

Design of a Low Cost Switching Board Enabling a Reconfigurable Remote Experiment

<https://doi.org/10.3991/ijoe.v15i12.10650>

Abderrahmane Adda Benattia, Mohamed Moussa,
Adelhalim Benachenhou (✉), Abdelhamid Mebrouka
Abdelhamid Ibn Badis University, Mostaganem, Algeria
adelhalim.benachenhou@univ-mosta.dz

Abstract—Most of currently remote laboratories implementations include interactive experimentation. In this case, students use real devices and equipment to perform real experiments, which need some flexibility of interaction with the hardware platform. The hardware platform is composed of a Raspberry Pi as a lab server, a switching board (SB), a practical work circuit board and some measurement instruments. The SB is used to make configuration of experimentation by establishing connection between the practical work circuit and measurement instruments. During the experimentation process, students change the setup using a web page. In the background, the hardware configuration is realized using SB, which is controlled by the lab server. The purpose of this work is to develop a new SB in order to provide more possibilities, interaction flexibility with the hardware platform, ease of use, improve performance in response time and finally reduce the cost of the hardware. The SB is based on switches instead of relays. This board can be plugged directly on a Raspberry Pi to facilitate the assembly. It extends the “SPI” bus in order to control some electronic components such as digital potentiometers. Its use is illustrated with a circuit with multiple combinations.

Keywords—Remote Laboratory, analog electronics, Switching board, Reusability, Raspberry Pi

1 Introduction

Laboratories are generally classified into two categories: Virtual Laboratories and Real Laboratories. In virtual laboratories, the student interacts with a mathematical model that simulates a physical phenomenon [1]. They can be accessed locally or remotely. In the latter, the student interacts with real equipment either locally (these are the classical laboratories) or remotely via the Internet. Remote real laboratories require hardware and software development [2][7].

MOSTALAB is a remotely accessible electronic laboratory developed within the framework of a call for projects of the “Agence Universitaire de la Francophonie” (AUF) in order to pool the laboratories within a network [8] and the ERASMUS+

e-LIVES project which aims to promote the concept of e-engineering which is the combination of remote lab and e-learning [9].

Maghreb universities are facing massification of students in the early cycles. Meeting the needs of practical work sessions by the multiplication of measurement benches and premises is very expensive. By bringing the traditional laboratories remotely, we meet the needs of premises; the solution discussed in this paper meets the need for multiplication of measurement benches.

The switching board will allow us to reconfigure the circuit and have several learning scenarios on demand.

Section 2 of this paper describes the design of the architecture of the remote laboratory and the place of the SB, its environment and software configurations. Section 3 presents a case study of remote experiment.

2 Remote Lab Architecture

This section contains a brief overview of our system.

Fig. 1 shows its components: a Lab server hosted on a Single-Board Computer Raspberry Pi [10], a SB, a practical work circuit under test and some measurement instruments.

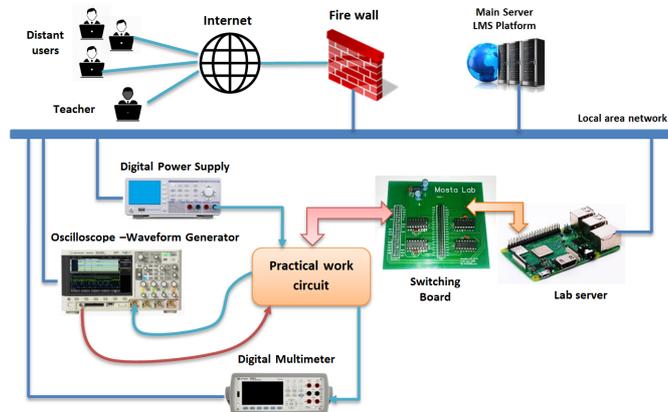


Fig. 1. Remote laboratory architecture

The SB selects the configuration of a desired experiment by establishing a connection between the components of the practical work circuit and the measuring instruments. During a lab session, students select an experiment using a web interface. In the background, the SB, controlled by the lab server configures the desired hardware. Figure 2 illustrates real implementation of this architecture.

