Experiences on using Arduino for laboratory experiments of Automatic Control and Robotics

F. A. Candelas, G. J. García, S. Puente, J. Pomares, C.A. Jara, J. Pérez, D. Mira, F. Torres

University of Alicante, Carretera de San Vicente del Raspeig, s/n, PO Box 03690, Alicante, Spain

(e-mail: {francisco.candelas, gjgg, santiago.puente, jpomares, carlos.jara, jpalepuz, damian.mira, fernando.torres}@ua.es)

Abstract: The free hardware platforms have become very important in engineering education in recent years. Among these platforms, Arduino highlights, characterized by its versatility, popularity and low price. This paper describes the implementation of four laboratory experiments for Automatic Control and Robotics courses at the University of Alicante, which have been developed based on Arduino and other existing equipment. Results were evaluated taking into account the views of students, concluding that the proposed experiments have been attractive to them, and they have acquired the knowledge about hardware configuration and programming that was intended.

Keywords: Arduino, Automation, Education, Experiments, Laboratory, Robotics.

1. INTRODUCTION

Before the rise of open hardware platforms such as Arduino, hardware and software prototyping was complex and expensive. Therefore, many universities and research centres began to develop cheaper and easier alternatives in the late twentieth century. But these solutions were not general and they were not popular outside the institution where they was used. This was true until Arduino was born in 2005 in the IVREA institute (Italy) as a student project run by Massimo Banzi, who applied the concepts of free hardware and software, which meant a major change (Kushner, D., 2011). The concept of free hardware relates to a design of a microprocessor-based electronic system which is available for free use.

Nowadays, there are available a great variety of Arduino boards with different processors, sizes and connectivity features. The Arduino hardware has become cheap and easy to acquire, with prices ranging from about 20 € to 52 € (+VAT) depending on models. Regarding the software to program Arduino, the same IDE (Integrated Development Environment) is used for all boards, and it is available for different OS (Arduino, 2015). This IDE is open and free, as well as easy to get, start and use. C/C ++ is used as programming language, which enables user to create from a simple program based on procedures in an single file, to a complex object-oriented program in multiple files. Other relevant aspect of the Arduino platform is the big amount of information available about it, ranging from the basic documentation in the official web site, to full books for different application fields (Banzi, 2011; Warren et al. 2011).

Another fact that shows the success of the Arduino platform is the amount of specific courses about it that are offered today, some of them supported by outstanding institutions, such as the course organized by the Spanish Committee in Automatic Control (CEA, 2015). Courses like this, not only are addressed to people with particular interest in Electronics and Robotics, but also to teachers who want to use the platform in education.

All previously mentioned features have made Arduino become a very popular platform and widely extended in the context of education, both in bachelor and higher degrees. Arduino is especially popular in education on Electronics, Automatic Control or Robotics (Barber et al. 2013; Granvillano C. 2014; Ishikawa et al., 2009; Sobota et al. 2013; Úbeda et al., 2009; Valera et al., 2014).

Considering all the features of the Arduino platform, as well as the works and results of other researchers, authors thought this platform was a good choice on which to build laboratory experiments for the subjects they teach. In this way, certain existing deficiencies related to the subjects were resolved, as the next section explains.

The remaining content of this paper is organized as follows. Section 2 explains the reasons that led to use Arduino, the subjects where it has been applied, and the academic context. The main aspects of the developed experiments are described in Section 3. Section 4 presents the results of a survey to the students about their view on the use of Arduino. Finally, the main conclusions of the work are discussed in Section 5.

2. CONTEXT AND MOTIVATION

The work developed includes the design and implementation of four laboratory experiments for some optional subjects taught in official engineering studies of the University of Alicante (Spain), and especially, in the Master's Degree in Automatics and Robotics (MAR). The experiments, which were implemented during the last two academic years (2013-2015), are the following:

- Simple temperature control for a 3D printer hot-end by using a PID. This experiment is developed in two subjects which introduce students to the Automatic Control: Automation and Robotics (6 ECTS) of the Degree in Computer Engineering, and Industrial Automation (6 ECTS) of the Degree in Chemical Engineering.
- Automation of a Cartesian robot. The main objective of this experiment is to teach students how to use and program a microcontroller as a controller for an industrial machine in the subject Industrial Computing (3 ECTS) of the MAR.
- Humanoid robot programming. This experiment is taught in the subject New Trends in Robotics (3 ECTS) of the MAR, and its main aim is to show students the working and control of a type of robot very different from other ones seen in other subjects of the master.
- Follower robot programming. Different activities are developed with a follower robot in order to introduce Robotics, not only in subjects of engineering degrees as the previously mentioned Automation and Robotics, but also in activities for free-elective credits.

