

# A Piedmont SchoolNet for a K-12 Mini-Robots Programming Project: Experiences in Primary Schools

M. Stella De Michele<sup>1</sup>, G. Barbara Demo<sup>2</sup>, Simonetta Siega<sup>3</sup>

<sup>1</sup> I Circolo Didattico Settimo Torinese, v. Buonarroti 8,  
10036 SettimoTorinese, Italy [primocircolo@comune.settimo-torinese.to.it](mailto:primocircolo@comune.settimo-torinese.to.it)

<sup>2</sup> Dipartimento Informatica, University of Turin, c.so Svizzera 185,  
10149 Turin, Italy [barbara@di.unito.it](mailto:barbara@di.unito.it)

<sup>3</sup> Istituto Comprensivo "Fogazzaro,, di Baveno (VB), via Brera 12,  
28831 Baveno - Verbania, Italy [simo.si@alice.it](mailto:simo.si@alice.it)

**Abstract.** PIONEER (PiedmOnt NEt for Educational Robotics) is a schoolnet for K-12 "Educational use of robotics" project originated and carried out by primary school teachers and headmasters. Its goal is to promote Papert's constructionism in a cooperative environment for setting up a model of mini-robot programming experiences in support to the standard curricula covered in school years K-12. Here we concentrate on primary school activities where educational aspects concerned by using small robots fill a long list. In this list there is of course mathematics, but also education to affectivity, creativity, communication, geography, and other. Experiences from our project are described.

**Keywords:** cross-disciplinary activities, inquiry based teaching technique, pupil centered teaching.

## 1. Introduction.

In July 2007 a group of Italian primary and secondary school headmasters signed the agreement "Net for the educational use of robotics" aiming to make use of mini-robot programming to carry out activities of mutual interest in their schools. The project is also called PIONEER (PiedmOnt NEt for Educational Robotics) since the concerned schools are scattered through the Piedmont region. The First Teaching District of Beinasco (Turin), with its headmaster V. Termini, was chosen as the leading institute, and the teacher S. Siega as the educational manager. The net also relied on the cooperation of G. Marciandò, who led the *Robotica Laboratory* of the Regional Institute for Researches in Education (IRRE), and of G.B. Demo from the Dipartimento of Informatica of the University of Turin.

PIONEER aimed at promoting Papert's constructionism in a cooperative environment for setting up a model of mini-robot programming experiences in support to the standard curricula covered during the K-12 school years [1]. All the educators who are members of the net had already been involved in ICT projects in different times and kinds of activities. In particular, most of them had been cooperating with G. Marciandò in his *Robotica Laboratory* activities promoted by

Piedmont IRRE. This institute was going to change its mission in summer 2008. Thus the idea of connecting several schools in a network had administrative and financial reasons, but also, and most importantly, educational goals primarily originated from teachers working in the field. They selected a schoolnet organization in order to gather experiences from different institutions and to create both a shared pedagogical environment and a common professional guidance. This conceptual change in school organization was deemed very important particularly in a situation where the administrative rules and the educational guidelines are often changed. The common environment is likely to provide a greater stability.

The educational researchers grouped in the net had already shared, in their previous activities, the belief that they can fruitfully take advantage of their common cultural background based on psychology and pedagogy [2, 3] to meet the current technology challenges. This mingling between tradition and innovation has given rise to a project for an original educational methodology where technology is used in order to offer children the pleasure to learn every subject "beyond the pencil and the book" [1]. In the drafts of a PIONEER Technical Group meeting we read that the net aims at "developing, documenting, evaluating and disseminating K-12 mini-robot-based educational activities that must be concrete, feasible and strongly affecting the children daily curriculum, following Marcianò's idea of robotics as a learning environment" [4]. Teachers also wished an experience exposing pupils to the method during several years of their education. Thus a K-12 project was decided where robots should be used with continuity rather than in occasional laboratory hours. Though some junior and senior secondary schools are also involved, most PIONEER experiences up to now concern kindergarten and primary schools, probably because primary school teachers are most accustomed to cross-disciplinary activities, and because innovative methods of teaching standard subjects are considered more successful if applied from the very beginning of the children school life.

