CHARACTERIZATION OF PHYSICAL COMPUTING DEVICES BY ATTRIBUTES FROM A PEDAGOGICAL PERSPECTIVE

Eric Schätz and Alke Martens
University of Rostock
Albert-Einstein-Straße 22; D-18059 Rostock; Germany

ABSTRACT

Due the wideness of the term Physical Computing, there is a need for a better structure of this topic. This paper is about an approach of structuring this field by finding attributes of different physical computing devices which can be used in class. Those attributes are meant to enforce teachers as well as researchers to analyze different devices from a pedagogical perspective in order to use those devices in class and use their full educational potential.

KEYWORDS

Physical Computing, STEM

1. INTRODUCTION

Computers which are used in class, have become much more diverse over the past few years: While usual desktop machines were the only computers students worked with about 20 years ago, laptops, tablets and way smaller computers and micro controllers were nowadays brought to class. As students were mostly taught for example how to use office software in former times, modern curricula (e. G. (Ministerium für Bildung, Wissenschaft und Kultur Mecklenburg Vorpommern, 2019)) also include the functionality of different computing systems as well as physical computing. At this point we need to clarify what we mean by *physical computing* and *physical computing device (PhCD)*: Physical computing means developing computer systems, which interact with their environment via sensors and actuators (Przybylla, 2018). Thus, a physical computing device is a computer which allows a developer to use sensor data or steer actuators. This definition, however, is still too wide, as almost every input- and output device can be called sensor or actuator and be accessed by developers. Therefore, almost every computing system could be called PhCD which is not a helpful basis for further research in this topic. In this paper, we introduce a way to structure this field.

2. EVALUATION OF TEACHERS PROBLEMS

At our lab, we offer training classes for teachers to use different devices in their lectures. Our observation is that teachers first seem to be excited but most of them tell us a year later, that they rather stick to the good old desktop machines – even if it is meanwhile widely agreed, that physical computing devices are an advantage when it comes to teaching certain aspects of Computer Science. This circumstance led us to make a survey with teachers and to ask them which are their biggest reasons of **using** and **not using** Physical Computing Devices in their lectures.

To prepare the survey, we made some interviews with a handful of teachers and asked them about their reasons for working or not working with PhCD. After this, we put abstract versions of these reasons in the survey. In the survey we asked teachers to decide for every reason on a scale from 1 to 10 if it has a low (1) or high (10) impact for them. Only 19 teachers participated on the survey, however, their answers provided some interesting insights. We compared the average answer for every point and came to the result that the three biggest reasons of **using PhCD in class** of those n = 19 teachers were:

1. Increase of student motivation (7.47)
2. More diversity of methods in class (6.82)
3. Making content easier to understand for students (6.78)

The three biggest reasons of **not using PhCD** were:

1. Missing material which is easy to use in class (6.44)
2. Missing overview of available devices (6.00)
3. Missing availability of devices in school (5.89)

In the following we are going to focus on the reasons why they did not use PhCD in class.

Let's start with the first reason for not using PhCD. Point 1 is simply not true and thus is very likely a psychological motivated answer. We know the school situations of all our participants, and next to the fact that all the related schools offer the hardware (see analysis of point 3), there are for example many different projects available on Microsoft MakeCode or there are even books available for the Calliope mini. However, the fact teachers tell us this point shows, that they may just don't have enough information about the topic (which is point 2). The third point should also not be true since in the federal state the asked teachers work at it is written in law that those systems must exist in schools. Thus, teachers just need to order them if they don't have them yet. Therefore, the second point may be the reason for points 1 and 3 as a missing overview of different devices may lead teachers not to know which devices they should order or for which material they have to look for.

3. CREATING A STURCTURE BY FINDING ATTRIBUTES

As pointed out in the introduction, physical computing can be done with many different computing systems but in very different ways. As those devices will never stop to develop, those attributes need to be abstract enough in order to be able to classify in the future upcoming devices we cannot think of at the moment. In order to develop such a system, we have analyzed different devices which can be used in school. Those devices reach for example from the Apple iPad via the Calliope Mini (Calliope gGmbH, 2022), the BBC Micro:Bit(Microbit Foundation, 2020) up to the OzoBot (Ozo Edu Inc., 2022). Our analysis leads to the result, that PhCD can be described by the following 9 attributes (Schätz and Martens, 2022a, 2022b):

1. Appearance
2. Supported programming languages
3. Robustness against wrong handling
4. Openness of the ecosystem
5. Effort for administration
6. Costs
7. Supporting systems for teachers
8. Processing degree of sensors and actuators
9. Persistence

4. TEACHERS PERSPECTIVE – ATTRIBUTES

As teachers plan a lecture, they run though different steps, which can be call the pedagogical planning process (Meyer, 2015). In each step, there are different aspects of PhCD to think about. Those steps are:

1. **Content analysis**
   Scientific background of the content which shall be taught in the lecture.
2. **Organizational analysis**
   Analyzation of the circumstances of the lecture, which includes the behavior of the class as well as the technical infrastructure provided by the school.
3. **Educational analysis**
   Selection of which parts of the content is important for the students and how can it be simplified.

