Open access • Journal Article • DOI:10.1109/MRA.2016.2636372

Bringing Robotics to Formal Education: The Thymio Open-Source Hardware Robot — Source link

Francesco Mondada, Michael Bonani, Fanny Riedo, Manon Briod ...+3 more authors

Institutions: École Polytechnique Fédérale de Lausanne

Published on: 20 Feb 2017 - IEEE Robotics & Automation Magazine (IEEE)

Topics: Open source hardware, Educational robotics, Mobile robot, Robot and Robotics

Bringing Robotics to Formal Education: The Thymio Open-Source Hardware Robot

Journal Article

Author(s):
Mondada, Francesco; Bonani, Michael; Riedo, Fanny; Briod, Manon; Pereyre, Lea; Retornaz, Philippe; Magnenat, Stéphane

Publication date:
2017-03

Permanent link:
https://doi.org/10.3929/ethz-b-000169253

Rights / license:
In Copyright - Non-Commercial Use Permitted

Originally published in:
IEEE Robotics & Automation Magazine 24(1), https://doi.org/10.1109/mra.2016.2636372

Bringing robotics into formal education using the Thymio open source hardware robot

Francesco Mondada[1], Michael Bonani[2], Fanny Riedo[2], Manon Briod[1],
Léa Pereyre[1], Philippe Rétornaz[1] and Stéphane Magnenat[1]

I. Introduction

Mobile robots are valuable tools for education because of both the enthusiasm they raise and the multidisciplinary nature of their technology. Because autonomous mobile robots sense the environment and take actions based on their perception, they seem to display *intentions of their own*. This fascinates the users of the robot and creates feelings of accomplishment and power when these users engage in creative activities with the robots. The permeating presence of robots in science fiction and their projected use in our society increases these emotions by giving a sense of touching the future. Children can spend hours looking at a robot interacting with its environment and adults can approach robots as a new hobby that replaces radio and electronics kits. Robots embed various technologies and therefore give access to a wide range of fields, such as complex mechanics, sensors, wireless transmission, mathematics, computer science, etc. The emotional potential of mobile robots and their varied technology makes them a potential ideal tool for education.

However, robots are still not as widespread in schools as they could be, for several reasons:

1) Although there are many robotics projects developing innovative and interesting educational robots, few reach a sufficient maturity to become distributed and accessible to schools.
2) A robot performing interesting behaviors is a complex piece of technology and therefore expensive. Indeed, for having an educational value and providing an interesting level of interaction, a robot must embed a wide set of sensors and actuators. Existing platforms with these features cost several hundred Euros. This prevents most schools, which have a limited budget for equipment, to acquire interesting robots.
3) Introducing robotic tools into teaching activities requires investment in time and training for the teachers. In Europe, despite a trend to increase the use of technological tools in the learning process, teachers are still insufficiently trained and are reluctant to introduce these tools in their teaching activity [1]. Therefore, to be accepted by teachers, robots must be both accessible with minimal effort and accompanied by well-prepared educational material. Moreover, infrastructure should be provided to allow the sharing of educational material among teachers.
4) Robot construction, use, and programming is often perceived as a boyish activity in our society [2], [3]. This strongly limits the potential of robots as general-purpose educational tools, especially in schools.
5) Finally, many teachers are reluctant to follow volatile trends, especially if these are based on purely commercial arguments. Teachers prefer to invest in stable tools, and this is the opposite as the trend of current consumer technology, which is often very volatile and based on continuous renewal rather than on durability.

Open source hardware projects can address several of these issues in a different way than closed-source purely commercial products. By *open source hardware* we mean, following the definition of the Open Source Hardware Association[1], "hardware whose design is made publicly available so that anyone can study, modify, distribute, make, and sell the design or hardware based on that design". In this paper, we show that this concept, implemented in the Thymio project gathering a community of users, developers and manufacturers, brings a strong added value to the robot and to the educational methods. In addition, we compare our experience with other robotics open source hardware projects not focused on education, and highlight challenges and opportunities specific to education.

II. Related Work

There is a large number of publications presenting educational robots with different prices and features, from low-cost systems for instance gtargeting education in Africa [4], [5] to extremely sophisticated humanoids [6], [7]. Among those, only a handful are commercially available, limiting their validation by educational scientists, who are typically not roboticists. This results in 90 % of publications about validation of educational results made on LEGO® Mindstorms® [8], a widely available commercial product.

*This research was partially supported by the Swiss National Center of Competence in Research "Robotics" (Thymio robot development and deployment), partially by GebertRuf Stiftung (design of accessories), partially by the Swiss NSF project CRAGP2 151543 "Robotics in schools", and partially by the EU-FP7 project ASSISIbf, no. 601074 (survey on open hardware). Many thank to Luc Bergeron and his team at cal.ch for the industrial design of Thymio; Didier Roy, David Sherman and all the INRIA team for their contributions and diffusion in France; Gordana Gerber for the educational material; all the Mobsya team for the effort in production and sale; and Fabian Hahn for the port of Blockly.

[1]Francesco Mondada, Manon Briod, Léa Pereyre, Philippe Rétornaz and Stéphane Magnenat are with the Laboratoire de Systèmes Robotiques, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland firstname.lastname@epfl.ch

[2]Michael Bonani and Fanny Riedo are with Mobsya Association, Ecublens, Switzerland michael.bonani@mobsya.org and fanny.riedo@mobsya.org

[1] http://www.oshwa.org/definition/

The latest version, EV3[2], is expensive (≈ 400 $) but offers a wide range of possibilities, especially at the mechanical level using LEGO® bricks and at the software level with its graphical programming environment. In comparison, the BeeBot robot[3] is a cheap (≈ 90 $) mobile system. However, it is not a real robot but rather an automaton: There is no perception of the environment, just a good encoder for precise movements and a simple programming interface based on buttons giving directions for the movement.

Among widely used robots, none is part of an open source hardware project. In niche projects, only few open source hardware educational robots are used in schools: Scribbler2, produced and sold by Parallax[4] (≈ 180 $), is a 188 mm large robot, designed to run on the ground and equipped with few light sensors, one distance sensor, two ground sensors and few LED displays. It runs on standard AA batteries and has a hacker port for interfacing electronic extensions. It is programmable with a graphical or a textual code interface. The main weakness of Scribbler2 is its performance per price ratio and the limited compatibility with other systems. Moreover, there seems to be no active community around its development. The e-puck [9] robot is a robot targeting university-level education. Well equipped with sensors and actuators, modular and compact, it can be programmed with industry-standard environments. Several simulators allow to run experiments of high complexity. There is an active community of users and developers around it. Its main weakness is its high price (≈ 870 $). Finch[5] (≈ 99 $) is a very simple robot that has been designed around a wired connection to the computer. This connection reduces electronics requirements, such as batteries or wireless communication, and allows to implement the control entirely on the computer. This results in a very broad set of possible programming languages available, which is the real force of this robot. However, the cable does not allow real autonomy and mobility. Finally, the mBot[6] is a mobile platform based on an Arduino board. Its electronics is simple and inexpensive and the robot only features a couple of sensors, which allows to drastically reduce its price (≈ 75 $), but also limits the perception possibilities and therefore the span of use.