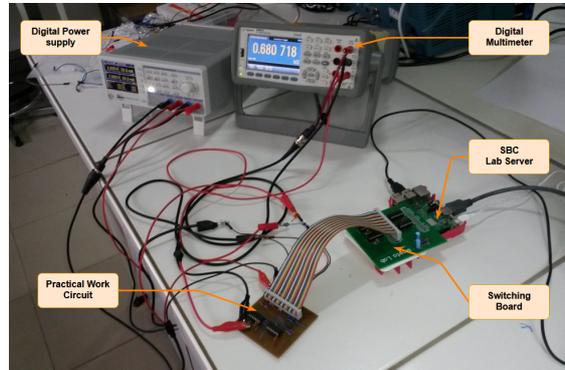


Fig. 2. Real hardware setup

2.1 Place of the switching board in the hardware architecture

A reconfigurable electronic practical work (PW) requires electronic components and switches that select a given configuration. In our system the switches are grouped in one card, the SB, and the electronic components in another. This architecture makes it possible to reuse the switches. Setting up a PW is done by plug and play a new card.

The SB is inserted between the SBC lab server and the components under test. By enabling/disabling the switches, the lab server selects a desired configuration from the predefined ones.

2.2 Evolution in the developed switching board

The first version of the SB shown in Figure 3 and introduced by Farah et al. [11] used relays. She then incorporated a pcDuino server, Fig4. [12]. It has evolved from a hardware and software point of view.

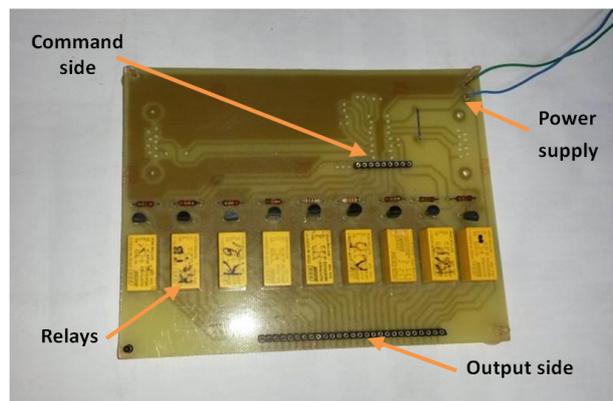


Fig. 3. SB with Relays

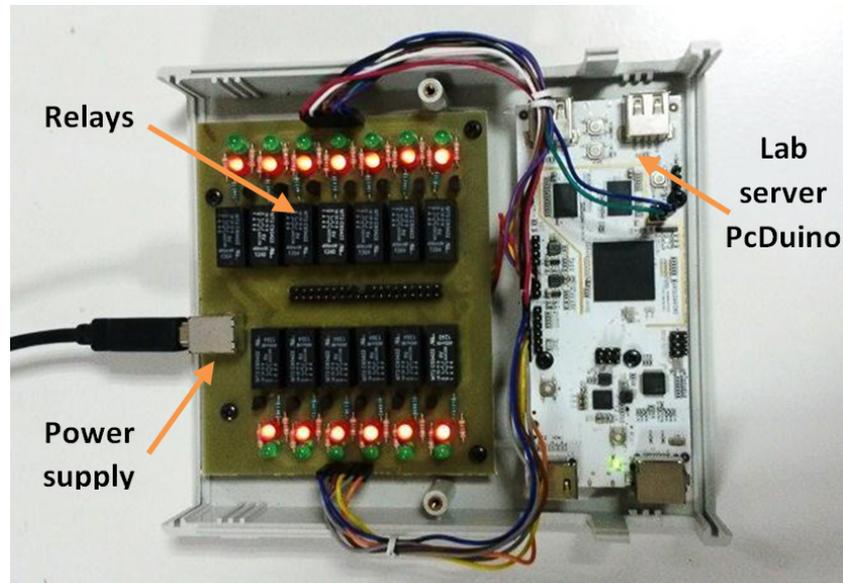


Fig. 4. SB with Relays and PcDuino Board

To facilitate the transition from one practical work to another, PW cards are connected to the SB by a 40-wire flat ribbon. As each single pole double throw relay (SPDT) uses three lines, the SB supported a maximum of thirteen relays.