4. **Methodical analysis**
   Evaluation of different possibilities how the content can be taught, which material can be used, etc.

As point 1 must take place before the actual lecture can be planned, we want to focus on points 2 to 4 and connect those to the attributes in the following. We will classify the attributes as not important (-), semi-important (0) or very important (+) for the different steps. This way we are going to show which points are more and which ones are less important for in the certain steps of the lecture planning process.

As mentioned before, there are nine attributes of PhCD to think about in an educational setting. Those attributes are sometimes still quite complex and are going to be explained in more details in the following.

4.1 Appearance

Some devices appear as a single device, some are modular. Single devices are for example tablet computers or laptops. Those devices can be used in very different settings in class. Devices which are modular might be attached to certain tables, as desktop computers are. Other devices can be single devices for themselves but need another device in order to work. A BBC Micro:Bit which always needs a PC or Tablet in order to get programmed can be pointed out as an example for this.

This attribute is mostly important for the methodical analysis (+) and semi-important for the educational (0) and content analysis (0).

4.2 Supported Programming Languages

Different devices force very different ways of programming. As a desktop computer can be programmed in very different ways, the BeeBot (Bee-Bot, 2022) for example is programmed just via buttons on top of the device. At this point the representation-form of the language (for example textual or graphical) and the used paradigms (like object oriented, event controlled) need to be analyzed.

This attribute is not important for the organizational analysis but very important for the educational one and semi-important for the methodical analysis.

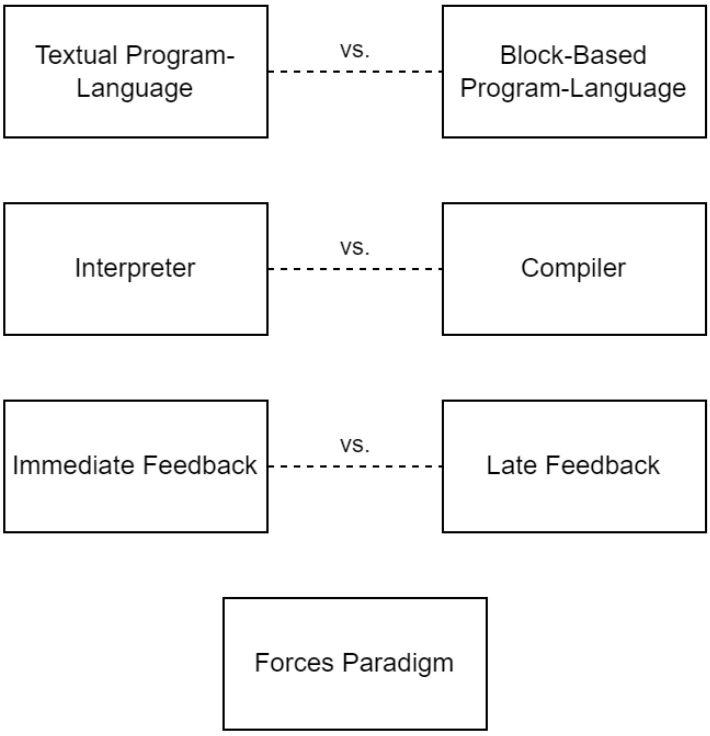

Figure 1. Aspects of programming language

4.3 Robustness against Wrong Handling

As students work with different devices, they will most likely not always use the device as it is supposed to. This can happen on accident or on purpose. There are three aspects to mention:

1. **Robustness of the software:**
   How stable does the development environment run? What happens if a student's program enters an infinite loop? How good is the bug detection for the students? Etc.
2. **Robustness of the case:**
   What happens to the device if the device gets dropped? Is it waterproof? Etc.? For example in a Sphero Bolt (Sphero, 2022) really robust and will not break, even if it gets dropped, as an Apple iPad would probably break.
3. **Robustness of the connections and pins:**
   As a LEGO Mindstorms (LEGO Group, 2022) for example uses ethernet-cable-like connections, which can mechanically not be used in a wrong way or an USB-C port is pretty robust as well, devices get much more fragile if they provide open pins, as the Raspberry Pi (Hüwe and Hüwe, 2019) or BBC Micro:Bit with an extension-board does. If students have access to the open pins, they could create a shortcut which can damage the whole device.