With respect to these robots, Thymio has a compact size (120 mm), many interaction possibilities, an affordable price and a large set of sensors. Its design makes the best use of the recent cheap components coming from the explosive growth of the mobile devices market. To the best of our knowledge, beside Thymio there are no educational products providing a similar integration of sensors and actuators at a price below 130 $.

Teachers are one of the main target user group of the Thymio project. For teachers, the motivation to use robotic tools in their classes depends on many factors [10]. Among them, the availability of material and training plays a key role. The BeeBot mentioned previously, because of its radical simplicity, is one of the tools that needs the least training. Moreover its use in learning activities is based on a simple grid on a printed map on the ground, allowing fast and easy integration in existing lessons. These are probably two important factors for its success in schools [11], [12] and the availability of a rich set of educational material[7]. However, the total lack of sensors reduces the BeeBot to an automaton, easy to use but limited to activities implementable by sequences of motions, without any closed-loop control that would allow teaching computational thinking or robotics concepts.

For more complex robots, both the development of educational material and the training of teachers require a huge work, demanding a good mix between robotics and educational skills. Moreover, educational material varies from school to school, for instance requirements are very dependent on local educational programs and languages. For example, in the particular case of Switzerland, every of the 26 states composing the country has its own school program and its own language among the four national ones. Educational material and the related training must therefore be adapted to each situation. Even for large companies like LEGO®, such a development is not sustainable. To solve this problem, a crowd-sourcing approach can be used. An active community of users can contribute to the development of the material in a distributed manner, adapting the material to the local situation. LEGO® itself is moving toward this direction by promoting communities of users [13] with a user-producer interaction close to those of open hardware projects. An open source community regrouping developers, manufacturers and end users is therefore a very interesting model to address the distributed development and sharing of educational material, and the diffusion of training sessions. This model is an extension to education of what is already practiced in projects such as Arduino [14]. In htis paper, we study a case of implementation of this model.

III. Design choices

We designed the Thymio robot along seven main axes: a low price to address a larger number of users, a feature set that suits both genders and multiple ages from young children to adults, a combination of sensors, actuators, and programming features that facilitates learning, a mechanical design that allows creative extensions by the users themselves, a set of ready-to-use behaviors to get a demonstration right out of the box, an accessible programming environment, and an open source community contributing to the design and the dissemination. The result is a miniature (10 cm in side) differential-wheeled robot suited for use on a desktop (Fig. 1, top). The robot is robust enough to be mishandled by children, as it can fall from a table without breaking itself. It features a translucent white hull and a wide range of sensors and actuators (Fig. 1, bottom). The robot embeds a battery, rechargable by USB, that provides an autonomy of about 2

[2] http://mindstorms.lego.com

[3] http://www.beebot.org.uk

[4] http://www.parallax.com

[5] http://www.finchrobot.com/

[6] http://www.makeblock.cc/mbot/

[7] https://www.learningplace.com.au/deliver/content.asp?pid=38840 and http://beebots.skola.edu.mt/

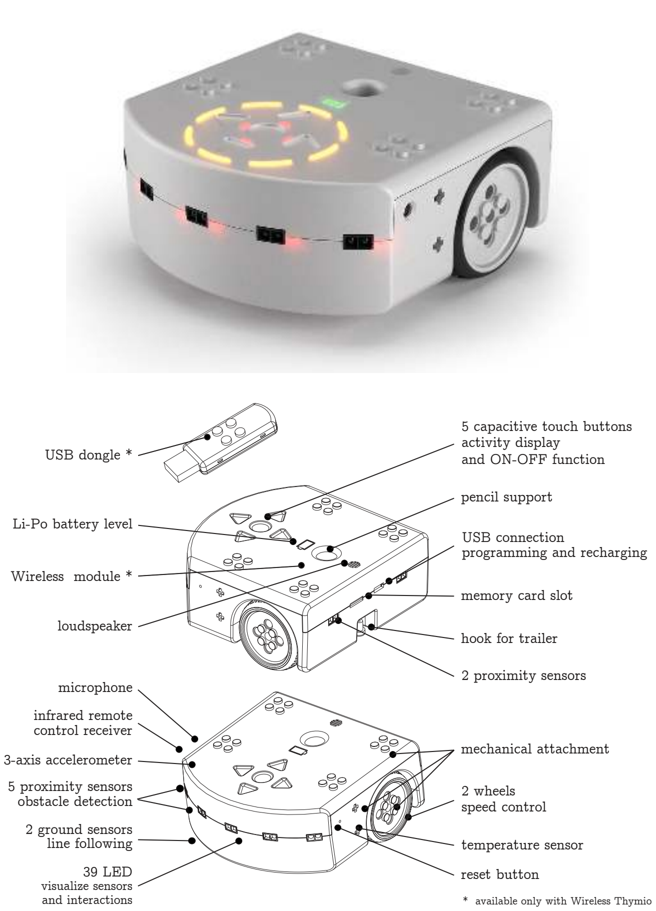


Fig. 1: The Thymio robot and its main components for the Wireless and the USB-connected versions.

hours. More details on the robot and the previous research results can be found in [15], [16].

A. *Low Price*

Our main concern during the mechatronic design of Thymio was to keep a low production cost. Price is a key factor for the adoption of the robot by schools [17]. In our previous experience with designing robots, most of the costs were due to the electronics and the sensors [18], not to the mechanical parts. As we wanted Thymio to be an affordable robot, we selected electronic components with great care and explored different design solutions in order to limit the number of expensive components. In particular, we focused on low-cost sensors that allow rich interaction possibilities, with both the environment and the user. The resulting Thymio robot possesses a large number of sensors: Seven horizontal infrared distance sensors, with a range of about 13 cm. They enable many interactions with the environment, like following an object or avoiding an obstacle. Two ground infrared sensors allow to follow a line or to detect the edge of a table. A three-axis accelerometer measures the orientation of the robot with respect to gravity, and can detect shocks and free falls. Five capacitive touch buttons organized as a direction pad form an intuitive user interface. Compared to physical buttons, these simplify the plastic hull of the robot and make it more robust. A remote control receiver provides additional distant buttons. A microphone can record sounds and detect claps. Finally, a thermistor measures temperature. Most of these sensors cost less then 0.2 $, the most expensive being the accelerometer with a cost of about 0.8 $, which is an acceptable price given the possibilities it brings to the robot. We also chose low-cost toy motors and control them in speed (max. 13 cm/s) using a custom-designed electronics. By measuring the back-electromotive force, speed control is possible without additional encoders.

Description	Price (USD)
electronic components	15 $
microcontroller	4 $
motors	2 $
plastic parts (hull, wheels, and light guides)	4 $
assembling	8 $
transport	2 $
Total	35 $

TABLE I: Production cost of USB-connected Thymio.