As said above, several members had already been involved in activities connected with mini-robot programming before the net was set up. To give an idea of these early experiences, in Section 2 the teacher S. Siega sketches activities in a fourth-grade class in Baveno primary school during the year 2003/2004 when a single Lego RCX robot was used. These can be considered the first net experiences because S. Siega currently is the PIONEER pedagogical manager. Sections 3 and 4 concern recent activities. In Section 3 M. S. De Michele describes her 2007/2008 experiences in a second-grade class with the Bee-Bot, by the TTS-group, programmable by pressing buttons on its back. Several teachers in PIONEER schools have used the Bee-Bot. For lack of space we sketch here only De Michele's activity, which is interesting because she was novice to programmable robots. Her experience can be useful to teachers envisaging to approach robotics with their pupils of the lower grades, and can inspire confidence that good results are achievable when pupils and teachers learn together. Section 4 is a short overview of recent activities where students write programs. From about the beginning, PIONEER schools have used different types of robots and programming languages. Among programming languages used to program the RCX Lego robot, Siega and her schoolchildren in 2004 began to use the NQC (Not Quite C) textual language, proposed by D. Baum [5]. Most pupils found using iconic languages less clear than using the textual NQC, particularly when icons have to be connected in a behavior description. As for the teachers, they observed that using the

same textual format both in programming and in natural language reading and writing allows interesting exchanges between the linguistic competences and those needed to conceive and develop robot programs [6], [7]. Thus G. Marcianò, wishing to have a children-oriented, easy-to-use robot programming language, defined the textual language NQCBaby, which is a Logo-like language following the mini-language approach [8]. NQCBaby is briefly described in Section 5, where also a short description is given of the software tools developed around it for a better use by pupils and teachers.

Future directions of PIONEER work are given in the conclusive Section 6.

## **2. Early experiences: from 2003/2004 to the net.**

As we have written in the Introduction, the teacher S. Siega is the current pedagogical coordinator of the network of Piedmont schools involved in the educational use of robotics. Since 2003 she began to program one RCX Lego Mindstorm in a fourth grade primary class, after having worked with her pupils using Microworld software and the Logo language. The pupils criticized both the RCX manual, which presents a too limited variety of examples, and the programming language, which was found to be not enough user-friendly. Pupils also said that the "robot" concept should apply not only to an object built using Lego bricks, but to any programmable, autonomous and mobile object. Due to this observation the awareness arose that by using different kits a larger number of children, belonging to different ranges of age, could be involved in robot activities. This is the important result that the schoolnet today can be proud of having achieved.

After the 2003/2004 single-class experience, G. Marcianò proposed the project "Educational use of Robotics" for the three school years 2004-2007. Three schools agreed with his plan: Siega's Istituto Comprensivo of Baveno, the Direzione didattica of Tortona and the Istituto tecnico of Novara. The latter is a senior secondary school. The project has made possible to study and, above all, to test the idea that robotics in school should be regarded as a subject pertaining not as much to the "new technologies" area, rather to the "new possible teaching methods" in a school-laboratory, i.e., a school environment where to "learn how to learn".

The first experiences were often initiated almost by chance, but they were quickly consolidated owing to the children's greatly positive response. Scientific measures of possible recognition and validation of educational applications have been proposed and documented [9]. In the meanwhile, the NQCBaby language was developed as a new instrument specifically designed for an educational use of robots in the school.

After three years, the natural evolution of the IRRE project was the creation of the network of Piedmont schools to which this paper refers, because of the spreading of good practices produced in nearby schools. The network shares in its work the realization of what S. Papert wrote: "The child programs the computer and, in doing so, both acquires a sense of mastery over a piece of the most modern and powerful technology and establishes an intimate contact with some of the deepest ideas from science, from mathematics, and from the art of intellectual model building. ... Programming a computer means nothing more or less than communicating to it in a

language that it and the human user both *understand*. And learning languages is one of the things children do best", from the Introduction of [1].

The use of different languages enables schoolchildren to communicate with different robots. If a pupil likes better to use icons, she/he may use them rather than a textual language: what matters is the concept of programming. Children enter commands to a robot and then check if the robot performs the intended action. The immediate feedback allows them to understand if they have done a good job OR IF they have made an ERROR. In this case they can correct and change the action of the robot immediately!

Practicing a method of learning by doing is a peculiarity of the PIONEER network of schools. This allows pupils to understand what they are doing rather than to learn mostly by heart. "When a student learns something in school, the most important thing is not the content, but the method of learning, which can be applied again in the future"<sup>1</sup>.