In the new SB, the relays are replaced by single pole single throw (SPST) type digital switches that only use two lines. The number of switches has increased to sixteen while leaving eight free lines. The new design has reduced the size and allows to directly plug the SB on the Raspberry Pi.

The SB therefore contains four MAX4677 ICs [13]. Each IC contains four independent unipolar CMOS analog switches (SPSTs) that have been designed to switch and route analog voltages. SPST switches consume little power and offer a reduced response time. Another characteristic is the pass resistance (R_{ON}) of 1.6Ω max.

The MAX4677 operates from dual power supply at $\pm 2.7V$ to $\pm 5.5V$ or from a single $+ 2.7V$ to $+ 5.5V$ supply, making it possible for powering directly from the SBC. To power on the MAX4677 IC in dual supply, we used the TC7660H circuit to reverse the $+ 5V$ to $-5V$. Fig. 5.a shows how to get the negative power from the positive one provided by the SBC.

Relays use only thirty-two lines out of the forty available. The remaining eight lines are used to provide power and extend the SPI and one wire buses. It is then possible to control components such as digital potentiometers. Fig.5 b. illustrates the assignment of the different pins of the SB.

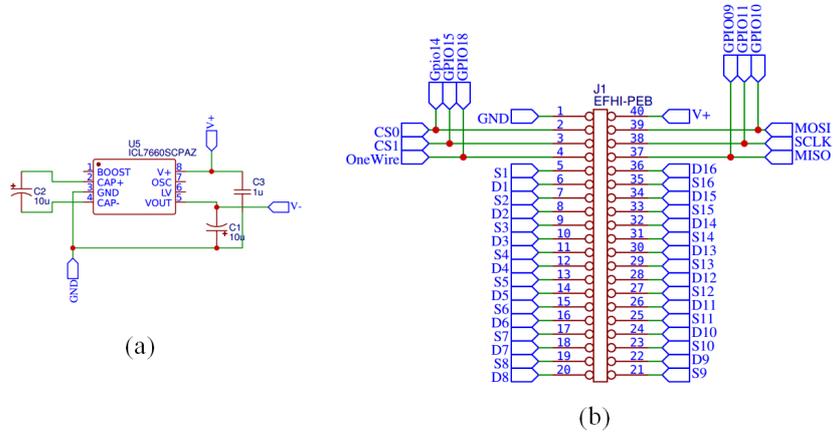


Fig. 5. a) TC7660H used on powerblock b) SB pins map

The final version of the switching board is shown in figure 6.

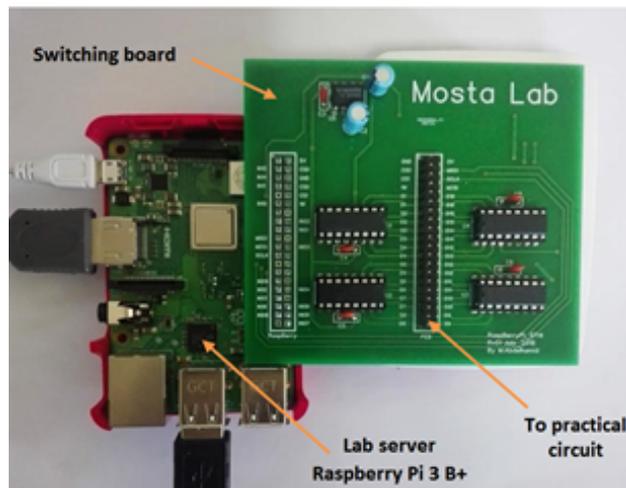


Fig. 6. SB with “SPST” switches plugged on Raspberry Pi

Bill of Materials and Prices: The total cost of this SB version is about \$ 42.

Table 1. BOM of the SB

Components	Amount	Price
MAX4677	4	7.69\$
TC7660H	1	1.14\$
Power supply 100V-230V to 5V/2A	1	3\$
Passive Components PCB – Connectors	-	7.1\$
Total		42 \$

Software evolution: In a remote experimentation, student control the state of the lab with a web page which is the front end, the user interface sends the parameters remotely to the backend as the state of the connection of the SB and values of different components included in the setup. The backend receives parameters in background from a high level interface, when doing manipulation such as selecting a practical work, changing a slider value or click on submit button.