However, choosing the optimal combination of sensors and processor is a difficult problem, as hardware interfaces are limited on a microcontroller and therefore compromises must be made. We evaluated several microcontrollers and chose the PIC24F from Microchip because it integrates a USB interface and can drive capacitive touch buttons directly, saving additional components. This microcontroller controls all sensors and actuators, excepted the battery recharging logic, which uses a specific chip for safety reasons.

In addition, we conceived Thymio to fit the requirements of series production. This is a very critical choice in open source hardware projects, and we will discuss this issue in detail in the next section. We produced until now more than 10’000 robots by batches of 1’000 units. Because manual work is expensive in western countries, we produce through subcontractors in China. Indeed, while most of the electronics consists of SMD components mounted by a robot, some are soldered (sensors, microphone, LEDs) by hand and the robot is also assembled by hand. We streamlined the production of plastic parts (Thymio has only 5 parts while our previous prototype had 11) and opted for injection molding. The plastic hull consists of two main parts screwed together to facilitate the vertical assembly and disassembly of the robot. Moreover, when selecting features we took into consideration the complexity of the injection mold.

Thymio finally costs 35 $ per unit when produced in a batch of 1 k units (see Table I), with fixed fees of 15 k$ (13 k$ for the mold and 2 k$ for the CE certification). This allows a selling price of CHF 129 ($\approx$ 130 $).

B. *Multi-age and Gender-neutral Feature Set*

While we optimized Thymio for cost, we also targeted a large range of users thanks to an important contribution by industrial designers of the University of Art and Design of Lausanne[8]. The variety of sensors, the multiple ways of interacting with the robot, its hull design, and its customization make Thymio accessible to girls and boys of different age groups. The white hull is initially neutral and children can choose their own color using the RGB LEDs or attach accessories to the hull. The robot can be used from children as young as 6 years old to adults: The youngest play with the robot through a set of basic behaviors (see section III-E) and can use Thymio as a support for handicrafts or construction. For children above 9 years old and adults, Thymio is suited to introduce programming and robotics. Learners can start by programming new interactive behaviors (see section III-F) and can go on with constructing around the robot and programming a behavior that interplays with the construction. High school and university students can use the robot to learn advanced robotics concepts through the integration with ROS [19]. Teachers can use this flexibility of use to adapt the look and the behavior of Thymio to their teaching activities, or even introduce this adaptation as activity in handicraft or computer science lessons. Moreover users looking for additional hardware features can interface Thymio with other open hardware devices, such as computer boards[9] or other robotic mechanics[10]. Finally, because of the open-source and open-hardware nature of the project, advanced programmers can improve their skills in C by modifying the firmware or in electronics by disassembling the robot.

C. Facilitating Learning

When designing Thymio, we took care of providing many incentives for the users to learn new things throughout their direct interaction with the robot. This translates into specific hardware and software choices.

At the hardware level, we render visible the activity of the sensors by adding a LED next to each of them, for a total of 39 LEDs. These LEDs locally color the hull and allow the user to see immediately where and when the robot perceives a change in its environment. There is a LED next to each proximity sensor that lights up as soon as an object is close enough to be seen, and shines brighter the closer the object gets. Similarly, a combination of blue and red LEDs show the temperature, and the infrared remote control receiver and the microphone both have LEDs that flash when they detect something. On the top of the robot, a circle of 8 LEDs shows the 3D inclination of the robot thanks to the accelerometer. This circle is also used to reflect some of the behaviors of Thymio. Finally, two strong RGB LEDs color the whole top of the robot. In addition to a visual feedback, the capacitive buttons also have an acoustic feedback. The link between a sensor and its feedback can be turned off when programming the robot to use the LEDs and the loudspeaker for other purposes.

At the software level, we provide a set of programming environments (see section III-F) that enable beginners to discover programming progressively. First, we teach them the basic rules of programming using a purely visual interface, then they discover the construction of syntax trees by assembling graphical blocks, and finally we provide a full text-based coding environment with advanced debugging tools, such as real-time inspection of the variables of the robot and plotting features, providing a visual way to understand time-related concepts.

D. Promoting Creativity

We wanted the robot to be a starting point for users to invent their own creations. Therefore, we thought Thymio as a support for handicrafts and constructions. Its white color is meant to be seen as a blank page that can be decorated and drawn upon, and its shape allows easy integration into a structure. The robot features a hole in its middle to insert a marker for drawing. It also has a slot for a microSD card, allowing users to add their own sounds or to load a pre-compiled code by the simple insertion of a dedicated card.

The square format of the hull allows to use the robot as a base for driving the user's own constructions. To that end, Thymio is compatible with LEGO® bricks: There are four attachment positions on the top of the robot and two LEGO® Technics crosses on each side. Additionally, the two wheels have attachment points, which permits to use them to actuate elements elsewhere in the structure (Fig. 2, third row left) or to lift the robot's own weight (Fig. 2, third row right). Therefore we chose more powerful motors than strictly necessary to move the robot around. The same fixation points used for LEGO® bricks can be used to attach paper structures. Paper can allow to simply change the body shape and add some body movements, as illustrated in Fig. 2 by the orca, opening and closing its mouth while moving forward or by the bat, moving its wings. But paper and cardboard can also radically change the locomotion principle, as illustrated in the second row of Fig. 2 by the zombie, where the wheels of the robot activate the legs. The paper structure can be also used to interact with the sensors, as illustrated in the second row of Fig. 2 by the bear, who extends his paw in front of the sensors to drive its iceberg (the robot). The same fixation points can be used to attach 3D printed customized parts, as illustrated by the winder shown in the fourth row of Fig. 2. Moreover, one can use paper to create environments, either flat with patterns that can be used in association with the ground sensors (Fig. 2, bottom) or 3D object, such as the trees beside the zombie in Fig. 2. Finally, it is also possible to link several Thymio by software, allowing the coordination of complex multi-Thymio robotic structures.

E. Fast access to robotics behaviors

Many existing robots need to be built or configured before having the first operational behavior. This is a barrier for many users and we wanted to avoid it carefully, having a robot able to show interesting behaviors right out of the box. Therefore Thymio provides 6 different basic behaviors

[8] http://www.ecal.ch

[9] Interface with a Raspberry PI under https://www.thymio.org/en:thymioexplorer

[10] Interaction with the Poppy open source hardware under https://youtu.be/0otXtF8J_Z4

