On the Use of Robots in a PBL in the First Year of Computer Science / Computer Engineering Studies

J.Oliver
Escola d'Enginyeria
Universitat Autònoma de Barcelona
08193 Bellaterra (Spain)
Joan.Oliver@uab.cat

R.Toledo
Escola d'Enginyeria
Universitat Autònoma de Barcelona
08193 Bellaterra (Spain)
Joan.Oliver@uab.cat

Abstract—**European universities are involved in a structural change in their careers in order to promote students exchange between countries. Taking advantage of this situation, Spanish University Council also is promoting changes in teaching methodologies.**

On the other hand, Spanish universities experience an important decrease of the number of students that arrives to the technical careers. This is mainly due to the fact that, in recent years, students move from the technical studies to the social ones. The reason is that social studies are considered, typically, more affordable.

Finally, students who arrive at Computer Engineering studies often suffer a loss of motivation in their first year due to the fragmentation of subjects that is introduced into the studies. For example, hardware and software topics are explained in different subjects as if they were different disciplines.

With the goal to overcome this situation the authors use a PBL based on robotics in the first course of computer engineering studies in our university. The aim is to motivate students who enter in the computer engineering career.

This paper presents the design of a PBL with robots held in our department for several years. It discusses about the involved subjects and the need of resources, explores the suitability of robots for the PBL, and introduces the results of the two year experience application of this new teaching methodology in our studies.

Keywords- Project Based Learning, Computer Engineering Education, Educational Robots.

I. INTRODUCTION

The implantation of the new degree careers in Spain due to the convergence of university studies in European countries implies a considerable effort to be made by professors in teaching innovation. Following the methodological changes introduced in other European universities, some professors are changing the classical teaching methodologies (based on magisterial classes) towards Problem/Project Based Learning approaches (PBL) [1], [2].

This methodological change in education arrives in a moment in what also is decreasing the number of engineering students in Spain. For example, student entrance in our Computer Engineering studies decreased in a significant number in a few years. The reason of it is due to:

- At first, engineering professionals are not so rewarded for industries and governments as it was in the past.
- In addition, to obtain good qualifications in technical studies it is more difficult than in other studies. Students are more prone to study those careers that imply less effort.
- Finally, the current distribution of the subjects in the first course does not help either to increase the motivation of the students. For example, in the first course of the computer studies, only the 42% of the topics (Table I) can be considered typical hardware/software subjects. And, if computer subjects are divided into hardware and software, it is found that both topics are explained as different subjects and for different professors, as if it wanted to reinforce the idea that hardware and software are different disciplines. In this sense, the main objective of the educational change introduced by the PBL is to improve the motivation of the students and to improve the learning of transversal competences between computer subjects.

In this sense, this paper analyses, designs and presents the results of a two year application of the PBL based on robotics in Computer Engineering. Section 2 and section 4 discuss about the curricula and proposes a scheduling on the use of robotics in the first course of computer engineering studies. Sections 3 and 4 argue about the requirements that educational robots must have to apply for the PBL. Section 5 makes a comparison about the use of two different robots in the experience, the Scribbler robot and an own UABot robot specifically designed for the PBL. In the end, the last section presents assessment and results of the experience.

II. SYLLABUS IN THE FIRST COURSE OF COMPUTER STUDIES

The experience described in this paper is carried out in the first course of Computer Engineering studies of the Universitat Autònoma de Barcelona. Table 1 resumes the subjects covered in the first year of the computer studies. It involves topics on software, hardware, mathematics and some electronics. In fact,

This work was supported by the MQD_2009_ARES_127 [14] of the Catalan Council. Authors also agree the financial support given to the project by the Escola d'Enginyeria and the Institut de Ciències de l'Educació of the Universitat Autònoma de Barcelona .

it is a common first course curriculum in computer engineering studies in Spain.

