

# CHAPTER X

## A REVIEW ON HOW TO EASILY PROGRAM ROBOTS AT HIGH SCHOOL

V. GÓNZALEZ, F. R. CAÑADILLAS, R. PÉRULA-MARTÍNEZ,  
M.A.SALICHS, and C. BALAGUER

RoboticsLab, University Carlos III of Madrid.

[veronica.gonzalez@uc3m.es](mailto:veronica.gonzalez@uc3m.es), [felix.rodriquez@uc3m.es](mailto:felix.rodriquez@uc3m.es),  
[raul.perula@uc3m.es](mailto:raul.perula@uc3m.es), [salichs@ing.uc3m.es](mailto:salichs@ing.uc3m.es), [balaguer@ing.uc3m.es](mailto:balaguer@ing.uc3m.es)

In this paper we analyze some of the most used programming methodologies but in the Spanish high school level. There are two main trends in high school programming, blocks and text code. Although there are some relevant studies that show block programming is an easy way to learn the fundamentals, when it has to be applied to a field like robotics sometimes may be very limited. Programming by blocks is not a very mature methodology based on structural programming. Some tools for this kind of programming are Ardublockly, Bitbloq, Scratch4Arduino, or Lego software. Nevertheless, programming by text code has been used since almost the beginning of computer science. The most common programming languages for this kind of methodologies are C++ through using Arduino, or Python. Thus, we show the main features for these programming tools providing a general vision on which to choose depending on what you pretend to teach.

### 1 Introduction

In the last few years technology has evolved exponentially, being this one of the most influential factors in our society. More and more, new generations have to adapt to this great evolutionary leap in order to meet the challenges of technology.

Nevertheless, these new generations have been growing surrounded of the new technologies, acquiring new knowledge about computer science and

engineering. This makes it essential the incorporation of specific branches in science within the educational contents.

Nowadays, educational robotics is one of the most emerging disciplines for learning and reinforcing knowledge about computer science, electronics, programming, mechanics, etc. (Mubin, 2013), (Benitti, 2012). Robots are used as "edutainment" platforms, generating a good learning atmosphere. In this way, a greater interest and curiosity are generated in the student (Eguchi, 2010).

In Spain, this discipline has been integrated within the educational system by means of a new subject called "Technology, Programming and Robotics" (Spain, 2015), representing an important evolution of education.

Nevertheless, within educational robotics, there are different basic skills where the programming is highlighted. To teach programming improves many skills of students, such as logical thinking and resolution process, understanding better the world around them. A successful learning causes an increase in the motivation of the student (Hirst et al., 2003), (Pásztor, 2013), as well as the self-confidence is reinforced (Yamazaki et al., 2015) when the desired objective is achieved. Communicative and social skills are also reinforced (Pap-Szigeti, 2007), since students work collaboratively to solve all the problems raised.

In order to confront the challenge of the programming teaching, teachers need support with different tools and methodologies. A review of some the most used programming tools for the educational field in the Spanish secondary schools are presented in this work. A description of the tool is provided, as well as the methodology used and its main features.

This paper is structured as follows. Section 2 presents a description of the selected programming tools. And Section 3 presents the preliminary conclusions and future works related to this paper.

## **2 Programming tools**

In this section we present some but not all of the most actual and usually used software to teach programming oriented to robotics at Spanish high school. For each, we show a general description to place the reader in a specific context. Next, we briefly present the methodology used, and finally the main features they offer to young programmers and teachers.

## 2.1 Arduino

Arduino software (Fig. 1) is one of the most used in the world for programming low-cost microcontrollers, very often used in robotics. It has been used mostly to teach basic programming or real-time projects.

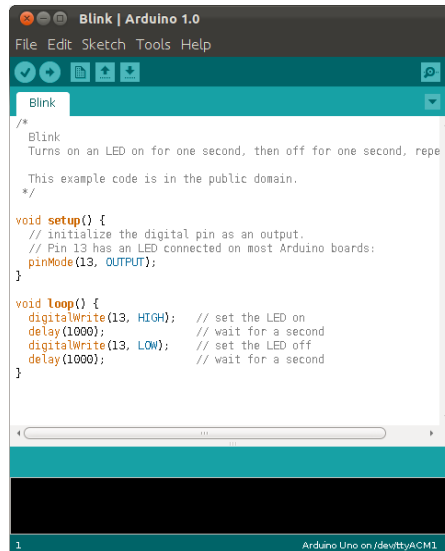


Fig. 1. Integrated Development Environment (IDE) for Arduino software.