Initially, we have developed the backend which is the server manipulation with Node.js based on JavaScript modules [14], we have used the “Wiring Pi” library to access GPIO pins on the Raspberry Pi Model B [15]. We controlled the values of the GPIO pins to activate relays of the SB in order to make connections for a circuit configuration. The lab server is designed to receive remotely the GPIO values as parameters from the client side via a user interface. We have used Ajax technique for transmitting parameters from user interface to backend [16]. The user interface is developed with HTML/PHP/JavaScript languages. The lab server and the experimentation web page are implemented on the same single board computer, to minimize equipment, but it was a limited solution for future extension.

In the recent version of SB, there is some notable improvement as described below. To optimize the code of the lab server, we have used the “onoff” library [17] under Node.js to access GPIO pins on the Raspberry Pi 3 model B+. The lab server receives remotely the GPIO values as parameters from the client side using the web socket technique, thus we minimize response time of interactions with switches on the board. We use Python language to implement control of some component on the practical circuit connected to the SB using SPI protocol of communication [18].

The user interface is developed with HTML5/JavaScript/CSS languages. It’s composed of input fields; submit buttons, results display as text or images and iframe for displaying results of measurement instruments. The lab server and the experimentation web page are implemented separately. We implemented the experimentation web page under Moodle LMS to allow some evolution of the system, such as duplication of rigs, or to manage the sustained pedagogical control of student.

3 Case Study

3.1 Two-port network characterization

This experimentation aims to study a two-port network circuit. After authentication, user access to the online lab through the learning management system Moodle. The latest is provided as a test including embedded answers using “cloze” questions. The aim of this PW is to characterize a two-port network circuit. Thanks to the SB, each student get a personalized variant of experimentation from the same practical circuit, this makes costs very lower regard the high number of student (fig.7).

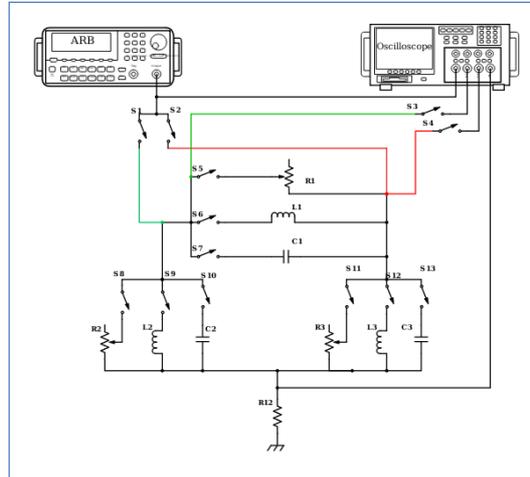


Fig. 7. Multi-variant setup of experimentation for two-port network study

To realize this PW according to a pedagogical scenario, the distant user wires a two-port network circuit to an oscilloscope and a wave generator, then performs measurements and calculates (Z) impedance matrix elements, finally submits results from a web page as shown in figure 8.

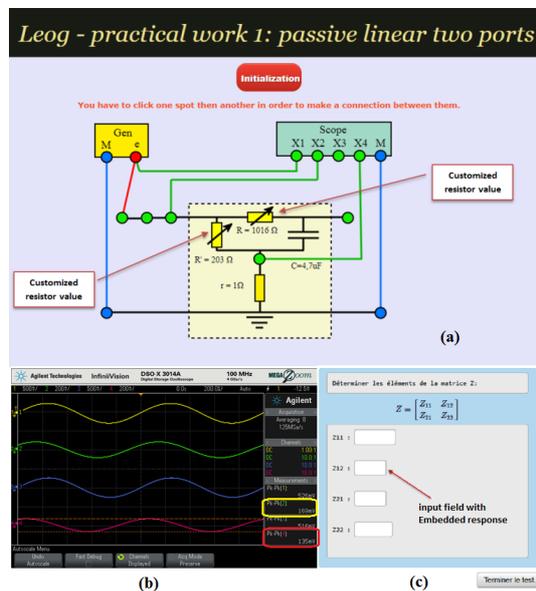


Fig. 8. (a) User interface for two-port network characterization, (b) Oscilloscope interface (c) Input field for user response