TABLE I. COMPUTER ENGINEERING FIRST COURSE SYLLABUS

Programming languages & Algorithms (105 hours)	Programming languages. Functions and procedures. Pointers. Files. Algorithm analysis. Search and ordering algorithms. Recursive algorithms.
Data structures (60 hours)	Object oriented languages (C, Python). Lineal data structures. Non lineal data structures: trees. Other non-lineal data structures.
Computer fundamentals (90 hours)	Digital systems: information representation, Boolean algebra, combinational and sequential circuits. CPU fundamentals: architecture, instruction set, address modes, assembler, interruptions. CPU peripherals: memories and memory hierarchy and input/output. Communications
Electronics (120 hours)	Electrical circuit basic laws. Permanent and transient analysis. Semiconductors: diode, transistor, OpAmp. Technology: bipolar and MOSFET. Introduction to data acquisition.
Algebra (105 hours)	Matrices, vectorial spaces, linear applications, rings, polynomial operations.
Calculus (120 hours)	Differential calculus and differential equations, integration, Fourier fundamentals.

It is a tough curriculum for a student who arrives at university with the aim to study computer engineering. Near the half of the topics are not strictly computer engineering topics. An important part of the program consists of mathematics and (in a smaller proportion) electronics. Professors are aware that these topics are not really pleasant to the first year students. In addition, hardware in the first course is not an easy subject either. Many of the students who arrive at university have only used computers for chatting or to play games.

Authors show in this paper that, robotics, as a transversal discipline, can play a fundamental role in the computer engineering program and computer science program in order to increase the motivation of the students. As it has been yet experienced in other European universities [3], the approach presented in this paper focuses on to increase the motivation students in the first year using a transversal PBL methodology. Robotics, seen as a tool, is ideal for this.

III. EDUCATIONAL ROBOTS IN THE FIRST COURSE OF PBL-BASED COMPUTER STUDIES

Robots are used in education in universities for different purposes and employing different educational methodologies. The most normal case, in computer science, is to employ the robot as a tool in programming learning and in areas like vision and artificial intelligence [4], [5]. Recently, professors interested in education have started to use the robot as a tool for the learning of subjects in a transversal way. That is, the robot, is used as a multidisciplinary element employed in the realization of projects that include different disciplines. For example, in [6]-[8] the robot is used in a transversal way in order to integrate in a project different disciplines of computer and electronics science.

In this paper, the robot is viewed as a packed unit that contains all the components and structures to introduce students to computer fundamentals and computer programming in the first year course. It is a perfect tool to motivate computer engineering students that enter into the university. The robot is used in our case to integrate fundamental hardware and software concepts in one project [9]. There are no experiences in the literature in which the robot is used to learn programming (at low and high level) and to learn computer fundamentals, at the same time.

From the software viewpoint, the robot is programmed using a high level language to follow any determined conduct. Typical languages like the classical C or the more actual Ruby or Python can be used to teach algorithms or data structures in the first course. The main routines can be programmed inside the robot, or the robot can be a slave that performs accordingly to actions mastered by a computer. This later case implies that the computer requests the robot for the sensors state, takes decisions, and controls the robot actions.

The hardware kernel of the robot is exploited to introduce computer hardware fundamentals. Educational robots use simple hardware kernels that can be modeled as typical von Neumann or Harvard elemental machines. This CPU usually is built using simple microcontrollers. For example, Scribbler [10] and Moway [11] use a PIC microcontroller. Even though these microcontrollers are simple ones, they contain all the components that included in typical processors: a CPU core, memory and peripherals. But, are these robots useful in our PBL methodology in order to teach computer fundamentals? In this aspect, section 6 presents results about the experience using two different robots. In the first year experience the Scribbler robot was used. As the results were not so good as expected, an own robot was built and used for the second year PBL application [12].