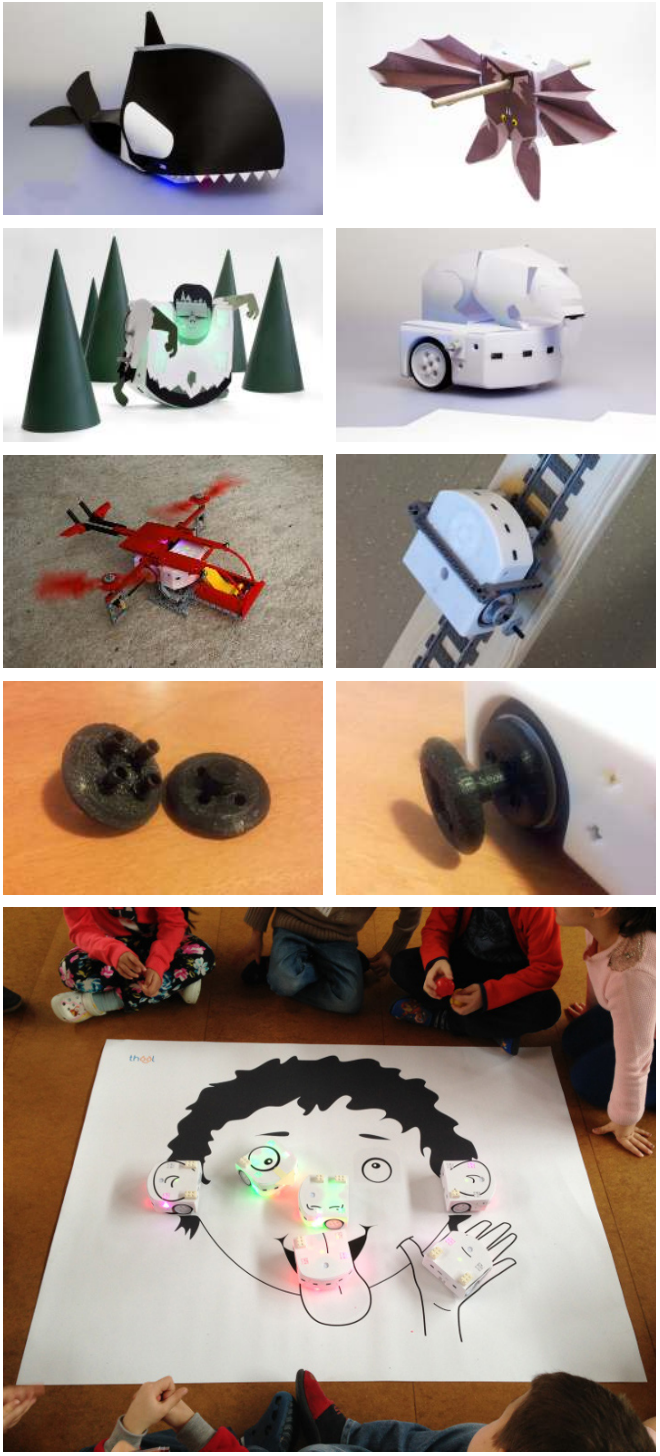


Fig. 2: Examples of extensions of the Thymio basic robot with paper or cardboard body extensions (top four images), using LEGO® structural extensions (third row), using 3D-printed extensions (fourth row) or using a printed environment (bottom).

mode	color	sensors	behavior
friendly	green	infrared (IR)	follows an object at distance
explorer	yellow	IR	moves avoiding obstacles
fearful	red	acc., IR	flees, notifies shocks and falls
investigator	cyan	IR	follows a black track
obedient	magenta	buttons, IR	follows moving orders
attentive	blue	mic.	moves following sound

TABLE II: The different basic behaviors with sensors used.

(see table II), which are accessible as soon as the robot is started and are stored in flash permanently, also if the Thymio is reprogrammed by the user. These basic behaviors allow people starting Thymio to immediately interact with it, while illustrating the many possibilities of the robot. The behaviors are interactive. For each behavior, the robot's body has a different color, allowing the user to easily recognize it. The user can navigate between the colors with the buttons and select the behavior she/he wants to use. These behaviors can be exploited in a construction to create reactivity without the use of programming, such as in the paper creations shown in the top four images of Fig. 2.

F. *Programming*

Thymio runs the Aseba open-source programming environment [20]. Aseba is designed to enable novices to program robots easily. On the robot side, it provides a lightweight virtual machine that runs on microcontrollers such as the PIC24F inside Thymio. A virtual machine allows instantaneous upload and safe execution of programs. On the desktop side, Aseba provides an integrated development environment (IDE) featuring a visual programming language (VPL) (Fig. 3), a scripting language (Fig. 5), and a mixed language, Blockly[11], to assemble scripts graphically (Fig. 4). These different languages cover the abilities of children of different ages and the progression of skills of learners.

The IDE integrates real-time feedback of the status of program execution, as this feature was recognized of critical importance to properly learn programming [21]. This capability is provided both with the VPL [22] and in the scripting environment, by displaying the content of variables in real time through texts or plots. The IDE also provides a help documentation[12]. In addition, the wiki pages of the project provide tutorials on programming with Thymio and a detailed description of the programming interface.

Aseba integrates with ROS [19] through the *asebaros*[13] bridge. Ros is one of the most widely used software framework in robotics research, and this integration allows to run sophisticated algorithms, such as simultaneous localization and mapping, in conjunction with Thymio. This makes the robot suitable for university-level education.

IV. Open source hardware, choices and impact

The choice of developing a robot within an open source hardware project has an impact on the robot design and the

[11] https://developers.google.com/blockly/

[12] currently in English, French, German, Italian and Spanish

[13] http://www.ros.org/wiki/asebaros

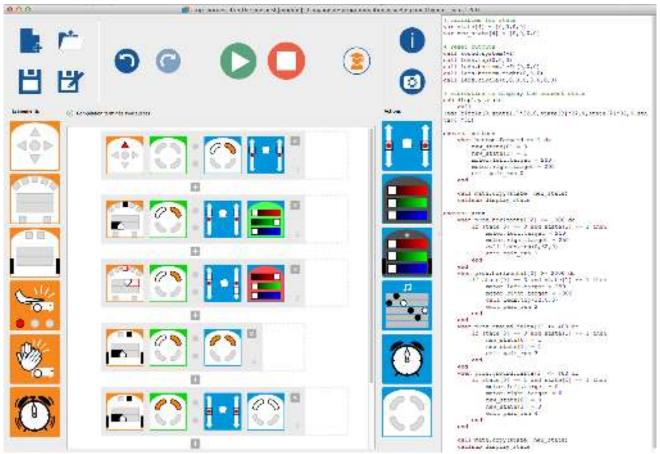
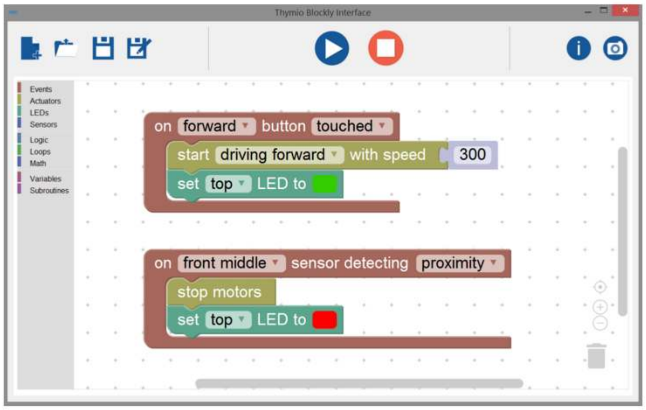