Theoretically, equation (1) is the base of all mathematical calculation, it's the Frequency-Domain impedance matrix. Student has to determine “ z_{ij} ” elements by changing wires of two-port network, observe and note measurements displayed on scope interface as shown in figure 8. In this case study capacitor=4.7uF, resistor R=1kΩ, R'=200Ω, and frequency value f=1kHz.

$$\begin{bmatrix} v_1(\omega) \\ v_2(\omega) \end{bmatrix} = \begin{bmatrix} z_{11}(\omega) & z_{12}(\omega) \\ z_{21}(\omega) & z_{22}(\omega) \end{bmatrix} * \begin{bmatrix} i_1(\omega) \\ i_2(\omega) \end{bmatrix} \quad (1)$$

Student calculates z_{ij} using measurements, for example, equation (2) determines the impedance z_{11} which is the ratio of the voltage at port1 to the current in the same port when port 2 is open. Student performs the same calculations for other z_{ij} values.

$$z_{11} = \frac{v_1}{i_1} \Big|_{i_2=0} \quad (2)$$

3.2 Transmission technique

From a technical side, in background, the front-end software sends automatically the configuration code of the practical circuit to the flexible SB via the lab server using “web socket” technique. This configuration is a string of binary code to control GPIO status on the lab server. These GPIO pins activate switches on the SB so that to establish connection between components on the practical circuit and measurement instruments. The designer of experimentation specifies this code of configuration according to the set of components and measurement instruments includes in the practical circuit.

Figure 9 illustrates the transmission technique, the submission of data input for the appropriate practical work circuit as resistor values are transmitted to the server using SPI bus transmission technique to control potentiometers. This allows measurement instrument software to make customized measurements, send back and display the results. Web socket technique is implemented in JavaScript, where transmission by SPI bus is implemented using Python language. We use “iframe” to access and display scope web page using appropriate uniform resource locator (URL) of each measurement instrument.

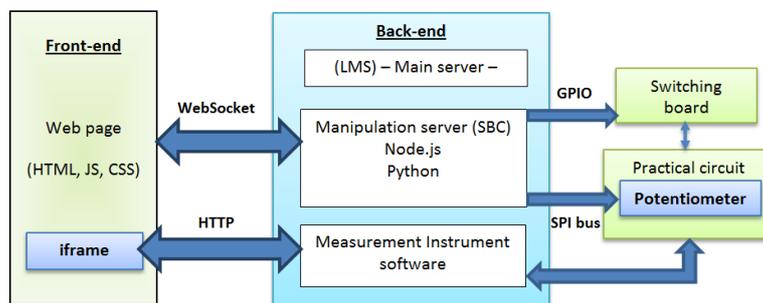


Fig. 9. Interaction technique

4 Conclusion

Considering the new enhancement in our switching board, we have obtained better control of practical work circuits and more flexibility when developing new experimentation setup based on our new SB, that combine performance and ease of use. Some existing experimentations are tested with our SB, such as: study of fundamental theorem, filter circuit (RC, CR, RLC ...), operational amplifier, and diode characterisation.

The use of SB in a remote laboratory is very important to maximize configuration of manipulation and reuse of a same practical work circuit, this makes costs very lower regard the high number of students. In the developed remote laboratory, students manipulate experimentation by interaction with a SB and some measurement instruments; the key of success of the manipulation is the performance in terms of response time, the reusability, the flexibility and ease of use.

