Snack-o-Mation: A Sweet Introduction to Industrial Automation and the Internet of Things

Marco Beck
University of Freiburg
Freiburg, Germany
marco.beck@student.uni-freiburg.de

Philipp Sommer
ABB Corporate Research
Baden, Switzerland
philipp.sommer@ch.abb.com

Philipp Scholl
ABB Corporate Research
Baden, Switzerland
philipp.scholl@ch.abb.com

Balz Maag
ABB Corporate Research
Baden, Switzerland
balz.maag@ch.abb.com

Sandro Schönborn
ABB Corporate Research
Baden, Switzerland
sandro.schoenborn@ch.abb.com

Figure 1: Snack-o-Mation in action during a TecDay event: students automate two Dobot Magician robots and a conveyor belt to deliver small sweets. This is a cooperative task where multiple teams are responsible for a single task in the automation system. Picture courtesy of Kantonsschule Zürcher Unterland, Bülach, Switzerland.

Abstract

We present Snack-o-Mation, an interactive teaching module of 90 minutes to educate students in Switzerland on Industrial Automation and the Internet of Things. Up to 18 participants are split up into multiple groups and asked to collaborate on several interdependent tasks. Two small robot arms and a conveyor belt demonstrate a vending machine for sweets. Participants interact with the system by sending radio messages from the micro:bit platform. We share our experiences and lessons learned from offering our module at 12 TecDay events at different secondary schools throughout Switzerland. Our module has been attended by over 450 students of age 12 to 19 during the last two years.

CCS Concepts

• **Applied computing** → **Interactive learning environments**.

Keywords

Robotics, Teaching, Internet of Things, Automation

1 Introduction

The rapid proliferation of the Internet of Things (IoT) is reshaping industries, economies, and everyday life. This makes it imperative to prepare the next generation for a future that is deeply intertwined with connected technologies. However, educational approaches often fall short in conveying the interdisciplinary and applied nature of IoT [1], particularly at school level. This paper explores innovative strategies for teaching IoT to students through interactive and hands-on methods that bridge theory and practice. By engaging students in hands-on activities with wirelessly connected devices we aim to foster a deep understanding and sustained interest in STEM (science, technology, engineering, and mathematics) fields.

Motivating young learners to pursue scientific and engineering disciplines is not only essential for addressing future workforce demands, but also for educating informed citizens that are capable of navigating a technologically complex world. This challenge is especially pronounced when it comes to female students, who are still underrepresented in technical domains. A study [2, 3] has shown that STEM events at secondary schools in Switzerland increase the likelihood that students will later chose a technical or scientific subject for their university studies.

2 TecDays

As part of its mission to foster scientific and technical literacy among youth, the Swiss Academy of Engineering Sciences (SATW) [4] organizes TecDays—immersive educational events hosted at secondary schools across Switzerland [5]. These events are structured as school-wide, full-day programs where students from age 12 to 19 engage in interactive modules led by experts from academia and industry. School students are able to indicate their interests for different modules from a catalog before the event. Based on the size of the school, students can select three modules out of up to 70 different modules offered by companies or academic institutions. Each module has a duration of 90 minutes and is held in a normal classroom or computer room at the school. The organizers of the TecDay will coordinate the logistics on site based on the requirements provided by the module facilitators (e.g., number of computers, space for the demonstrator, WiFi access).

3 Snack-o-Mation Module

In this section, we give an overview of the Snack-o-Mation module developed specifically for the TecDay events. The module has a duration of 90 minutes and is facilitated by 1-2 instructors in the local language of the school district (German, French, Italian). We provide the source code and teaching materials on GitHub[1].

3.1 Teaching Goals

The main goal of Snack-o-Mation is to provide a hands-on introduction into the world of robotics, automation, and the Internet of Things (IoT) [6]. Our demonstrator setup provides a vending machine-like automation system to deliver small chewy candy wrapped in paper (we use MAOAM Bloxx or similar). A potential customer can order a specific number of candies being delivered from two storage locations by means of two robotic arms and a conveyor belt, as shown in Figure 1.