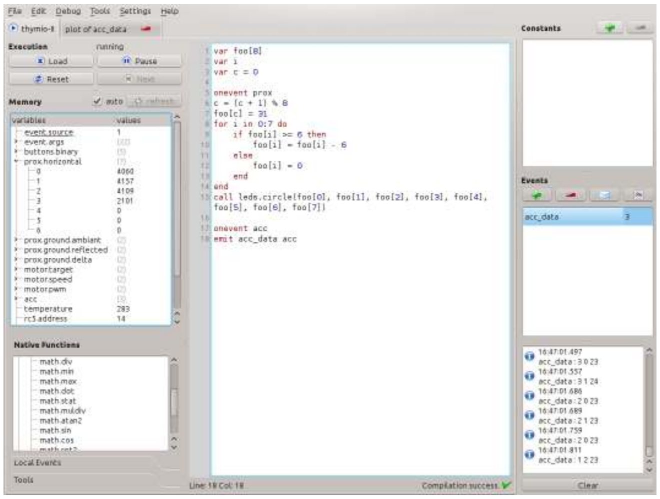
Fig. 3: The visual programming language.



```
on forward button touched
    start driving forward with speed 300
    set top LED to

on front middle sensor detecting proximity
    stop motors
    set top LED to
```

Fig. 4: The Blockly programming environment.



variables	values
event.source	1
event.args	(32)
buttons.binary	(5)
prox.horizontal	(7)
0	4060
1	4157
2	4109
3	2101
4	0
5	0
6	0
prox.ground.ambiant	(2)
prox.ground.reflected	(2)
prox.ground.delta	(2)
motor.target	(2)
motor.speed	(2)
motor.pwm	(2)
acc	(3)
temperature	283
rc5.address	14

```
var foo[8]
var i
var c = 0

onevent prox
c = (c + 1) % 8
foo[c] = 31
for i in 0:7 do
    if foo[i] >= 6 then
        foo[i] = foo[i] - 6
    else
        foo[i] = 0
    end
end
call leds.circle(foo[0], foo[1], foo[2], foo[3], foo[4],
foo[5], foo[6], foo[7])

onevent acc
emit acc_data acc
```

acc_data	3

```
16:47:01.497
acc_data : 3 0 23
16:47:01.557
acc_data : 3 1 24
16:47:01.686
acc_data : 2 0 23
16:47:01.689
acc_data : 2 1 23
16:47:01.759
acc_data : 2 0 23
16:47:01.811
acc_data : 1 2 23
```

Fig. 5: Aseba Studio, the integrated development environment

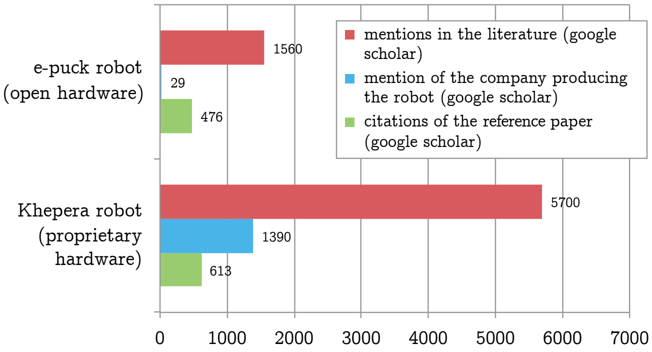


Fig. 6: Comparison of the impact of e-puck and Khepera.

way this device is used in the community of users. In this section we analyze more in detail the implications of this choice in the context of educational robotics. We compare these implications with the result of a survey that got 35 answers from people active in various open source hardware projects based on a worldwide call for contributions. Among these 35 answers, 11 come from project leaders, 13 from core design team members, 8 from contributors and 3 from enthusiastic users. 54% of these respondents are between 25 and 35 years old, 26% between 35 and 50 years old, 9% are less than 25 years old and 11% are older than 50. 67% either work in an academic environment or are students. 34 of the 35 answers come from male respondents.

A. *Motivation*

We can distinguish two levels of motivation, the institutional and the personal one. The institution initiating a project is mainly looking for recognition and/or money. Our group has a good experience in disseminating robotic hardware with the Khepera [23] and e-puck [9] robots. Khepera was disseminated with a proprietary strategy, e-puck with an open source hardware one. Both were targeting similar users and have been sold in similar quantities. What we can observe after more than 10 years is that Khepera generated royalties for the university, but less relative academic visibility than e-puck (Fig. 6). The e-puck robot, being open hardware and with an image better linked to the university, generated no income for the university but much more visibility. Khepera was more linked with the name of the company producing it, the e-puck robot was nearly not linked with the producer when mentioned in the literature. In the case of Thymio, developed in a research project, the institutional motivation was to get visibility more than money. Therefore, based on the past experience, the open source hardware strategy seemed more adapted.

The personal motivation to participate to such a project is very different than the institutional one. When asked about their personal motivation, the people participating to the survey give as main motivation a link to their professional activity and to the specific project, followed by a more general motivation by the nature of the project and finally by the engineering practice (Fig. 7). The more abstract goal of improving our society is mentioned only by few respondents.

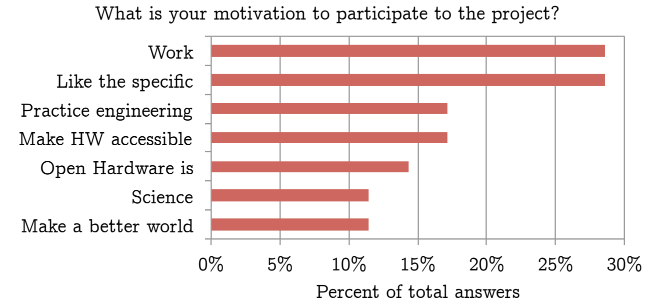


Fig. 7: Motivation of contributing to open source hardware.

In the Thymio project, most engineering contributors were hired for a research project developing the robot. Industrial designers also contributed as part of an institutional project. Therefore the link with the professional activity and the particular project is evident. On another side, the motivation of contributing to society is strong in our team, as developing a robot targeting education has an important societal component. Moreover, our project also has a strong scientific motivation, with several ongoing studies on the acceptance by teachers and the effect on children's learning. Therefore, sharing a strong fundamental motivation such as education or scientific achievements, is a key element for building a solid community [24], especially if it is interdisciplinary like ours.

It is also interesting to look at what people expect as benefits in participating in such a project (Fig. 8). We can observe both a technological component in getting better results and impact, and a human-relations component that is not in the basic definition of open hardware but results from the community created around the project. People expect to work with other people to achieve more together, to create a network, and to learn through new experience and contacts.

In the Thymio project we had similar expectations. Working together with several partners was for everybody a win-win situation, and creating a community of users was the only solution to allow the development of high quality accessories and educational material. We established a wiki[14] as the meeting point for the learners, the robot developers, and the teachers. This wiki introduces the robot, explains the basic behaviors, gives access to the programming environment and its documentation, and provides code samples and examples of constructions. It is open for editing by anyone, and although we initially provided most of the material, the other members of the community have started to contribute. One of the most important contributions to the wiki, in term of effort, has been its translation into four languages.