5 Acknowledgement

This work is partially supported by the ReLaTraPE project funded by the Agence Universitaire de la Francophonie (AUF) and by the e-LIVES project funded by the European Commission, under agreement number 2017–2891/001 -001. The European Commission support for the production of this publication does not constitute an endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

6 References

- [1] 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, October-December 2009. <https://doi.org/10.1109/tlt.2009.42>.
- [2] F. Y. Limpraptono & I.S. Faradisa, “Development of the Remote Instrumentation Systems Based on Embedded Web to Support Remote Laboratory”, Proceedings of Second International Conference on Electrical Systems, Technology and Information 2015 (ICESTI 2015), Lecture Notes in Electrical Engineering 365. https://doi.org/10.1007/978-981-287-988-2_60.
- [3] Lima, N., Viegas, C., Alves, G., & García-Peñalvo, F. J. (2016). “VISIR’s Usage as an Educational Resource: a Review of the Empirical Research”. Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM’16) (Salamanca, Spain, November 2-4, 2016) (pp. 893-901). New York, NY, USA: ACM. <https://doi.org/10.1145/3012430.3012623>
- [4] R.J.G. Da Costa, “An IEEE1451.0-compliant FPGA-based reconfigurable weblab”, PhD Thesis Information Sciences and Technologies, University of Coimbra, Portugal, January 2014.
- [5] M. Tawfik, E. Sancristobal, S. Martin, “Virtual Instrument Systems in Reality (VISIR) for Remote Wiring and Measurement of Electronic Circuits on Breadboard”, IEEE

- TRANSACTIONS ON LEARNING TECHNOLOGIES, <https://doi.org/10.1109/tlt.2012.20>
- [6] Farag, “An Innovative Remote-Lab Framework for Educational Experimentation”, International Journal of Online Engineering (iJOE), Vol 13, No. 2 (2017), <https://doi.org/10.3991/ijoe.v13i02.6609>
- [7] J. Sáenz, J. Chacón, L. De La Torre, A. Visioli and S. Dormido, "Open and Low-Cost Virtual and Remote Labs on Control Engineering," in *IEEE Access*, vol. 3, pp. 805-814, 2015. <https://doi.org/10.1109/access.2015.2442613>
- [8] <https://www.auf.asie-pacifique/nouvelles/actualites/resultats/>
- [9] <https://e-lives.eu/>
- [10] <https://www.raspberrypi.org/>
- [11] S. Farah, A. Benachenhou, G. Neveux, D. Barataud, “Design of a Flexible Hardware Interface for Multiple Remote Electronic practical Experiments of Virtual Laboratory”, International Journal of Online Engineering (iJOE), Vol 8 (2012). <http://dx.doi.org/10.3991/ijoe.v8iS2.2004> <https://doi.org/10.3991/ijoe.v8iS2.2004>
- [12] S. Farah, A. Benachenhou, G. Neveux, D. Barataud, G. Andrieu and T. Fredon, "Flexible and real-time remote laboratory architecture based on Node.js server," *2015 3rd Experiment International Conference (exp.at'15)*, Ponta Delgada, 2015, pp. 155-156. <https://doi.org/10.1109/expat.2015.7463251>
- [13] <https://datasheets.maximintegrated.com/en/ds/MAX4677-MAX4679.pdf>
- [14] <https://nodejs.org/>
- [15] <http://wiringpi.com/>
- [16] https://www.w3schools.com/jquery/ajax_ajax.asp
- [17] <https://github.com/fivdi/onoff>
- [18] <https://www.python.org/>

7 Authors

Mohamed Moussa is member of the International Association of Online Engineering (IAOE). He is a Phd candidate at LEOG laboratory, Université Abdelhamid Ibn Badis Mostaganem, BP227 27000 Mostaganem, Algeria (email: mhamed.moussa@univ-mosta.dz).

Abderrahmane Adda Benattia is member of the International Association of Online Engineering (IAOE). He is a university teacher in the Department of Electrical Engineering at Ibn-Khaldoun University of Tiaret, Algeria, a research assistant in the LEOG laboratory at University of Mostaganem, Algeria. His current research interest concern Remote Laboratories and Automatic Assessment.

Abdelhalim Benachenhou is professor at Abdelhamid Ibn Badis University of Mostaganem in Algeria and head of LEOG laboratory. He is member of the International Association of Online Engineering (IAOE) and Vice-chair of the e-engineering alliance. His research interests include e-learning and remote laboratories.

Abdelhamid Mebrouka is member of the International Association of Online Engineering (IAOE). He got his Master degree from Université Abdelhamid Ibn Badis Mostaganem, Algeria.

Article submitted 2019-04-11. Resubmitted 2019-06-25. Final acceptance 2019-06-28. Final version published as submitted by the authors.