Electronics is the third topic included in the first course syllabus. It constitutes the hardware of any robot. And the robot is a perfect tool to introduce students to basic electronics and electronic devices. However, last years we have assisted to a decrease in the number of hours devoted to explaining electronics in computer studies. Due to it, electronics is used in the PBL only to introduce sensors and actuators to the students

IV. PBL SCHEDULING IN COMPUTER ENGINEERING

Computer Engineering subjects in our university are organized in two semesters per year. In the first semester students learn Algorithm fundamentals (45 hours) and Basics in electronics (60 hours). In the second semester students learn Computer fundamentals (75 hours), Data structures (60 hours) and Electronics devices (60 hours). So, when students arrive at the second semester they have some knowledge about basic programming and electronics.

The fact that Computer fundamentals are taught in the second semester induces that the PBL has to be made in this semester. As the PBL emphasizes the hardware/software link, the main subjects involved are Computer fundamentals (CF) and Data structures (DS). DS is the subject that follows Algorithm fundamentals. Electronic takes part in the PBL, as a secondary subject included in CF.

Fig. 1 shows the scheduling established between the topics in order to coordinate them during the semester. The main characteristics are:

- Theoretical classes of CF and DS are explained in parallel during the semester. At the end of some lessons, student assessment is made through class exercises. At the end of the classes there is an examination of each subject.
- -The seminars about the robot architecture start as soon as the semester begins. Whenever it is possible, problems about the theoretical concepts applied to the robot architecture are explained. Assembler is introduced. As soon as students have some experience in assembler, little robot control drivers are proposed as programming exercises.
- -During the first part of the semester, and in practical classes, students practice with Python (the high-level language used to introduce students in algorithms and in programming) in DS, and assembler in CF.
- -At the middle of the semester, the project is presented to students and must be developed until the course finishes. The project is divided in a compulsory part in which students must control the robot through a labyrinth and a free part in which the robots must dance following the rhythm of some music.
- Final assessment of the subjects is proportional to individual theoretical assessment and the project qualification. The project qualification is unique for the both CF and DS.
- In order to achieve the maximum mutual understanding between students and robots, a robot is loaned per group. With the robot at home students find themselves more attracted to the project. The basic idea is that they must feel motivated for the subject.

The PBL experience is carried out in a group of 20 students. 20 is the maximum number of students that can assist per laboratory session. Therefore, a mobile park of 10 robots is used and lend to the students.

V. PROJECT DESIGN AND EVALUATION

The project comprises elements that include software and hardware topics. This is accomplished using:

- A host to control the robot. In the host (a PC) the students implement high level algorithms to control the robot. The algorithms are responsible of the robot behaviour: line follower, light dependent tendencies, avoiding obstacles, guiding problems, etc.
- A microcontroller-based robot. Students must design some drivers in assembler to agree with the hardware requirements of the PBL. The drivers (written in assembler) must acquire data of the robot sensors and must control the motors.