With the developed experiments, authors have tried to achieve three basic objectives, as follows:

- Experiments should be attractive to students, and easy to start with them, so that students focus on the important issues, and they do not lose much time learning how a specific device is configured. The Arduino platform meets this purpose. Besides, the knowledge learned about Arduino in one subject is useful to other later subjects.
- Some existing equipment with little or no previous use, but which could be interesting for teaching the subjects, should be reused, replacing the original controller with an Arduino.
- It was no possible to acquire more than one unit of the equipment for some experiments, mainly due to the costs of some of the robots. However, each student could have an additional Arduino board on which to carry out the development and a basic testing. This enables students to develop a first program that can be executed in the final equipment, and makes easier to share that equipment.

In addition to the previously mentioned objectives, there is another important aspect that has been taken into account to choose the Arduino platform. The entering students to the MAR come from different degrees in engineering (Computer Science, Electronics, Industrial Engineering, Production Design, Telecommunications, etc.), and, while all them have a good basis in engineering, it has been found that the levels of knowledge about programming and hardware aspects depend heavily on the source degree. This fact also depends on the double orientation of the MAR, professional and research. There are students who are professionals enrolled mainly to catch up in hardware or programming technologies. In this way, it can be considered two kinds of students in general. On the one hand, students with good background in hardware or Automatic Control, but not in programming. On the other hand, students who are experts in Computer Science and programming but have a limited knowledge of Automatic Control.

Bearing in mind the characteristics of Arduino, authors thought that this could be a good platform to develop laboratory experiments which are attractive for the two kinds of students. On the one hand, students with a more industrial or electronics profile could learn object-oriented programming with C++, benefiting from their knowledge on hardware. On the other hand, students from Computer Engineering could learn to connect devices such as sensors and actuators, while taking advantage of his training in programming. This was other important reason to do the work described here.

3. DEVELOPED EXPERIMENTS

3.1 Temperature control for 3D printer hot-end

The objectives of this laboratory experiment are to introduce the Arduino environment to students, teach students how to develop a simple circuit for temperature control, and program and adjust a digital PID controller.

On the one hand, students have to implement and verify the necessary circuitry for controlling the temperature in the extrusion head, commonly called hot-end, of a standard 3D printer. As Figure 1 shows, the hot-end has a resistor for heating the filament flowing through it, and a thermistor to measure the temperature. Thus, two basic circuits are necessary to connect the hot-end to the Arduino ONE used as controller (see Figure 2). The first circuit for the temperature reading through an analog input, and the second one for actuating the resistor via a digital output.

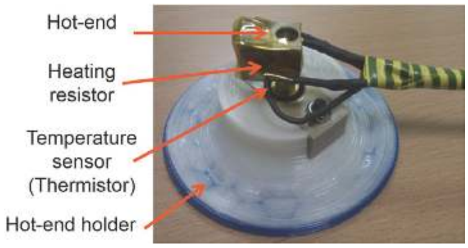

Fig. 1. 3D printer hot-end mounted on a holder.

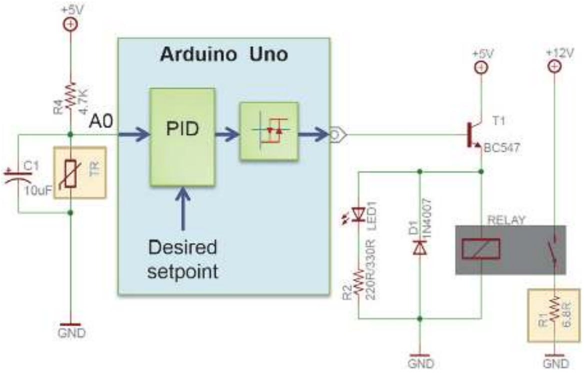

Fig. 2. Circuits connecting Arduino with the hot-end.