This attribute is very important for the organizational analysis (+), not important for the educational analysis (-) and semi-important for the methodical analysis (0).

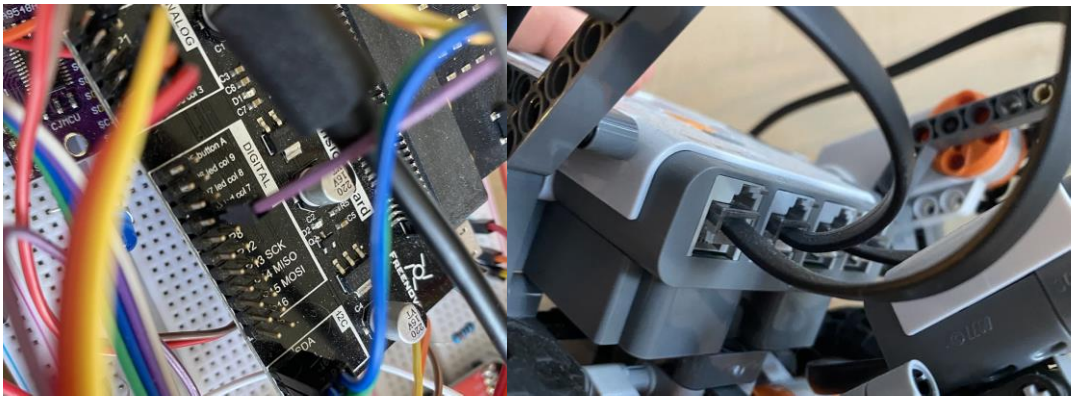

Figure 2. Connector-Pins from a BBC Micro:Bit Breakout-Board compared to a Lego Mindstorms

4.4 Openness of the Ecosystem

Different products usually belong into some kind of an ecosystem. At this point Apple-Devices can be pointed out but also other manufactures follow this strategy. This attribute can be divided into three sub-attributes:

1. **Software extensibility:**
   Can additional development environments be used? Is it possible to use external software modules or create own ones?
2. **Hardware extensibility:**
   Lego Mindstorms robots for example can the extended by adding other LEGO-parts to the robot. For the BBC Micro:Bit also exist many different extensions which are quite easy to use. The OzoBot for example cannot be extended.
3. **Manufacturer-binding:**
   Are there extensions from other manufacturers available?

This attribute is very important for the organizational analysis (+), semi-important for the educational analysis (0) and semi important for the methodical analysis (0).

4.5 Effort for Administration

As teachers are usually quite busy, they commonly do not want to spend too much time in administrating the devices. However, it is always important that these devices are always ready to work once a teacher wants to use them. This attribute can be divided into two sub-attributes:

1. **Software-administration:**
   How easy is it to install update on the devices? How much effort does it take to reset the device after a student worked with it? Can a student do some damage with the device?
2. **Hardware-administration:**
   Out of how many parts does the device consist - how easy is it to see if everything got returned by a student? Etc.

This attribute is very important for the organizational analysis (+), not important for the educational analysis (-) and semi-important for the methodical analysis (0).

4.6 Costs

Due limited resources at schools, costs are always important to look at. The costs can be categorized in initial outlay and continuous costs. As the initial outlay is easy to evaluate, the continuous costs should be planned for

1. spare parts
2. possible repair costs
3. necessary software subscriptions

This attribute is very important for the organizational analysis (+), not important for the educational analysis (-) and semi-important for the methodical analysis (0).

4.7 Supporting Systems for Teachers

Some devices already come with books or a pool of projects so that it gets easier to get to start with those products for teachers. Sometimes there is a good documentary by the manufacturer and sometimes there is a good community as it is for the Raspberry Pi, where developers help users and other developers.

This attribute is semi-important for the organizational analysis (0) and not important for the educational analysis (-). However, this can be really important for the methodical analysis (+): If there is a lot of good material for a teacher available, it is easier for him to prepare a group work, etc. because he can use existing resources and concentrate on improving those for his purposes.