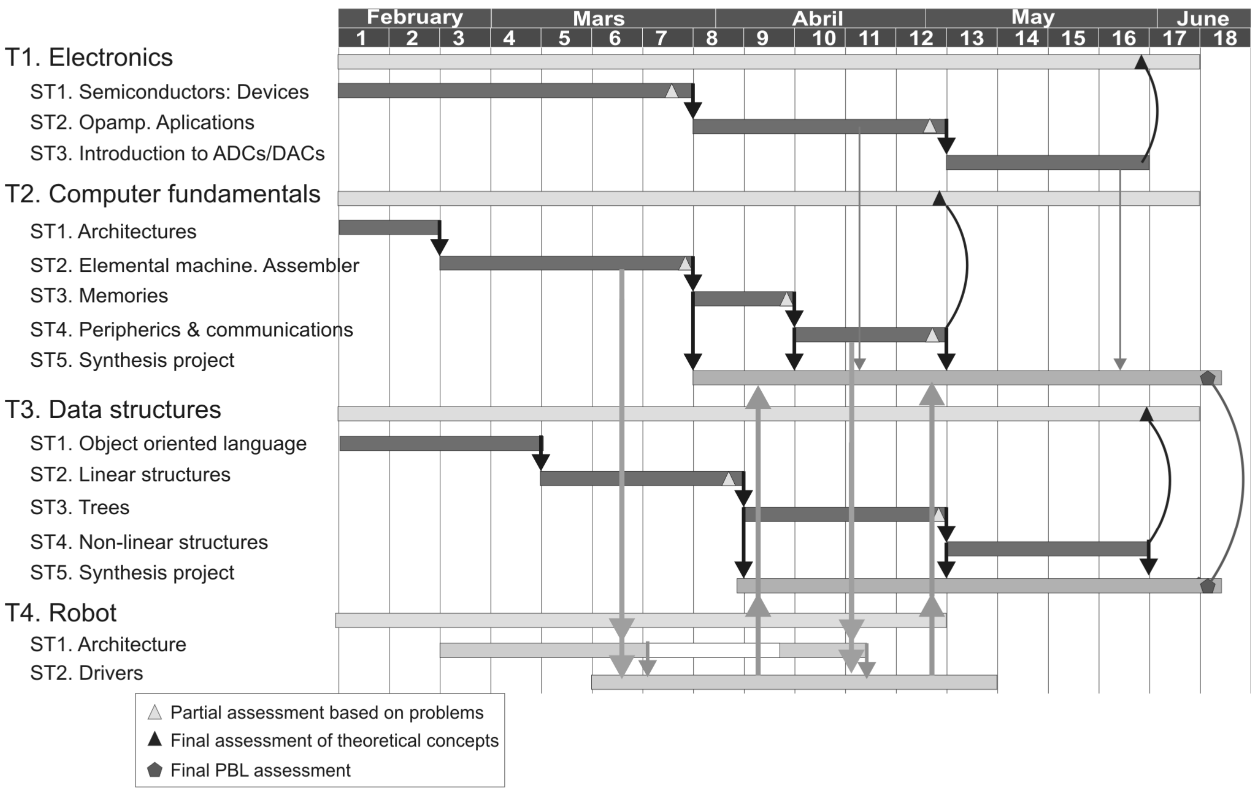

Figure 1. Scheduling of the robot-based PBL

- Of course, the project also involves other topics of computer engineering like, for example, communications at low level using a simple RS232 protocol (usually built in microcontrollers).

- Finally, professors are aware that it is difficult to work in real time. In order to facilitate the work in real time, a setup with a camera is prepared by the professors with the aim to help the robot to follow the path.

The final examination of the PBL is based on the behavior of the robot. It consists of two tests. In the first one, robots must follow lines in a labyrinth based on clues that open doors while the robot moves around the labyrinth. The second test is a free one. Robots have to dance with elegant movements according to the music. Some students become so motivated with the experience that, even they personalize the robot.

VI. ROBOTS IN THE PBL EXPERIENCE

As has been previously commented, the robot that is used in the PBL has to allow the introduction of hardware and software competences. So, its election is crucial in the success of the experience.

The market offers good educational robots that can be used in the design of a PBL for the first course of computer science/computer engineering studies. At this moment, the PBL with robots has been carried on for two years in Computer Engineering studies.

In the first year of the experience, we decided to use the Scribbler robot because it offered a good compromise in programming tools and the architecture is simple. However, Scribbler demonstrated to be no so as good as expected in this PBL. Due to it, authors decided to change the robot in the second year of the PBL application to the new UABot robot [12].

Professors encountered pros and contras on the use of the robots during the application of the methodology:

A. PBL with Scribbler

Scribbler demonstrated to be a good robot to implement PBL in Computer Engineering at the software level. The programming topics can be easily introduced using high level languages. In addition, Scribbler works with the Pyro/Myro environment [13], which allows the development of good software projects. So, Scribbler is a good educational robot with an easy front-end to initiate students in programming with PBL.

However, using it in a mixed hardware/software PBL, shows that it lacks from facilities to access to the CPU. This is due to the following facts:

- Scribbler is a closed product to be used using specific software programming tools. It worked very well in the software part of the project.