On the other hand, the student has to develop a program for Arduino, which reads and scales the sensor temperature, compares it with the reference, applies a PID control algorithm, and activates the heater resistor by using a hysteresis loop. Student has to tune the PID to meet an overshoot and a settling time, as well as adjust the parameters of the hysteresis.

Students had 6 hours to carry out individually the different phases of the experiment. An Arduino and a set of the necessary components to assembly the circuits, including the hot-end, were provided to each student. For the assessment, each student has to deliver a report summarizing his results.

3.2 Automation of a Cartesian robot

This experiment is based on a Cartesian robot which moves a carriage along the axes horizontal and vertical, covering a planar working area close to 4m^2, as Figure 3 shows. The main goals of this experiment are to show students the basic elements of an industrial automation system, and teach them to develop a control program that interacts with the sensors and the actuators to move the carriage to the desired target positions accurately. Students also have to bear into mind the alarm conditions such as the emergency button or the alarm signals from the motor drivers.

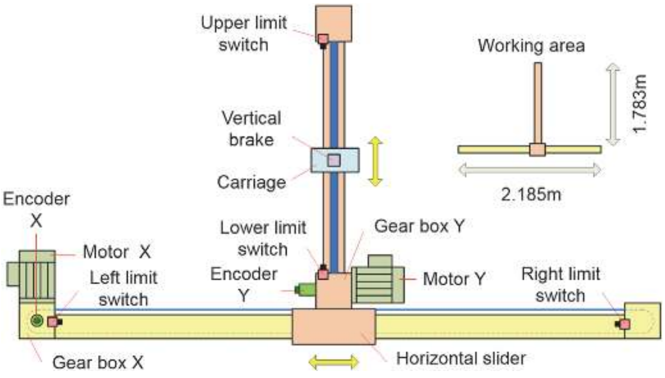

Fig. 3. Scheme of the Cartesian robot.

The robot includes basic elements used in many industrial automation equipment, as Figure 4 shows. As actuators, the robot has two triphasic power motors, managed by two inverters, and a pneumatic brake on the vertical axis. Each inverter has several input logic signals to control the direction and the speed, as well as output signals indicating the state of the motor (running, overcurrent alarm). The sensors are four limit switches and two incremental encoders. In addition, the robot has a control panel with a joystick for manual movements, an emergency button, and several lights.

The Cartesian robot is usually controlled by a PLC S220 from Siemens, which uses 24V logic. But in this experiment, the PLC is replaced with an Arduino Mega 2560. This Arduino is characterized by having many I/O lines, which is essential to interact with all the elements of the robot. In order to have the option of switching between the original PLC and the Arduino, and to convert the voltage levels between the 24V I/O signals and the 5V logic of Arduino, the authors have developed the adaptation circuits, which also include electrical insulation to avoid surge problems in the Arduino and the PC. The circuit for the input signals is also responsible for directing the signals from the encoders and other relevant sensors, such as the limit switches and the emergency button, to the interrupt inputs of Arduino, being necessary to share interruptions for some sensors.

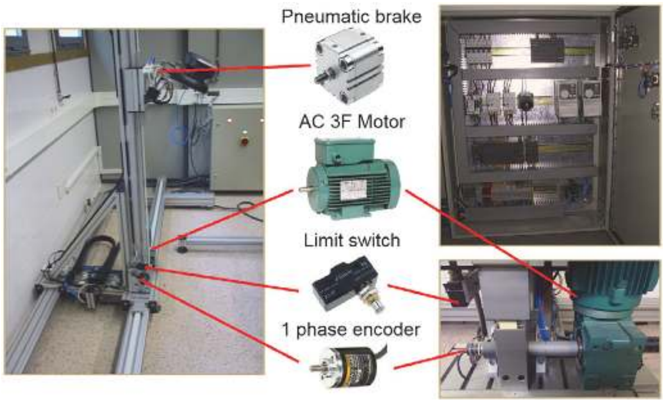

Fig. 4. Cartesian robot and its main components.