In addition to those elements, in our project we expected another important benefit from open hardware in term of image: we wanted a match between the non-profit nature of the project and the non-profit nature of education in general.

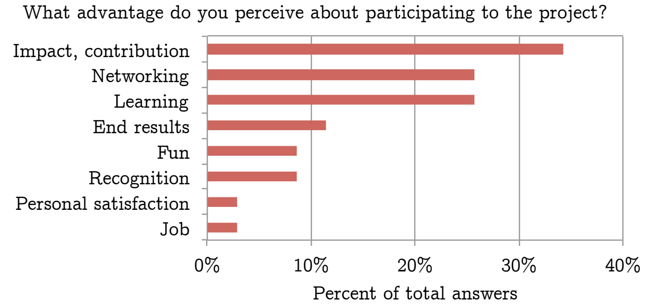


Fig. 8: Advantages of contributing to open source hardware.

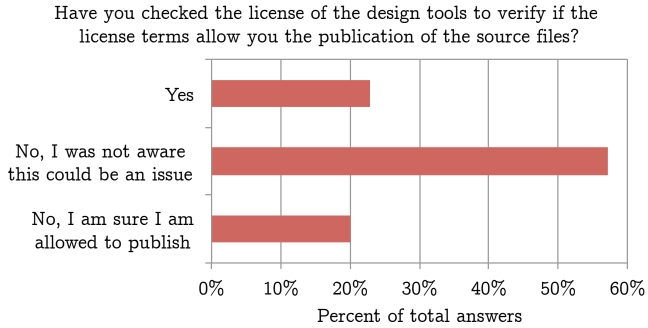


Fig. 9: Awareness about CAD tool licenses limitations.

B. License of project and license of tools

When starting an open source hardware project, one of the typical questions is which license to use when disseminating the project source files. We will not discuss this matter here, as it is a very common and well-covered issue.

There is another licensing issue which is not well known and that we discovered very late in our project: the constraints of the license of the mechanics and electronics CAD tools. Indeed, when asked about this issue, the participants to our survey seem generally not aware of the fact that CAD licenses can be very restrictive about the way the source files can be published (Fig. 9). More than the half or respondents to our survey state that they were not aware that this could be an issue, and only one third checked the license of their CAD software. Some stated that they did not checked the license because they were sure they can freely publish their design, but one third of them are using software not allowing the publication of source files when using educational licenses, and all these respondents are academics or students. This issue is very serious, as most contributors of open source hardware projects are academics and use academic or educational licenses. Most licenses do not allow to use the design created with this version of the software for commercial activities. As producing or selling the product is part of the definition of open hardware, these licenses simply forbid publication under the standard open source hardware conditions.

To clarify this issue we contacted twelve of the major editors of mechanical CAD and PCB routing software. We asked them if their educational license allows the publication of the source files, and we specified that the published files could have been downloaded by a company to produce

[14] http://www.thymio.org

⚙ = mechanics ⚡ = PCB design	Software	Editor	Free publication of source files from educational license	Explicit mention of possible publication after editor approval	Price increase between the educational license and one allowing publication
⚙	AUTOCAD or INVENTOR	AUTODESK	YES		0%
⚡	Diptrace	Diptrace	YES		0%
⚡	Eagle	CADsoftUSA	YES*		0%
⚙	CREO	PTC	NO	YES**	0%
⚡	Allegro PCB design	Europractice / Cadence	NO	YES	0%
⚡	Target3001	Ing.-Büro FRIEDRICH	NO	YES	0%
⚡	Altium Designer	Altium	NO	NO***	193%
⚙	VariCAD	VariCAD	NO	NO	658%
⚙	Cobalt	Ashlar	NO	NO	875%
⚙	Vectorworks Fundamentals	Vectorworks	Not answered after two months		
⚙	CATIA or SolidWorks	Dassault Systems	Final answer not arrived in two months		
⚙	NX or SolidEdge	Siemens	Too busy staff to answer in two months		

Reaction to this survey

* if explict open source license
** approval automatically granted for open source and European research projects
*** academic package can include a commercial license allowing publication

Fig. 10: Publication possibilities as function of the CAD editors, situation at the end of March 2016.

the system for commercial purposes. Fig. 10 summarizes the result of this survey. Only three of the twelve editors have education licenses allowing this type of publication. Two others explicitly mentioned the possibility to ask for permission before publication. A large mechanical CAD editor was puzzled by our questions and after realizing the impact of the license, introduced a special condition allowing publication of files in clearly-labeled open source hardware projects. In previous situations of open source publication, the same editor asked the universities to purchase commercial licenses to permit publication. This can multiply by factors of hundreds the price of the CAD license. This blocking factor for open hardware also applies to the publication of scientific results, as promoted by many governments in the last decade and generally called "open science". The change of policy in CAD files publication that we obtained is a sign of the very positive trend set by open hardware and by open science in general.

As a conclusion, this legal issue is totally underestimated by both people participating to the projects and by the CAD editors. In a period where editors are looking for additional revenues and are attacking universities for misuse of licenses[15], this can be a very dangerous situation.

C. Who designs and produces the hardware

Among the fundamental choices when starting an open source hardware project, there is the choice of the type of production. In the definition of open hardware we gave, it is stated that one should offer an "hardware whose design is made publicly available so that anyone can [. . .] make [. . .] hardware ". Behind the "anyone" should we consider every single person or only companies able to produce the product? This choice has implications on who designs the system and how. In our project we have two different types of hardware: the robot itself and the accessories used in specific activities. The robot is the expensive part and has a very neutral design, allowing adaptation to specific situations. This adaptation is achieved by custom accessories that increase the attraction of the robot in its specific role, enabling activities for different ages and genders.

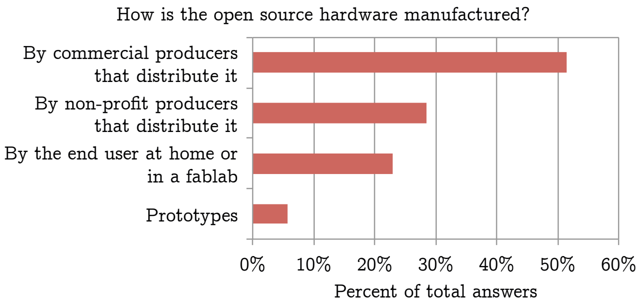


Fig. 11: Production methods in the mentioned projects.

For the robot itself, we opted for the interpretation of the definition that considers under "anyone" only the professional structures able to mass-produce hardware, for a reason of price and complexity of the product. When looking for the best performance per price ratio, we decided to use techniques that require heavy equipment. For instance we decided to produce all mechanical parts by injection molds. This ensures a price per part of some cents, which cannot be achieved by others techniques. Letting end-user produce their own parts would results in higher prices for less performances. This choice has another implication: only a core team of highly skilled engineers can contribute to the project. It is not trivial to design a mechanical part that can be molded in the simplest possible way.