4.8 Processing Degree of Sensors and Actuators

If a developer uses a sensor, there has usually already been done a lot of work by the board itself. There are several steps which must be done so that a developer can use it:

1. There is an **environmental appearance** which gets caught by a sensor by using a physical law. For example, does a brightness-sensor works on the physical law, that a diode changes its resistance once it gets illuminated.
2. The sensor expresses it by an **analog information**, for example a voltage
3. This gets converted into a **digital information**, which are discrete values on a scale for example from 0 to 1023.
4. This digital information needs to be converted into a **measurement** a human can work with, for example centimeters for length.
5. As a lost step values need to be **verified** and clarified from wrong values, which always appear once a while.

Some devices return values after step 3, other ones after step 5. This really affects the way; a student can solve a given task with a certain device.

ISBN: 978-989-8704-52-8 © 2023

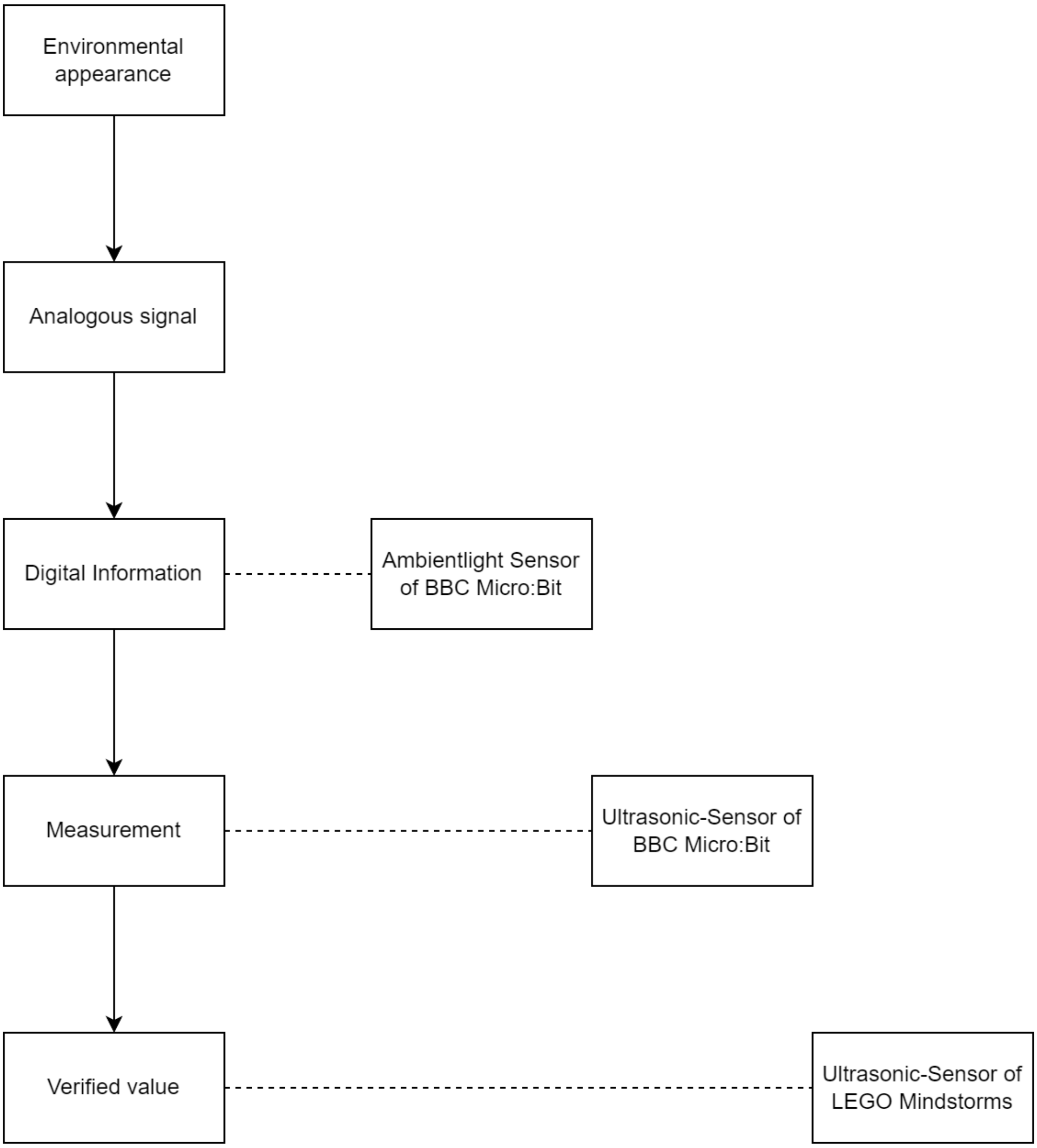

Figure 3. Different steps addressing the processing degree of sensors

Actuators can also be classified by three categories:

1. Actuator does likely what it's supposed to.
2. Actuator does likely what it's supposed to and returns if the action was successful.
3. Actuator does exactly what it's supposed to.