- -As used in the old Parallax robots, Scribbler has a kernel based in the PIC microcontroller. Therefore, even though the robot can be programmed using the Myro/Pyro environments, an internal translation to pBasic is made during programming. So, due to the fact that the CPU is not accessible, it is impossible to use the PIC core as a CPU example in the teaching of computer fundamentals.

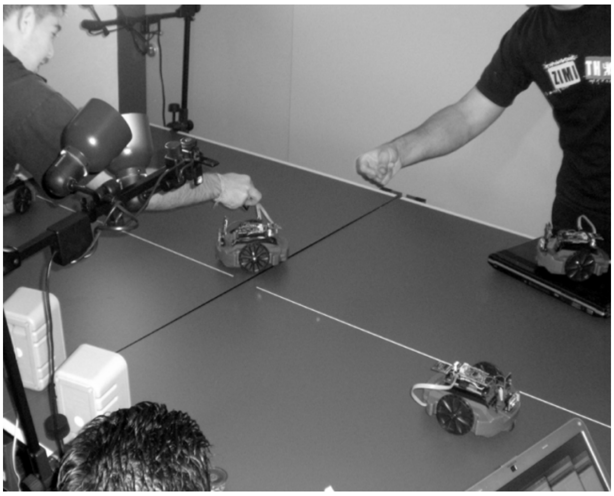

Figure 2. Final practical class using Scribbler robot. On the top of the robot there is the ATmega hardware platform.

- -The authors also believe that the PIC architecture is not suitable to introduce assembler programming skills. Its dependence to an excessive use of the MOV instruction is considered by CF professors that it is not the best choice.

- To overcome these problems, a hardware platform based on an ATmega microcontroller was mounted on the top of the Scribbler robot. The choice of an ATmega device was considered after being proved that the AVR architecture fits better to explain computer foundations (as detailed in the next section). The ATmega hardware platform and the Scribbler robot (based on a PIC) communicated themselves through the Scribbler Hacker Port.

Fig. 2 shows the final hardware setup of the experience using the Scribbler robot. The correct functioning of the hardware attached to the Scribbler was proved. The students acquired the knowledge and competences foreseen in the program course. And professors considered proved the success of the PBL in the learning of computer foundations.

B. PBL with UABot

The first year PBL application with Scribbler demonstrated that the tuning of the robot was too much complicate if it had to be done each year. After working with ATmega devices in the platform, professors were aware about the benefits that could introduce the working with the AVR architecture. So, the choice of ATmega devices was based on the good behavior foreseen for teaching purposes on computer fundamentals:

- The RISC AVR architecture of the ATmega devices is simple and adequate to introduce computer fundamentals. Despite it is based on a Harvard

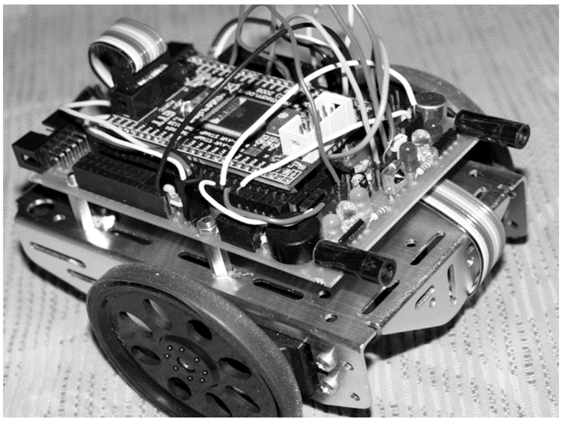

Figure 3. UABot, the robot used in the second year PBL implementation.

architecture (instead of a typical von Neumann one), it does not introduce any learning problem to students.

- -The architecture has a nice instruction set. It is a bit short but it has sufficient assembler instruction set. It is considered to be a good kernel to explain computer fundamentals.
- The AVR architecture has enough addressing modes in order to introduce well addressing computer modes.
- Interruption concept is easy to understand and use in ATmega devices.
- Atmel distributes good free-software to program ATmega devices in assembler and in C.