### 3 First programming activities using the Bee-Bot.

The Bee-Bot, produced by the TTS group, is a big bee that can be programmed by pressing buttons on its back for moving forward, backward, turning left, right, starting to move or deleting previous commands. As we have written, several teachers in PIONEER schools have carried out activities with the Bee-Bot. Here we recall fragments from the report that M. Stella De Michele wrote to document the activities that she, new to robots, carried out with her second grade schoolchildren during the last (2007-2008) school year. M. Stella is specialized in teaching humanities, but in 2007 she promptly agreed to become in charge of the robots experiences in her school and to use the Bee-Bot with her seven-year-old second-grade pupils, so as to start learning with them how to program mini-robots and how to use them for standard curriculum teaching.

"I think it necessary that school confronts with the technology to which children are exposed in everyday life. I had used computers for some years with my classes, but I was curious to use an object that can move around following your description of a path, given either by writing a textual description or by pressing buttons as in the Bee-Bot case.

Our story with robots began when schoolchildren found one Bee-Bot on our classroom windowsill. We tried to understand why this bee, different from those we are used to, was there. Possibly she had got lost because of the pollution and had come into our classroom to rest. The bee was greeted, given a nickname (Maya), and the children introduced themselves. They soon found out that by pressing the buttons on its back they could teach it how to move on the floor (i.e. in a two dimensional space): going straight or turning left or right exactly of a **quarter of a cake** (second grade pupils have not yet dealt with angles and their measures). We discovered that the bee could **stroll around** the classroom by pressing more buttons in a sequence and then the **go** button. When a child asked whether we could make the bee go from one child to another, i.e., from a starting point to an end point, some of the classmates

---

<sup>1</sup> [16], page. 3.

observed that buttons should not be pushed randomly, as they had been doing when they wanted the bee go strolling on the floor.

Making the Bee-Bot go from one child to another requires children to take decisions: first we must decide where to go from where, i.e. design a path connecting two points. Different children may suggest different paths. We take some of them into consideration, and for each path we decide which buttons to press and how many times. Then we verify if the Bee-Bot moves the way we want. If it does not, that means that we have given the bee the wrong teaching, and in order to change its behaviour we have to modify, by successive adjustments, the sequence of buttons to be pressed. If we want to teach the bee a wholly new behavior, we have to take some time in *planning* exactly what we want the Bee-Bot to do.

We have to be precise and discover how far the bee moves at each step and so on. Thus we introduced the concept of **measure**: if Maya has moved for a while, how can we tell how far she went? How do we measure the distance covered? First we used several non-conventional tools, then we chose the ruler, because it is a common tool and gives a number for the quantity of space covered at each step. To determine how far the bee goes with a given number of button pressings, one child suggests the arithmetical operation of adding (the length of one step to the previous ones), another suggests multiplying the number of steps times the space covered by the single step). Thus the teacher recalls that both are right because product is defined by means of the sum, and a child shouts: «Teacher, is this robotics or math? ». Children drew the paths on their exercise books with squared sheets, and at this point the introduction of the Cartesian plane, suggested by some of them, turned out to be perfectly natural." After the experience of one year we are not proposing here a generalization. The above activity report is an excerpt of a class journal, which we will use to compare and discuss our experience with the ones of other PIONEER colleagues with lower-grade classes. Though we have not yet performed a specific evaluation of children's achievements, we can compare the abilities acquired by them with those of all the other pupils in the same age we had in over twenty years of teaching. We notice that, by using a Bee-Bot, lower-grade pupils develop skills for:

- counting and logical thinking;
- solving topological problems;
- accessing problem-solving education;
- getting used to an *inquiry-based learning (and teaching) technique* even in activities, as those described above, perceived as close to mathematics. This is an uncommon experience in lower grades [10].

In addition, we perceive that pupils have a playful approach to robotics and begin to understand what programming a robot is. We are planning in our school an evaluation session adapted from the one described by Kurebayashi for older students [11].

It is important to point out that the above activities naturally involved several educational aspects other than the more obvious ones concerning mathematics. For example, we considered different reasons why the bee had come into our classroom. The environment pollution was considered an acceptable reason, and children all together wrote the "Bee-Bot Story". Moreover, different forms of pollution, causes, consequences and remedies were discussed: thus some environment-preserving education has been covered. Pupils introduced themselves to the bee, gave it their welcome while holding it in their hands, gave it a name, involved it in their school life

by showing concern for the new "thing": this is Education to affectivity and to diversity. For each robot session we had a discussion time followed by a self-activity where every child wrote down a few lines on what we had done. Children learned by doing activities with a concrete object and teachers learned with them.

