pt:8082/validation/. [Accessed: 24-Apr-2013].