This way also really affects the sub-problems a student has to solve while working on a given task.

This attribute is really important for the educational (+) and methodical analysis (+) but not for the organizational one (-).

4.9 Persistence

Sustainability gets more and more important as well from the environmental as from the financial perspective. Products which are bought for school, typically from a public hand, are supposed to last as long as possible. At this point, devices need to be evaluated about

1. their expected lifetime
2. the permitted support time
3. the possibilities in repairing broken devices

This attribute is only important for the organizational analysis (+) but not for the other ones.

5. CONCLUSION AND OUTREACH

This paper explains an approach to structure the field of physical computing devices by defining different attributes form a pedagogical perspective. After some interviews, we developed a survey with abstract reasons and got 19 answers. These reasons supported our assumption, that a lot of work is required in the field of teacher education in relation to physical computing as devices for education. Our result showed that very likely, missing competence is the major point of not using PhCD at schools. One important step in developing these competences in teachers is to structure the field. First of all, teachers must see and learn, which kind of PhCD is available and what are features compared to other PhCDs. Thus, after intense discussions with teachers and students, we came up with nine central attributes of PhCD. We have connected those attributes to the lecture planning process to point out their individual relevance form a perspective of a teacher.

Based on those attributes, teachers are guided to evaluate different PhCD as well as different projects which are based on PhCD for their lectures. Researchers also can use those attributes as a basis for further research. We are going to focus on selected attributes in the future.

In our continuing research, we now use the attributes for structuring the field of PhCD and their purposeoriented usage in schools at different educational levels and with different computer science related content.

ACKNOWLEDGEMENT

At this point we would like to thank the teachers which answered our survey as well as all the students which supported the intense discussion about the attributes.

REFERENCES

Bee-Bot (2022). Bee-Bot - B-Bot.de [WWW Document]. URL https://www.b-bot.de/produkte/bee-bots/bee-bot/?gclid=Cj0KCQiA1NebBhDDARIsAANiDD1IhwgSOvARWaEGScP-tFS-Vnl3zbvjiKReln1BQhIwUJ_wqs5zY2kaAlQSEALw_wcB (accessed 11.17.22).

Calliope gGmbH (2022). Calliope [WWW Document]. Calliope. URL https://calliope.cc/ (accessed 7.18.22).

Hüwe, P., Hüwe, S. (2019). IoT at Home: Smart Gadgets mit Arduino, Raspberry Pi, ESP8266 und Calliope entwickeln, #makers do it. Hanser, München.

LEGO Group (2022). MINDSTORMS EV3 | MINT ab der neunten Klasse [WWW Document]. LEGO® Educ. URL https://education.lego.com/de-de/products/lego-mindstorms-education-ev3-set/5003400#ev3-set (accessed 5.21.22).

Meyer, H. (2015). Didaktische Modelle: Buch als E-Book. Verlag an der Ruhr, Berlin, GERMANY.

Microbit Foundation (2020). The new BBC micro:bit with speaker and microphone [WWW Document]. New Microbit Annouced. URL https://microbit.org/news/2020-10-13/meet-the-new-bbc-microbit/ (accessed 5.21.22).

Ministerium für Bildung, Wissenschaft und Kultur Mecklenburg Vorpommern (2019). Rahmenplan Informatik für die Qualifikationsphase der gymnasialen Oberstufe.

Ozo Edu Inc. (2022). Ozobot | Robots to code and create with [WWW Document]. Ozobot. URL https://ozobot.com/ (accessed 5.21.22).

Przybylla, M. (2018). From embedded systems to physical computing : challenges of the Digital World in secondary computer science education. Universität Potsdam, Potsdam.

Schätz, E., Martens, A. (2022a). PHYSICAL COMPUTING SYSTEMS – A SYSTEMATIC APPROACH. Presented at the CELDA 2022, Lisbon, p. 7.

Schätz, E., Martens, A. (2022b). Attributes of physical computing systems – the perspective of pedagogy, in: Edulearn22 Proceedings. Presented at the 14th International Conference on Education and New Learning Technologies, Palma, Spain, pp. 5561–5565. https://doi.org/10.21125/edulearn.2022.1309

Sphero (2022). Coding Robot: Sphero BOLT | Teach STEM for Kids [WWW Document]. URL https://sphero.com/products/sphero-bolt (accessed 11.17.22).