Our module consists of the following parts:

(1) Introduction of the facilitators and company, examples for industrial automation systems (15 min)
(2) Demo of the final solution (5 min)
(3) Hands-on with the micro:bits and MakeCode (45 min)
(4) Collaborate to get sweets delivered (20 min)
(5) Wrap-up and goodbye (5 min)

3.2 Introduction & Examples

At the start of each session, the facilitators introduce themselves and their educational background, such as where they attended secondary school and which subject they have pursued during university studies. Furthermore, we give a brief introduction of ABB and ABB's research lab. We then introduce examples and the evolution of real-world industrial automation and their influence on society.

[1]https://snack-o-mation.github.io

3.3 Demonstration

Following the introduction part of the module, we then ask everyone to come close to the demonstrator setup, which is usually placed in the front of the classroom. We introduce the two robots and the conveyor belt and explain the working principle of the Snack-o-Mation system. Using the graphical user interface (GUI) shown on the screen or projector, we explain how to retrieve the coordinates of each robots' manipulator. Furthermore, we introduce step-by-step which operations are necessary to pick up sweets, place them on the conveyor belt and pick them up again for delivery. Next, we demonstrate by an example how a successful delivery of sweets to the customer should look like. Finally, we make the system forget about all the pre-configured coordinates by clicking a button on the GUI. We then explain that now the system needs to be re-programmed by the participants to make it work again.

3.4 Introduction to the micro:bit

The largest part of the module is dedicated to an exploration and introduction of the BBC micro:bit platform [7, 8]. We start by distributing the micro:bit devices and USB cables to the participants. Then we ask them to connect the devices to their laptops or PCs and open the Microsoft MakeCode[2] site in the browser. For the next 45 minutes, we let them explore how to interact with the micro:bit platform. The goal is to introduce all the concepts necessary to complete the collaborative exercise later on, such as user input and radio messages.

First, we explain how to build an application using the graphical editor offered by MakeCode (e.g., showing a heart-shaped symbol on the LED matrix at startup) and download it to the micro:bit device. Then, we let them explore freely for about 10 minutes with the different output modalities (symbols, showing text, making music) of the micro:bit. We then introduce the blocks from the *input* group in MakeCode, which tell the micro:bit what actions to execute based on different input events, such as button pressed, shaking the micro:bit, etc. Finally, we introduce the concept of sending and receiving radio messages between different micro:bits. We ask the participants to send messages upon a user input and trigger an action on reception of a radio message on another micro:bit.

3.5 Collaborative Exercise

After the participants got familiar with the micro:bit platform, we then proceed to complete the reprogramming of the Snack-o-Mation system in a collaborative manner. To do so, we split into up to 6 groups of 2-3 participants each. We hand out printed instructions which describe each group's dedicated task in detail.

Five groups are asked to transmit the coordinates (x,y,z) for the position of the corresponding robot arm in the given delivery step, i.e., storage locations, positions on conveyor belt, and final delivery location. To obtain the coordinates for a given task, the students are asked to move the robot arm by hand to the desired position while holding down a button on the robot's arm. The coordinates need to be sent in a radio message encoded as a string like `"1#x=100"`, where the first number identifies the group, followed by the name and value of the corresponding coordinate. The sixth group is asked to send the amount of sweets available in the two storage locations