#### 4 Primary school programming languages

The current PIONEER methodology includes the use of four different kinds of robot kits, with different features and functions that allow different kinds of learning: the Bee-Bot, the Scribbler by Parallax, RCX and NXT by Lego. Children can use five programming languages, according to their skills but also depending on the robot kit that is being used. Also, through a long-lasting cooperation with B. Demo of the University of Turin, a compiler for the NQCBaby language is available with a very simple and user-friendly interface that children have immediately accepted. Pupils describe the desired robot behaviors in NQCBaby programs that are translated into NQC [5]. Thus they keep a competent use of the language primitives and are enabled to learn.

After four years of experiments, enrichments and modifications of the methodology, the schools involved in the PIONEER project may claim that the educational use of robotics, in favorable circumstances, allows kids to attain powerful skills for their cognitive development. Schools with longer time experience have been able to observe that students involved in robotics activities for six school years, i.e. from their primary school second grade to the junior secondary third grade, are able to solve meaningful problems and write the related programs with robots equipped with sensors and actuators.

So, in the last year it has been possible to experiment both in the kindergarten and in the lower grades of the primary school the *Bee-Bot*, the bee-shaped robot, a programmable machine that involves children in the use of the first computer procedures, as was explained in the previous section. After the Bee-Bot, it is possible to work with the *Scribbler*, the blue turtle (also called "the messy robot") that aims at simulating what the children program in Logo with Microworlds.

In the upper grades of the primary school, the Lego Mindstorms bricks allow to use various languages (both iconic and textual), to implement paths with several types of sensors, and to find different meaningful solutions to given problems. To conclude, in junior secondary school, activities using the most recent Lego NXT robot, more complex and refined in its components, meets the different needs of teenagers, without forgetting the application of the PIONEER project methodology aiming at the student cognitive development rather than at promoting coding skills. During 2007/2008, in Baveno school, four different types of mini-robots have been used; they have been programmed by means of six different languages, depending on pupils' grades and previous experiences. Such numbers show the growth of experiences with robot use in Baveno school during about five years from first activities. Students educated through robots in the schools of the network come out having an idea of the ubiquitous technology not as a black box or a magic, rather as a world they can control because they understand it.

## 5. A textual programming language and related software tools

An integrated development environment (IDE) and a compiler of the programming language NQCBaby into the NQC language for the RCX robot are currently available to schools, while a compiler of NQCBaby for the NXT robot is being developed by students of the University of Turin, Dipartimento di Informatica [10]. A platform-independent method is a PIONEER future aim for providing a single child-oriented textual language, to be used for programming all different robot types. This language is based on the NQCBaby language, therefore based on the native children tongue and, following the Logo philosophy, with primitives coming from the children language. As a matter of fact, our approach consists in allowing children to use easier languages, rather than building tools to make easier the existing languages, such as the "wood icons" for the iconic programming language proposed in [12]. The PIONEER methodology defines an NQCBaby gradual introduction to schoolchildren with language enrichments from children at beginning-to-write level that use NQCBaby0 to NQCBaby6 level, usually for junior secondary school. NQCBaby0 is the kernel of the language. It is the textual form of the button commands on the Bee-Bot back.