For the accessories, less technically challenging and stronger linked with creativity and educational value, we promoted techniques that are accessible to "anyone" in the broader sense: paper, cardboard, LEGO® constructions and 3D printing. This allows a much broader spectrum of contributors, including teachers and lay people.

Our survey shows that the most common production method is a centralized one by commercial companies (Fig. 11). Indeed, large open source hardware projects like Arduino are based on this model. One could argue that this model makes these robotic projects close to standard commercial models, leading to a lack of involvement from users and loosing the open source spirit. In our case, we decided to alleviate this effect by creating a non-profit association, called Mobsya, in charge of producing and selling the robots. Moreover, the users are better integrated in the second layer of open source hardware consisting of accessories, where anybody can contribute their own designs and put them on the wiki.

Another strong difference between a standard consumer product and our vision of open hardware is that Thymio should be durable. Schools do not have large budgets for technological tools and make strong investment in training of their teachers when adopting a new tool. Therefore, the lifetime of the products should be as long as possible. The

[15] We are aware, just in Switzerland, of two situations where large amounts of money are asked to universities for non-respect of software licenses.

open hardware approach gives to the user, or to a generic technician, better conditions to repair the system. Supporting this type of operation has an impact on the robot design; for instance Thymio can be easily opened with four standard screws, and we introduced connectors between key elements such as motors, speaker, the battery and the main PCB.

Another key element in supporting repairs by end users is the documentation of calibration methods. When choosing very low cost components, one faces large dispersion of characteristics. For example, in the Thymio the right and left wheel motors can differ in their electrical characteristics, resulting in the robot not going straight for similar speed commands to both wheels. To correct this problem, we introduced factory calibration, storing in the microcontroller memory correction factors that are specific for each robot. To allow the user to replace a broken motor, it is essential to give her also the possibility to re-calibrate the robot and adjust the parameters of the new motor. In Thymio, this results in the design and the documentation of calibration processes that can be performed by anyone, getting close to the original definition of open source hardware.

D. Added value of open source hardware in distribution

Beside the sale of more than 10'000 robots by Mobsya, there are several elements that show a specific advantage of having chosen an open source hardware model.

The first element is linked to the financial support of the project. Although the non-profit nature of the project blocked the possibility of having shareholders injecting capital, this same nature enabled a lot of institutional donations linked with the societal goal of the project. The financial support through donations instead of acquisition of shares, allows keeping a total decisional freedom and void debts at the same time. The main disadvantage of this approach is the amount of financial support, limited in our case to several hundreds of thousands of dollars.

The second interesting element of the open source hardware approach is the very high acceptance of the resulting robot by universities and their offices in charge of promotion of science. Although these are not the primary users targeted by the robot, these are institutions with high visibility and influence on school programs. In particular INRIA in France and the University of Cambridge in the United Kingdoms decided to use Thymio as tool for promoting their activities and introduced it into schools. In France, INRIA developed several teaching modules and supported the training of more than 1000 teachers to the use of Thymio. This resulted in Thymio being one of the tools chosen for the education of digital science in the reference books of the French Academy of Science and its foundation for promotion of science in education, called “La main à la pâte.” The open source nature of Thymio gives these institutions a large control, as they can ensure maintenance and even further develop the robot. The impulsion given by the academic institutions has been followed by many teachers who contributed their own teaching material. Currently one can find more than 50 teaching modules, a large tutorial, and many reports on school activities using Thymio.

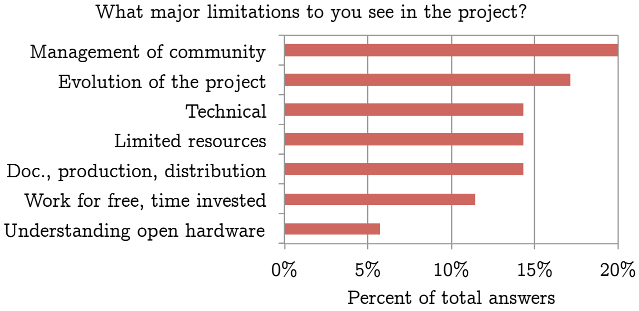


Fig. 12: Limitations of open source hardware projects.

In some cases, the open source nature of this project was a crucial political argument for the choice of this platform. For instance, in Geneva, Switzerland, the state government decided to move toward open source non-proprietary tools in education. All primary school and most secondary school computers have been installed with the Ubuntu distribution of Linux. On this operating system the LEGO® Mindstorms® software tools are not available, and their proprietary nature is not compatible with the philosophy dictated by the government. Thymio was therefore extremely welcome, and the state of Geneva organized their own training courses for teachers, adapted existing educational materials to their own curriculum and organized a lending system for all schools that cannot afford to buy the robots.

E. Problems of open-source hardware projects

The very specific structure of an open-source hardware project has also drawbacks. Asked about problems met in their project (Fig. 12), the participants to our survey mention the management of the community, the evolution of the project, the efforts in documentation, production and distribution and the limited resources. These problems are mainly caused by the mismatch between the conditions of the initial phase of the project, and the needs of long term sustainability. Initially, these projects are created by academic people with engineering background, typically in a flat structure. However, long term sustainability requirements are closer to the ones of a successful startup, and involve communication, marketing, and professional management in order to acquire sustained funding. Yet, 62% of the participants to our survey report projects that are managed by the founder of the initiative, who might not be the best person to lead such a structure. This differs from open-source software, because in our model initiating hardware production requires the investment of a large amount of money, and modifying hardware is much slower than modifying software. Moreover, in the case of robotics, although software plays a critical part, the perception of the product is highly centered on the hardware, creating tensions between hardware- and software-oriented people. Therefore, to grow such open hardware projects, there is a structural conflict between open-source ideals (for example meritocratic decision process) and the concrete financial and logistic constraints.

In the Thymio project we have addressed these problems by having several partners, each of them specialized in one aspect: design, production and sale, creation of educational material, etc. Despite this approach, we also encountered difficulties in nurturing an active community covering all aspects of the project, and issues in finding a long term vision that both injects life into the project and is financially sustainable. This is especially difficult with a large number of partners, as differing views can result in conflicts. A key element that allowed to keep cohesion in our community was to share, among all partners, strong basic values about improvement of society through education.

V. Conclusion

The introduction of robots in formal education is a very challenging task, not only because of technical requirements such as low cost and interactivity, but also because of factors depending on the school environment, such as the diversity of the educational programs, the dependence on local structures and languages, or the required training of teachers. Most of the current robotic tools for education capitalize upon existing hardware platforms, programming environments, or robot architectures. Because they do not consider all elements at stake, these tools fail to meet some challenges of formal education. On the contrary, in this paper we presented a global approach that tackles these questions in a holistic way, based on open source and non-profit principles. The result is the Thymio robot and its surrounding open source community including engineers, art designers, production and sales people, and teachers. The open source hardware and the related contributions are split in two categories: the robot itself, produced by a non-profit organization and requiring very advanced skills in both design and production, and the accessories for activities, based on techniques broadly accessible and with a larger panel of contributors.