Fig. 3 shows UABot, the robot built using as a processor an ATmega128 microcontroller and used in the second year course. The robot has the following functionalities: line follower, frontal and lateral obstacle detection, sound processing (introduced using a piezobuzzer and a microphone), leds, light detection, communication using RS232. Displacement can be done using servomotors or DC motors (using LD293 drivers).

The use of this new robot in the PBL methodology has the following advantages and drawbacks:

- The hardware of the robot fits well with the PBL requirements. CF can be explained and assembler drivers are easy to program.
- At software level, the system works in a similar way as it did in Scribbler. Since the main program resides in the PC, the robot can be programmed using the high level language (Python in our case) introduced in theoretical classes.
- Communication is easily established using a COM port. In addition, a serial-Bluetooth device can be used to establish a wireless communication with the robot.
- At this time, we are designing of a nice, simple graphical front-end that facilitates the programming of the robot. However, its lack does not affect the development of the PBL in the first course.

Summarizing, the application of this PBL demonstrated that it is difficult to find good educational robots that could cope with hardware and software requirements at a time. Only personal built-in robots can satisfy these hardware/software imposed requirements sacrificing, in this case, a graphical user interface.

VII. Project Assessment

One group of 20 students per year participated in the PBL. A 93% of all the students that participated in the PBL passed the examinations. The students who did not approved, left the course in the first week. These students considered that working in the PBL implied too much work.

The professors are satisfied with the work carried out by the students. The statistics show that students are also satisfied with all the project and, specially, with the competences introduced in the PBL.

Considering transversal competences, students learnt working in an autonomous way and in group. During their project presentation (made in a public session) some students even disguised the robot.

During the course, students were self-motivated and enjoyed their work. This can be seen from the survey passed to students

The survey to students was passed in the last course day. It asked for the student opinion about the PBL and the work that the PBL implied. Table 2 resumes the results, presented in terms of the percentage of students that answered the survey. The graduation of each question goes from A (which means *a lot / very good*) to E (which means *very few / very bad*).

The survey reflects some important points to be commented:

TABLE II. **Results of a Survey Passed to Students**

	Second year					First year				
	A	B	C	D	E	A	B	C	D	E
Degree of learning	29	65	6	0	0	21	47	32	0	0
PBL difficulty	6	41	53	0	0	16	58	21	5	0
I worked …	0	76	24	0	0	5	74	21	0	0
From PBL I learned …		B	C	D			B	C	D	
...team working		65	29	6			58	37	5	
…to schedule the work		18	76	6			21	68	11	
…comprise HW/SWkernel		76	24	0			47	37	16	
…comprise real-time proc.		71	24	6			37	37	26	
…working at low level		59	41	0			58	32	11	
Professor care		100	0	0			68	26	0	
I would like to repeat the PBL		Y	N				Y	N		
		100	0				95	5		

- The project helped to motivate the students for the subjects, as it was expected from the design of the PBL.
- Professors designed a PBL that it was not simple, but neither complex. The PBL also required a considerable effort to the students. Students, even having one similar view, they got self-satisfied with the project.
- The opinion of the students also reflects that, if they had to start the experience now, they will began it again. This affirmation is very valuable for us. It is the result of a work well done. However, the assertion has to be taken with care because our students are not used to learn using PBL methodologies.

VIII. Conclusions

The paper described the application of a based-in-robots PBL methodology in the first course of the Computer Engineering studies in our university.

The experience, at this time held for two years, was performed using different robots. The reason is the lack of adequacy of the first robot to the requirements. It means, basically, that the robot to be used in the PBL has to allow the introduction of hardware and software competences described in the syllabus of the subjects. In our case, where software and hardware topics are meshed in the project, professors experienced that it became an indispensable requirement.