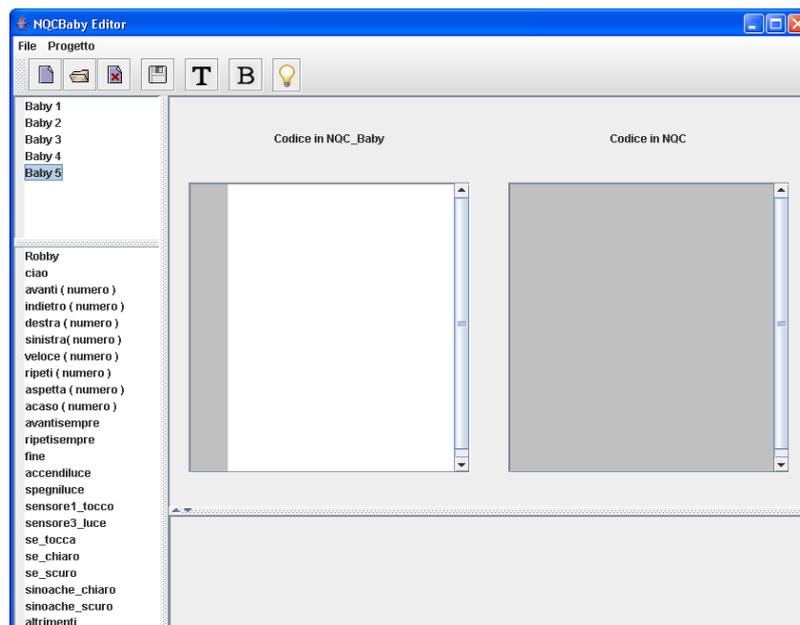


Figure 1. PIONEER IDE window

Children write their NQCBaby programs using the Integrated Development Environment (IDE) interface shown in Figure 1. The "white board" in the center of the window is where children write their NQCBaby code. On the top left side, we have the toolbar where the button T is used for translating the NQCBaby code. Errors are reported at the bottom with the code line. Language levels are written on top of

the left column indicated as Baby1, Baby2 and so on. Each successive level encapsulates the previous ones and deals with a different robot needing/allowing new primitives or new hardware components, sensors or actuators. Ordered introductions of new components, for example sensors, and related primitives for using them in robot-programmed behaviors go along with the progress of schoolchildren's logical and linguistic abilities [7] [10]. In this way, robot programming fits the learning achievements and becomes an original tool that contributes to strengthening the advances in standard linguistic and logical curricula. The language grows with children, with their school education and with what they can/want to do with their different robots.

Following the mini-language approach, NQCBaby is not a complete language, because our purpose is not that children become skilled professional programmers, rather that they have the opportunity to use concrete robots for doing **concrete programming**, i.e. for solving problems by using the basic yet complete structures of algorithmics, as from Jacopini-Böhm theorem [13], [14].

When the RCX robot is used, NQCBaby is translated into NQC. When an NXT robot is used, NQCBaby is translated into the NXC (Not eXactly C) language by means of a compiler under development. For NXT the last extension of the language provides primitives that better fit the NXC language, target of the translation. In Figures 2 and 3 two NQCBaby examples are shown, in an English translation for the sake of comprehension.

<pre> Hi Robbi   speed(3)   forward(100)   speed(7)   backward(100)   repeat(3)     right(90)     left(90)   end   repeat(2)     backward(10)     forward(20)   end thanks-by-a </pre>	<pre> task main() {   SetPower(OUT_A+OUT_C,3);   OnFwd(OUT_A+OUT_C); Wait(100);   SetPower(OUT_A+OUT_C,7);   OnRev(OUT_A+OUT_C); Wait(100);   repeat(3)   { OnFwd(OUT_A);OnRev(OUT_C);     Wait(90);     OnFwd(OUT_C);OnRev(OUT_A);     Wait(90);     Off(OUT_A+OUT_C);   }   repeat(2)   { OnRev(OUT_A+OUT_C);Wait(10);     OnFwd(OUT_A+OUT_C);Wait(20);     Off(OUT_A+OUT_C);   }   Off(OUT_A+OUT_C); } </pre>
--	--

Figure 2. First NQCBaby example

The NQCBaby program shown in Figure 2 describes a robot strolling around: it might be a program where pupils check primitives of the language without a specific goal. The left-hand column is NQCBaby translated into English, the right-hand column is the same code translated into the NQC language.

A second program in NQCbaby is shown in Figure 3. We find in it the function **flip-coin** that in both the NQC and NXC languages corresponds to a call of the function **random**. The program describes the behavior of a robot that goes forward for

a while then chooses to turn left or right depending on the result of flipping a coin. The NXC version of the program is on the right column. By comparing the NBCBaby and the target code versions of programs here shown, we have examples of what we mean by saying that NQCBaby is a children-oriented rather than robot-oriented language.

<pre> Hi Susi repeat-always   speed(75)   forward(500)   if (flip-coin = heads)     right(1);   else // it's cross     left(1);   end; end-repeat; thanks-bye </pre>	<pre> task main() {   while(true)   { OnFwd(OUT_AC, 75);     Wait(500);     if (Random() &gt;= 0)     { OnRev(OUT_C, 75); }     else     { OnRev(OUT_A, 75); }     Wait(360);   } } </pre>
--	--

Figure 3. Randomly going left or right

## 6. Conclusions

Experiences here described began with one teacher and a small number of pupils. Nowadays, the project counts about 100 teachers in 17 different primary schools for about 1000 schoolchildren from the age of 5-6 to 13. Future activities will concern evaluating the competences acquired by these already fairly large number of students. Moreover, teachers in the net will continue developing the methodology but also using it as an everyday teaching tool in several disciplines, which is one of the peculiar goals of the project. An effort is also toward extending the number of junior secondary schools involved, in order to follow the students that have programmed robots in primary school as they progress in their education life. The homogeneousness and the common support of the pedagogical methods while carrying out robot activities, though the geographical distribution and the different types of schools involved, is another peculiar aspect of our project.