The open source hardware development and dissemination of Thymio addresses several of the issues found in educational robotics. It allowed to broadly distribute the robot with minimal changes dues to management of intellectual properties, royalties, financial support and so on. This was achieved with an excellent match between the philosophy of the project and the one of the community of users in education. In particular, the open source approach allows to provide a durable robot, easy to maintain and repair, with at the same time a community of users providing educational material and mutual support.

By making a survey among contributors to open source hardware projects, we could observe that our project shares some characteristics with the majority of the projects represented in the survey. We also faced difficulties in establishing the community, we chose production and distribution methods that are broadly applied, and we share most of the motivational elements that are behind other projects. We identified an underestimated legal issue for open source hardware projects in the licensing term of CAD software. Indeed only very few editors have licenses allowing the publication of source files created with educational license. Two surveys among the users and the CAD editors show that both people contributing or leading open source hardware projects and the editors themselves are not aware of this issue. Finally, we could show some elements, specific to educational robotics, that differentiate our project from other open source hardware projects. In particular, our project takes advantage of an alignment between the principles underlying open source and the nature of education institutions. This fits a political trend toward open source in education and in science in general. We also found a solution to the problem of production methods by splitting our hardware in two categories, enabling both advanced technology for the robot and a large variety of accessories. Hence, Thymio appeals to a broad community of end users, which addresses durability of equipment and gender issues that are critical in education.

References

[1] CERI, "New millennium learners," in *OECD/CERI International Conference "Learning in the 21st Century: Research, Innovation and Policy"*, OECD, 2008.

[2] L. Leonard, "LEGO® MINDSTORMS® for Schools in Early Years Education," in *Kreativität und Innovationskompetenz im digitalen Netz: Wie kommt das „Neue" mit Hilfe von Internettechnologien in die Welt?* (V. H.-P. und Michaela Luckmann, ed.), ch. 8, pp. 316–324, Salzburg Research, 2009.

[3] I. Nourbakhsh, "Robot diaries: Creative technology fluency for middle school girls [education]," *Robotics & Automation Magazine, IEEE*, vol. 16, no. 1, pp. 16–18, 2009.

[4] M. Rubenstein, B. Cimino, and R. Nagpal, "AERobot : An Affordable One-Robot-Per-Student System for Early Robotics Education," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6107–6113, 2015.

[5] E. B. B. Gyebi, M. Hanheide, and G. Cielniak, "Affordable Mobile Robotic Platforms for Teaching Computer Science at African Universities," in *Proceedings of the 6th International Conference on Robotics in Education*, (Yverdon-les-Bains), 2015.

[6] D. Hood, S. Lemaignan, and P. Dillenbourg, "When children teach a robot to write: an autonomous teachable humanoid which uses simulated handwriting," in *International Conference for Human-Robot Interaction HRI '15*, pp. 83–90, 2015.

[7] E. Mazzoni and M. Benvenuti, "A Robot-Partner for Preschool Children Learning English Using Socio-Cognitive Conflict," *Educational Technology & Society*, vol. 18, no. 4, pp. 474–485, 2016.

[8] F. B. V. Benitti, "Exploring the educational potential of robotics in schools: A systematic review," *Computers & Education*, vol. 58, no. 3, pp. 978–988, 2012.

[9] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli, "The e-puck, a robot designed for education in engineering," in *Proceedings of the 9th conference on autonomous robot systems and competitions*, vol. 1, pp. 59–65, IPCB: Instituto Politécnico de Castelo Branco, 2009.

[10] M. Chevalier, F. Riedo, and F. Mondada, "How do teachers perceive educational robots in formal education? a study based on the thymio robot," *Robotics & Automation Magazine, IEEE*, 2016.

[11] K. Highfield, J. Mulligan, and J. Hedberg, "Early mathematics learning through exploration with programable toys," in *Proceedings of the Joint Conference Psychology of Mathematics Education (PME 32-PMENAXXX)*, pp. 17–21, 2008.

[12] M. Demichele, G. Demo, and S. Siega, "A piedmont schoolnet for a k-12 mini-robots programming project: Experiences in primary schools," in *Workshop Proceedings of Intl. Conf. on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2010)*, 2008.

[13] C. Hienerth, C. Lettl, and P. Keinz, "Synergies among producer firms, lead users, and user communities: The case of the LEGO producer-user ecosystem," *Journal of Product Innovation Management*, vol. 31, no. 4, pp. 848–866, 2014.

[14] P. Jamieson, “Arduino for teaching embedded systems. are computer scientists and engineering educators missing the boat?,” in *Proceeding of the 2010 International Conference on Frontiers in Education: Computer Science and Computer Engineering*, pp. 289–294, 2010.

[15] F. Riedo, *Thymio: a holistic approach to designing accessible educational robots*. PhD thesis, STI, Lausanne, 2015.

[16] S. Magnenat, J. Shin, F. Riedo, R. Siegwart, and M. Ben-Ari, “Teaching a core cs concept through robotics,” in *Proceedings of the 2014 Conference on Innovation &#38; Technology in Computer Science Education*, ITiCSE ’14, (New York, USA), pp. 315–320, ACM, 2014.

[17] S. Kradolfer, S. Dubois, F. Riedo, F. Mondada, and F. Fassa, “A sociological contribution to understanding the use of robots in schools: the thymio robot,” in *Social Robotics*, pp. 217–228, Springer, 2014.

[18] M. Bonani, *Robotique collective et auto-assemblage*. PhD thesis, École Polytechnique Fédérale de Lausanne, Lausanne, 2010.

[19] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng, “ROS: an open-source Robot Operating System,” in *ICRA Workshop on Open Source Software*, IEEE Press, 2009.

[20] S. Magnenat, P. Rétornaz, M. Bonani, V. Longchamp, and F. Mondada, “ASEBA: A Modular Architecture for Event-Based Control of Complex Robots,” *IEEE/ASME Transactions on Mechatronics*, no. 99, pp. 1–9, 2010.

[21] J. Sorva, “Notional machines and introductory programming education,” *ACM Transactions on Computing Education (TOCE)*, vol. 13, no. 2, p. 8, 2013.

[22] S. Magnenat, M. Ben-Ari, S. Klinger, and R. W. Sumner, “Enhancing Robot Programming with Visual Feedback and Augmented Reality,” in *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE ’15*, pp. 153–158, 2015.

[23] F. Mondada, E. Franzi, and P. Ienne, “Mobile robot miniaturization: A tool for investigation in control algorithms,” in *Proceedings of the Third International Symposium on Simulation on Experimental Robotics (ISER-93)* (T. Yoshikawa and F. Miyazaki, eds.), vol. 200 of *Lecture Notes in Control and Information Sciences*, pp. 501–513, Springer, 1993.

[24] A. Ståhlbröst and B. Bergvall-Kåreborn, “Exploring users motivation in innovation communities,” *International Journal of Entrepreneurship and Innovation Management*, vol. 14, no. 4, pp. 298–314, 2011.