To ensure that all students know the robot operation and they are able to make a basic control program, without preventing the most outstanding students to consider more advanced features, two blocks of tasks are proposed. The first block includes these mandatory tasks: calibrate the axles, control the vertical axis brake, detect the limit switches, move the carriage in both axes with the joystick, manage the emergency button, and detect possible alarms from inverters. As optional work, the following task are proposed: use of speed ramps for movements, develop a graphical interface for a PC and communicate this with the Arduino, as well as move the carriage to a target position, either according to a number of encoder steps, or according to a distance in meters.

For the assessment, students had to show the teacher how the tasks proposed were carried out by real robot using their program, while they explained how the program solved the tasks. The experiment was developed by pairs of students, not only because the number of students (12) or the fact that only there was one robot, but also because this experiment was considered to be a bit complex. Students had 12 hours of laboratory for this experiment.

Although only one robot was available, the teacher supplied each group of students with an Arduino Mega, a prototyping board, and a set of components such as LEDs, push-buttons and encoders. These elements enabled students to simulate many of the features of the robot while they were developing the program.

3.3 Humanoid robot programming

The Robonova is a humanoid robot (biped and with two arms) of about 31cm high, developed by HiTec. Although this robot is no longer sold, today there are many similar alternatives. Thanks to its 16 degrees of freedom, this robot can perform many movements including walking, bending or lifting, up or down stairs, etc. The robot was supplied with the controller MR-C3024 which was quite limited in choices of programming and connection of sensors. Therefore, owners used to change the controller (Perea et al. 2010).

In this experiment, students have to become familiar with the working of the humanoid robot, and program it to walk. For the experiment, the Robonova has been modified to use an Arduino DUE as a controller, as Figure 5 shows.

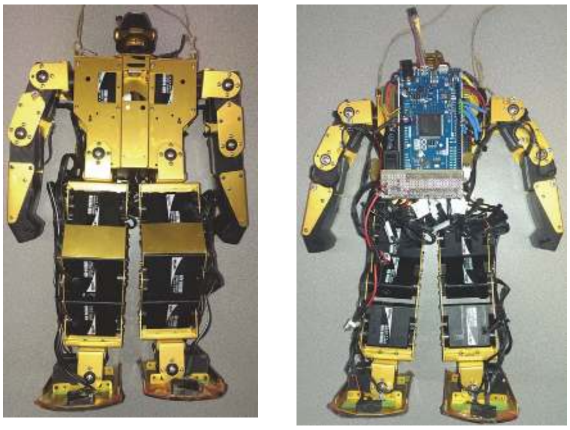

Fig. 5. Robonova modified with Arduino Due.

The Arduino DUE is characterized by using a 32-bit ARM processor, which is far more powerful than those of traditional Arduinos, besides having more memory for program and data. This way, the Arduino DUE is able to execute complex programs while accurately generates the PWM control signals for the 16 digital RC servos of the robot, which are directly connected to it. The Arduino also receives the signals from four bumpers located at the feet of the robot, and three IR distance sensors placed on the shoulders. In addition, three LEDs that can be controlled from the Arduino have been included in the head of the robot.

In the program, all servos are managed as a vector of 16 objects created from the standard Arduino library "Servo". These objects are initialized for the range [600μs, 2.400μs] allowed by the Robonova servos, which matches to an angle of 0 to 180 degrees, and differs from the standard range [544μs, 2.400μs]. The library generates the PWM signals to the servos by software, with aid of the Arduino timers. In order to generate the 16 PWM signals with greater precision and stability, it is possible take advantage of the fact that the Arduino DUE has more timers than other versions, by setting the parameter SERVOS_PER_TIMER in the source file "servo.h" to 4.

As a mandatory part of the experiment, the student has to program the Arduino DUE of the Robonova in order to walk a distance, standing as straight as possible. For the assessment of the student, a smooth and natural movement, like that of a person, is valued. To program this robot, the student must keep in mind especially the servos of both legs, but moving the arms can also help to keep the balance. The task of walking is not as easy as it may seem because the student not only has to define the movements, but also guarantee that the robot has a stable balance.