Besides all the cross-disciplinary innovative activities that students will experience with robot programming, other important results specifically concern digital literacy. PIONEER pupils learn how to write in a formal language, what an integrated development environment tool is, and how to use the one we implemented specifically for this project. By using different translators for different robots, they acquire the general concept of a translator, and of its error-finding action. We can definitely say that their digital competences are to those of pupils only using an Office suite or a similar one, what the musical technique of piano players is to the one of stereo music listeners, following the *Pianos Not Stereos* paper by M. Resnick, Bruckman and Martin [15].

Other activities in primary schools will concern inquiry-based teaching techniques that also look possible in scientific subjects, particularly in mathematics. Some hints have been given in Section 3. This would be quite a positive change with respect to often currently used teaching techniques that present mathematics as a mechanical exercise, particularly in primary schools, but unfortunately also in secondary schools where, for example, solving problems of Euclidean geometry is disappearing.

**Acknowledgments.** Many thanks to all the members of the PIONEER project and to the schoolchildren who learned robot programming with us. Thanks to the University of Turin students who designed and implemented software tools here mentioned.

## References

1. Papert, S.: *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, New York (1980).
2. Vygotskij L.S., *Thought and language*, Ed. Universitaria G. Barbèra, Florence (1934).
3. Vegetti M.S., *Man psychology: for a science of a socio-historic education*. In: Liverta Sempio O., Vygotskij, Piaget, Bruner. *Concepts for the Development*, Raffaello Cortina Ed., Milan. (1998).
4. Marciandò, G.: Robotics as a learning environment, In: *Didamatica Conference 2007*, 22--30, Cesena (2007).
5. Baum, D.: NQC language, <http://bricxcc.sourceforge.net/nqc>
6. Marciandò, G.: Robotics languages for schools, In: *Didamatica Conference 2006*, 185--197, Cagliari (2006).
7. Demo, G.B., Marciandò, G., *Contributing to the Development of Linguistic and Logical Abilities through Robotics*, in 11th European Logo Conference, pp. 46, Comenius University Press, Bratislava (abstracts only, full papers at <http://www.eurologo2007.org/proceedings>) (2007).
8. Brusilovsky, P., Calabrese, E., Hvorecky, J., Kouchnirenko, A., and Miller, P. *Mini-languages: A Way to Learn Programming Principles*. *Education and Information Technologies* 2 (1), pp. 65-83 (1997).
9. Marciandò, G., Siega S., *Informatics as a language*, in *Proc, Didamatica 2005*, Potenza (2005).
10. Demo, G.B.: *Programming Robots in Primary Schools Deserves a Renewed Attention*, In: *Proc. First World Summit Knowledge Society*, Athens, 24-28 September, (2008)
11. S. Kurebayashi, S. Kanemune, T. Kamada, Y. Kuno, *The Effect of Learning Programming with Autonomous Robots for Elementary School Students*, in 11th European Logo Conference, pp. 46, Comenius University Press, Bratislava (abstracts only, full papers at <http://www.eurologo2007.org/proceedings>), (2007).
12. Horn, M. S., Jacob, R. J. K.: *Tangible programming in the classroom with Tern*, In: *CHI '07 Conference On Human factors in computing systems*, 965-1970, San Jose (2007).
13. Böhm, C., Jacopini, G.: *Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules*, *J. Communications of the ACM*, 9, 5, 367--371 (1966).
14. Demo, G.B., Marciandò, G., Siega, S.: *Concrete Programming using Small Robots in Primary Schools*, In: *8th IEEE International Conference on Advanced Learning Technologies*, pp. 301--302. IEEE Press, New York (2008).
15. Resnick, M., Bruckman, A., Martin, F.: *Pianos Not Stereos: Creating Computational Construction Kits*. *J. Interactions*, 3, 6 (1996).
16. J. Kopciowsky, *The learning process according to Feuerstein*, La Scuola, Brescia (2002).