Arduino can be programmed directly in C or C++ languages (Arduino, 2016). These kind of languages usually are very difficult to learn if the students do not have previous knowledge in programming. Nevertheless, Arduino uses a simplified version to try to help non-professional programmers.

Distinguished features are, for instance, the *setup* and *loop* functions. The first has the particularity to be called at the beginning of any program. It usually contains initializations and the assignment for the analog or digital inputs/outputs. On the other hand, the second function must contain the main code, but with the particularity that is continuously executing.

## 2.2 Bitbloq

Bitbloq (Bq, 2016) is an IDE developed by the company Bq. Bitbloq can be used to create programs using blocks or code (Fig. 2).

The methodology used by Bitbloq is based on creating some pieces of code by blocks which contain a specific functionality. Moreover, it is pos-



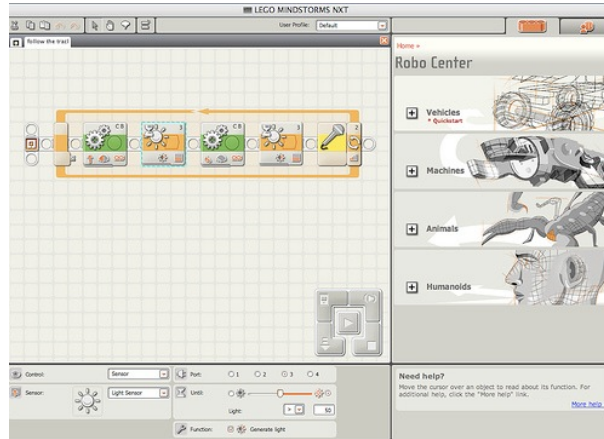


Fig. 3. Lego Labview IDE.

The software is conceptually very simple and together with the simplicity of the interface is ideal for children and professors with little programming experience. But, this type of programming has one inconvenience: the creation of complex actions can be complicated, since the blocks programming does not offer great freedom. Also, unlike other software, this software has a home edition, which is free, and an education version, which has to be paid.

## 2.5 Scratch4Arduino

Scratch4Arduino (S4A) has been developed at Citilab by the Edutec Research Group (Citilab, 2015). This software is based on the software Scratch, developed originally by the MIT.

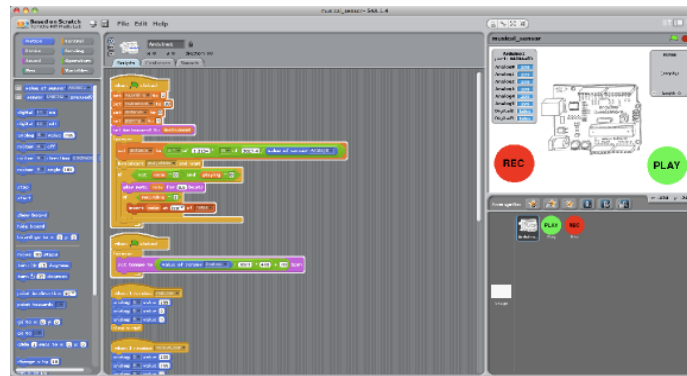


Fig. 5. Scratch4Arduino IDE.

S4A programming uses blocks that represent the same as the elements available for Arduino and Scratch. It has many types of general blocks like events, loops, conditionals, operators, etc. Furthermore, it has specific blocks for the input/output of analog and digital signals, useful for an Arduino board. Finally, it has also available all Scratch blocks for creating graphical and sound elements.

S4A is distinguished by its simplicity (Fig. 5). Thanks to that, it is an ideal tool to start programming. Through a connection to an Arduino board, it can use electronic components for controlling graphic elements. Moreover, it allows to program mobile robots, but being permanently connected by USB, a very important disadvantage.

## 2.6 Visualino

Visualino (Visualino, 2016) is a software with a visual programming environment for Arduino, which is composed of different work environments where the different elements used for programming of the board are distributed. This software allows you to create code via block diagram. This software is Open Source and multi platform, requiring the version 1.6 of the Arduino IDE.

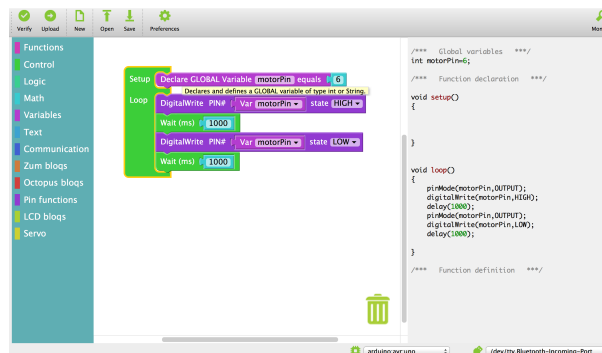


Fig. 6. Visualino IDE.