Before facing up the robot programming with the Arduino IDE and C++, the student has to estimate the values of the key positions for the servos, and the time intervals that are required to run the basic trajectories that are necessary for the desired tasks, like take a step, lift the robot, etc. This can be done with some software application that simulates the Robonova, as for example the free program "RZ1 Action" (Micono Utilities, 2008). After estimating the positions and times, the student can begin the program design. The program must sent the positions previously stored in memory to the servos, in the right time, and repeat a sequence to get the robot walks step by step. The use of an interpolation algorithm is recommendable to smooth the movement. Finally, when the program is running, the student have to readjust the movements to get the robot walks properly.

As optional part of the experiment, the student can consider that there may be big obstacles in the middle of the path by which the robot moves. Thus, he has to modify the program so that the robot can detect obstacles with its sensors, and avoid the objects in its path.

Since the subject in which the experiment was developed had only 5 enrolled students, students were enabled to carry out the experiment individually. Although there was one Robonova only, each student had available an Arduino DUE, a prototyping board, and a set of components such as LEDs, push-buttons, IR sensors, and servomotors. This enabled students to test some functions of the program without the Robonova, and made easier to share the robot.

For the assessment, the teacher asked each student to show the robot running and to explain his program. The quality of movement of the robot when walking, the program structure, and the degree of development of the optional part, were considered for the qualification. Student had about 15 hours to carry out the experiment.

3.4 Follower robot programming

This experiment is based on the GoShield-GR robot (see Figure 6), designed jointly by the online shop GoShield and some of the authors of this work. The GoShield-GR is a line tracker robot specially designed for tracking competitions, a task for which the robot has twenty photo-reflective sensors on its bottom side. The robot's brain is an Arduino DUE.

Taking advantage of the number of devices included in the GoShield-GR robot, different activities are proposed to the student, so that he learns the operation of the robot step by step. This way, he will be able to program increasingly sophisticated functions within the time schedule of the course. The number of activities presented to the students depends on the course in which the robot is used.

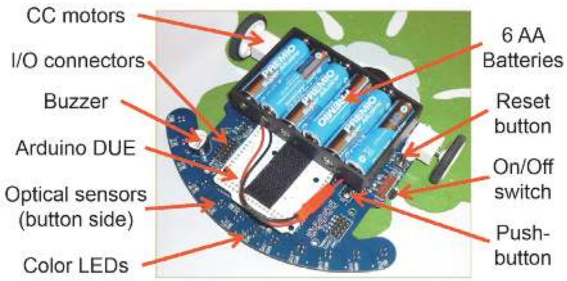

Fig. 6. Follower robot GoShield-GR.

The catalogue of activities developed with this robot, ordered from the simplest to the most complex, is summarized below:

- Hello world, and semaphores. These beginner exercises are intended to program the LEDs so they flash with a certain cadence, or emulate traffic lights with the right timing.
- Morse code. The student has to use the buzzer to generate beeps representing the word "SOS" in Morse code.
- Roulette game. Using the LEDs, the buzzer and the push-button, the student has to design a roulette game where the LEDs turn on sequentially, and the player must try to press just when a particular LED is on.
- Basic line tracker. The robot must follow a line with the minimum oscillation of its direction regarding to the line, at a constant speed. Figure 7 shows a picture of a test of this exercise.
- Speed control. The robot must match the speed of its motors depending on the part of the racetrack in where it is, that is, depending on the straights and curves.
- PID control for the direction. When the robot runs at high speed, it is necessary to apply a PID control to the direction in order to the robot does not leave the circuit.
- Identification of intersections. The student have to add to the program the logic that enables the robot to identify intersections and decide where to go, on a circuit with multiple ways. The intersections are identified by special marks on the track.

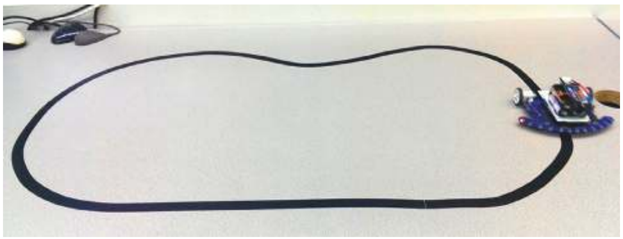

Fig. 7. Follower robot during a test for a basic track.