The main conclusions of the application of the PBL methodology are:

- It is possible to drive students through a problem/project based learning mixing software and hardware in Computer Engineering. In fact, the PBL can be introduced in any technical career with hardware and software similar syllabus to the one included in our career.
- The designed PBL also allowed the students to understand better the mixed hardware/software kernel of computers.
- The project demonstrated that it is difficult to find in the market good educational robots that can cope with the hardware requirements for a first course PBL in Computer Engineering. Our own UABot copes with in-situ low level programming (in assembler) and ex-situ high level programming (using Python or C languages).
- As stated in other PBL papers, authors agree in the fact that the effort that a good PBL demands is enormous.

In what follows, authors detail some guidelines to follow when repeating the experience:

- First of all, a study of the topics to be included in the PBL must be made. The realization of the PBL demands to dedicate more time to topics than that used in classical teaching methodologies. So, in similar teaching times, a reduction of the contents should be foreseen.
- A real scheduling and viability of the PBL must be planned. The schedule must clearly specify when each topic must be taught.
- A lot of time must be spent chosing the correct robot that satisfies with the project requirements.
- And, teachers should remember that they will spent many hours on the project.

IX. References

[1] E. de Graaf, A. Kolmos, “Characteristics of Problem-Based Learning”, *The International Journal of Engineering Education.*, vol. 19, no. 5, pp. 657-662, 2003.

[2] A. Kolmos, H. Algreen-Ussing, “Implementing a problem-based and project-organized curriculum - a cultural change”, *in 2000 The 3rd International Conference for Educational Development (ICED)*, July 22-26, Bielefeld, Germany.

[3] A. Friesel, “Retaining electronic engineering students by project and team-work from the first semester”. *In 2010 Proc. of EDUCON Conf.* pp 1327-13333

[4] D. Kumar, Learning Computing with Robotics, Institute for Personal Robots in Education (IPRE) - http://www.roboteducation.org. Fall 2008.

[5] D. T. Burhans, “A Robotics Introduction to Computer Science”, *ACM SIGCSE Bulletin*, vol. 32, no. 3, pp. 148-151.

[6] M. Carbonaro., M. Rex, J. Chambers, “Using LEGO Robotics in a Project-Based Learning Environment”, *Interactive Multimedia Electronic Journal of Computer Enhanced Learning (IMEj)*, Wake Forest University, vol. 6, no. 1, June, 2004

[7] A. Price, R. Rimington, M. T. Chew, Demidenko S, “Project-Based Learning in Robotics and Electronics in Undergraduate Engineering Program Setting”. Fifth IEEE International Symposium on Electronic Design, Test and Application. Ho Chi Minh City. Mars 2010. Pp 188-193.

[8] M. Otake, R. Fukano, S. Sako, M. Sugi, K. Kotani, J. Hayashi, H. Noguchi, R. Yoneda, K. Taura, N. Otsu, T. Sato, “Autonomous collaborative environment for project-based learning”, *Elsevier Robotics and Autonomous Systems*, vol. 57, pp. 134-138, 2009.

[9] J. Oliver, R. Toledo, E. Valderrama, “A Learning Approach Based on Robotics in Computer Science and Computer Engineering”. IEEE EDUCON Education Engineering. Madrid-Avila. April 2010. pp 1343-1347. ISBN: 978-84-96737-69-3, 978-84-96737-68-6(CDROM)

[10] Scribbler is introduced by Parallax “http://www.parallax.com/tabid/455/Default.aspx” and the Institute for personal Robots in Education “http://www.roboteducation.org/”

[11] Web page for Moway: http://www.roboteducation.org/

[12] J. Oliver, M. Prim, R. Toledo, “Un robot para un ABP en las Ingenierías Informáticas”. Tecnologías Aplicadas a la Enseñanza de la Electrónica. Madrid-Ávila, 2010. ISBN: 978-84-96737-69-3, 978-84-96737-68-6(CDROM)

[13] Myro. http://wiki.roboteducation.org/Myro_Reference_Manual

[14] “ABPbot: Disseny d’un portfoli integrat per a assignatures HW/SW en titulacions d’informàtica emprant una plataforma robòtica”. Project for the Improvement of the Educational Quality at the Universities of Catalonia (MQD’09). Catalonia Council. 2009.