The programming is done by block flowchart, whose programming is based on Blockly of Google (Google developers, 2016) and Bitbloq of Bq (Bq, 2016). The different elements are organized in work environments: variables, mathematics fields, control blocks, pin functions, etc., in addition to having environments for the ZUM board and Octopus components of Bq. The program can be loaded to the board via USB or Bluetooth.

The interface of Visualino (Fig. 6) is very visual and simple, which contains the basic functions that Arduino IDE has. Unlike other software as

Bitbloq, Visualino is very practical, since inside the code editor you can see and edit the block flowchart, and in parallel, the native code of Arduino is shown. Also, it lets you organize the elements of the code in the different functions, *setup* and *loop*.

### **3 Conclusions and Future Work**

As we have discussed, it is easy to see that software is an important part for teaching robotics and may be a hard task to select which one to use. For that reason, we can reaffirm that software based on blocks is the best option for students who do not have a previous experience but it may be limited when trying to develop more complex tasks. Nevertheless, traditional programming software like Arduino IDE may be the best option for students with previous knowledge using blocks.

As a future work, we will do an essay with students. In this essay, we will compare the difference between two groups of students. The first one will learn to program using only blocks and the second group will use only the traditional methodology. The final goal will be to obtain some metrics useful to test new software systems.

### **Acknowledgements**

The research leading to these results has been obtained in collaboration with CREA Robótica Educativa and the RoboCity2030-III-CM project (Robótica aplicada a la mejora de la calidad de vida de los ciudadanos. Fase III; S2013/MIT-2748), funded by Programas de Actividades I+D en la Comunidad de Madrid and cofounded by Structural Funds of the EU.

### **References**

(Web) Arduino, 2016. Arduino Guidelines and Tutorials. < <https://www.arduino.cc/en/Tutorial/HomePage> >. [Online. Last access: 17/04/2016].

(Journal Article) Benitti, F. B. V. 2012. Exploring the educational potential of robotics in schools: A systematic review. *Comput. & Educ.*, 58(3): 978-988.

(Web) Bq, 2016. Bitbloq. < <http://bitbloq.bq.com/> >. [Online. Last access: 17/04/2016].

(Web) Citilab, 2015. Scratch4Arduino. < <http://s4a.cat/> >. [Online. Last access: 20/04/2016].

(Conference Proceedings) Eguchi, A. 2010. What is educational robotics? Theories behind it and practical implementation. In Proc. Soc. Inform. Technology & Teacher Educ. Int. Conf., Chesapeake, VA: AACE, pp. 4006-4014.

(Book) Evans, B., 2011. Arduino Programming Notebook, Ardumania.

(Web) Google developers, 2016. Blockly. < <https://developers.google.com/blockly/> >. [Online. Last access: 17/04/2016].

(Journal Article) Hirst, J. A., et al. 2003. What is the best programming environment/language for teaching robotics using Lego Mindstorms?. *Artif Life Robotics*, 7: 124-131.

(Web) LEGO, 2016. Aprende a programar. < <http://www.lego.com/es-es/mindstorms/learn-to-program> >. [Online. Last access: 17/04/2016].

(Book) Monk, S., 2015. Fritzing for Inventors. Mc Graw Hill Education.

(Web) National Instruments, 2016. Software de Desarrollo de Sistemas NI LabVIEW. < <http://www.ni.com/labview/esa/> >. [Online. Last access: 17/04/2016].

(Chapter in Book) Pap-Szigeti, R., “Cooperative strategies in teaching of web-programming,” in *Practice and Theory in Systems of Education*, pp. 51-64, 2007.

(Journal Article) Pásztor, A. 2013. Mobile Robots in Teaching Programming for IT Engineers and its Effects. *Int. J. Advanced Comput. Sci. and Applicat.*, 4(11): 162-168.

(Report) Spain. 2015. Decreto 48/2015, de 14 de mayo, del Consejo de Gobierno, por el que se establece para la Comunidad de Madrid el currículo de la Educación Secundaria Obligatoria. *Boletín Oficial de la Comunidad de Madrid (BOCM)*, no. 118, pp. 10–309.

(Web) Visualino, 2016. Visualino. < <http://www.visualino.net> >. [Online. Last access: 20/04/2016].

(Conference Proceedings) Yamazaki, S., et al. 2015. Comparative Study on Programmable Robots as Programming Educational Tools. In Proc. 17th Australasian Computing Educ. Conf. (ACE 2015), Sydney, Australia, pp. 155–164.