The robot gives many options and enables students to work with different aspects about programming simple mobile robots. Student have to carry out the activities individually or in pairs depending on the course. In any case, until now, the assessment of the student has been done by testing the working of the student's program on the robot, and the explanation that the student gives about his program.

4. STUDENT OPINION

At the end of the courses, students were invited to make a voluntary survey in order to know their opinion on using Arduino in laboratory experiments, and also to know their level of initial and final knowledge about programming and hardware. Table 1 summarizes the survey questions.

Table 1. Survey questions

N.	Question
Q1	Before starting the course, I had good programming skills.
Q2	Before starting the course, I had knowledge of electricity and electronics.
Q3	I found easy and quick to get started with the software and hardware Arduino.
Q4	The available time to develop the experiments was enough.
Q5	The material available to develop the experiments was adequate.
Q6	Doing the experiments to control real equipment encourages the development of practical contents.
Q7	Using Arduino for experiments, against other control equipment with closed hardware, is a wise move.
Q8	I found it easy to create programs for Arduino with C ++.
Q9	Using Arduino, I have learned or improved my knowledge of how to connect input and output devices.
Q10	The knowledge gained about Arduino will enable me to use this platform to solve other control applications, automation and robotics on my own.
Q11	I bought, or I'm going to buy an Arduino board to solve the experiments of the course or other applications.
Q12	Probably I will use Arduino to solve applications in the future.
Q13	I think Arduino is applicable also in the professional field, in addition to education.
Q14	I missed working with a controller with more industrial characteristics.

The questions Q1 and Q2 refer to the previous knowledge of students about programming and hardware before the course. Q3 to Q5 address the availability of time and resources, as well as the difficulty to get started with the software and hardware of Arduino. The degree to which Arduino motivates conducting the experiments is addressed in Q6 and Q7. Student's opinion about their results are captured in the Q8 to 10. Finally, Q11 to Q14 addressed other aspects.

For each question, students had to answer an integer number between 1 and 5 indicating their level of agreement with the question, where 1 means "I disagree" and 5 means "I totally agree". Figure 8 summarizes the results of the survey.

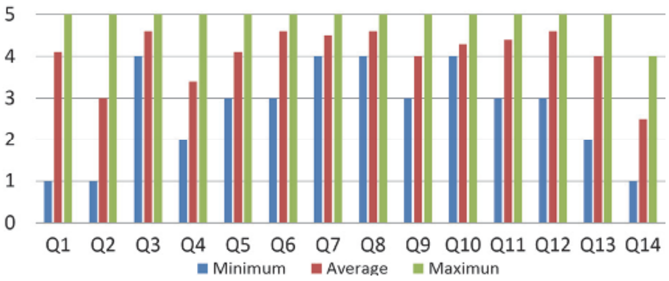

Fig. 8. Results of the survey: degree of agreement of the student (1-5) for each question (Q1-Q14).

When the results are analysed, firstly it can be seen as the minimum and maximum values for Q1 and Q2 confirm the disparity in prior knowledge levels that students had for both programming and hardware. The result for Q3 shows that, in general, students thought their beginning with the Arduino platform was easy. Regarding the availability of time captured by Q4, the student opinion was varied, but not very bad. Students were strongly agree with the fact of using Arduino in laboratory, as reflected in the results of Q6 and Q7. The answers to questions Q8 to Q10 reflect students mainly believed they acquired or improved their knowledge. Also, responses to Q11 and Q12 reflect how students were quite willing to buy an Arduino board on their own. Regarding the use of Arduino on a more professional level, in contrast to the educational, the opinion was diverse, although positive in general, as indicated by the Q13. Finally, the responses to Q14 show that there was some disparity of opinion relating to whether it would have been more interesting to use a controller with industrial features.

5. CONCLUSIONS

This paper describes how authors have managed to bring the Arduino platform to different laboratory experiments of engineering courses at the University of Alicante, which address different aspects of Automatic Control and Robotics. To put in practice Arduino in the labs has been relatively easy, thanks to previous experience of the researchers, as well as, the easiness of starting and using of Arduino platform. Furthermore, good results have been achieved on the interest shown by students and his learning, which has been verified through the academic results and a survey to students.

The Arduino platform can be started and learned quickly. Thus students are focused on the problems of the experiments, such as the development and programming of control algorithms that make the robot or the machine works with the given specifications.

One problem that arises in the teaching of subjects in the Master in Automation and Robotics is that students come from different engineering degrees and they have quite different knowledge levels on hardware and programming. With Arduino-based practices, it has found that students can learn best the skills they lack. Thus, students who knew best aspects of hardware, learn object-oriented programming, while students who have a better basis on computers enhance their knowledge about hardware.

It has also been achieved to take advantage of several laboratory equipment that was available but it did not have a good use, after replacing the control electronics by a suitable Arduino controller. In addition, the low cost of Arduino hardware and its free software encourages students to buy their own devices in order to practice outside the laboratory classroom, and even develop their own projects.

Although Arduino proves to be an excellent educational tool, and may also be useful for many real applications, it is undesirable to leave aside completely the professional equipment. It is therefore advisable to maintain laboratory experiments using industrial and commercial equipment as far as possible, since those are the devices that students will find in the professional field. The students themselves say so. The compromise solution is to combine Arduino platform with real robots or machines, as it has been done in one of the experiments described in this paper.

REFERENCES

Arduino (2015). *Arduino web page*. Online (July 2015): http://www.arduino.cc.

Banzi, M. (2011). *Getting Started with Arduino*. 2nd Edition. O'Reilly, USA.

Barber, R., De La Horra, M., Crespo, J. (2013). *Control* Practices Using Simulink with Arduino as Low Cost Hardware. *Proc. of the 10th IFAC Symposium on Advances in Control Education* (ACE2013). Sheffield, UK. August 2013.

CEA, Spanish Committee of Automation. 2015. *Curso práctico on-line de Arduino Avanzado*. Online (May 2015): http://www.ceautomatica.es/curso-online-de-cea-curso-practico-line-de-arduino-avanzado

Granvillano C. (2014). Arduino as a programmable logic controller (PLC). *Open Electronics*. Online (June 2015): http://www.open-electronics.org/arduino-as-a-programmable-logic-controller-plc/

Ishikawa, M., Maruta, I. (2009). Rapid prototyping for control education using Arduino and open-source technologies. *Proceedings of 8th IFAC Symposium on Advances in Control Education*. Kumamoto, Japan, October 2009.

Kushner, D. (2011). The Making of Arduino. *IEEE Spectrum*. On-line: (May 2015): http://spectrum.ieee.org/geek-life/hands-on/the-making-of-arduino

Micono Utilities. (2008). *RZ1Action: Action making software for Robonova*. Online (May 2015): http://micutil.com/RZE/ROBOMIC.html

Perea, I., Puente, S., Candelas, F.A. & Torres, F. (2010). Nueva tarjeta de sensorización y control para robot humanoide. *Proceedings of XXXI Jornadas de Automática.* Jaén, Spain. September 2010.

Sobota, J., Pišl, R., Balda, P., Schlegel, M. (2013). Raspberry Pi and Arduino Boards in Control Education (I). *Proc. of the 10th IFAC Symposium on Advances in Control Education* (ACE2013). Sheffield, UK. August 2013.

Úbeda, D., Gil, A., Lucas, J.A., Jiménez, L.M., Reinoso, Ó. (2009). Ardilla: plataforma de prácticas docentes on-line mediante Arduino. *Proc. of XXX Jornadas de Automática*. Valladolid, Spain. September 2009.

Valera, A., Soriano, A., Vallés, M. (2014). Low-Cost Platforms for Realization of Mechatronics and Robotics Practical Works. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, Vol. 11 (4), 363-376. Elsevier.

Warren, J.D., Adams, J. & Molle, H. (2011). *Arduino Robotics (Technology in Action)*. Apress, USA.