[2]https://makecode.microbit.org

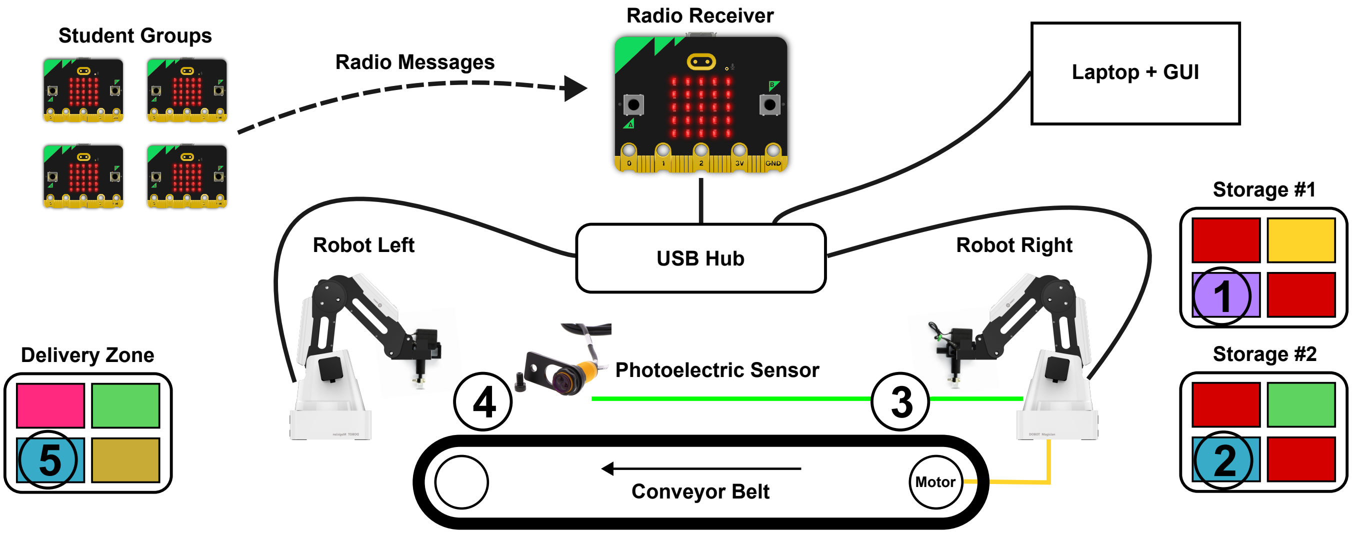

Figure 2: The Snack-o-Mation setup: Two Dobot Magician robots and a micro:bit are connected to a laptop via USB. The robot on the right side controls the motor of the conveyor belt and is connected to the photoelectrical sensor. The goal of the collaborative exercise is to send the coordinates through radio messages to pick up an item from one of the storage areas (position #1 and #2), place it with the right robot onto the conveyor belt (position #3), transport it on the conveyor until it is detected by the photoelectrical sensor (position #4), then pick it up again with the left robot and deliver it to the customer (position #5).

encoded in a string like "6#1,2", which indicates the group number and that the storage locations contain one and two items.

As the different groups work on solving their individual tasks, we use the GUI to keep track of the status of the different groups. Tasks that were completed change colors from grey to green in the GUI. This allows the facilitators to keep track of the progress of the different groups and provide assistance if needed. Furthermore, it motivates the different groups to complete the task as fast as possible. Upon sending all three coordinates for a specific position, participants can verify that the robot arm moves to the proper coordinates at the demonstration table. In addition, we have a few extra tasks at hand which can be assigned to groups that have already completed their original assignment. This helps to keep everyone engaged while we wait for completion of all the assignments.

3.6 Wrap-up

Upon completion of all tasks assigned to the groups, we ask everyone to come in front to our demonstrator. We then start the delivery of several sweets from the storage locations. This demonstration marks the end of our module and participants can take the successfully delivered sweets with them. In case there is some time left, we also engage the participants in a closing discussion and ask for their view on the topic of robotics. For example, if they perceive automation and robotics as a positive impact on society or if there are some aspects that they rather see from a critical point of view.

4 Hardware & Software Setup

In this section, we describe the components of our demonstrator setup, as shown in Figure 2. When selecting the hardware components for our module, we have taken care that our setup can be transported easily and can be installed within a short amount of time on a regular table in a classroom.

4.1 Robots and Conveyor

We use two Dobot Magician robots [9], a desktop-grade 4-axis robot arm built for classroom applications. Both robots are placed in proximity of the Dobot mini conveyor belt kit [10]. The conveyor belt is supplied with power by one of the robots, which can also control the speed of the motor driving the conveyor belt. Using a photoelectric switch attached to the conveyor, we can detect when an object approaches one end of the conveyor belt. Both Dobot robots are supplied with a standard 230V AC plug and are connected to a laptop using the USB serial port.

Our hardware setup can be transported inside a StormCase iM2975 from PELI, which can be pulled by a single person and fits into the trunk of regular car. Unpacking, assembly and connecting our hardware setup can be achieved by a single person within approximately 10 minutes. Other than having access to a power plug, screen/projector and WiFi, we do not rely on any other infrastructure provided by the school. Laptops or PCs are provided by the school or brought along by the participants.

4.2 Micro:bit Platform

The participants of our module will use the popular BBC micro:bit platform [11] to interact with our Snack-o-Mation system by sending radio messages. The second version (v2) of the micro:bit platform is built around an ARM Cortex-M4 microcontroller with accelerometer, magnetometer and microphone as sensor inputs, as well as two buttons and a small LED matrix display. The device can be connected via USB to a computer for programming and to supply it with power. For programming the micro:bit, we use the Microsoft

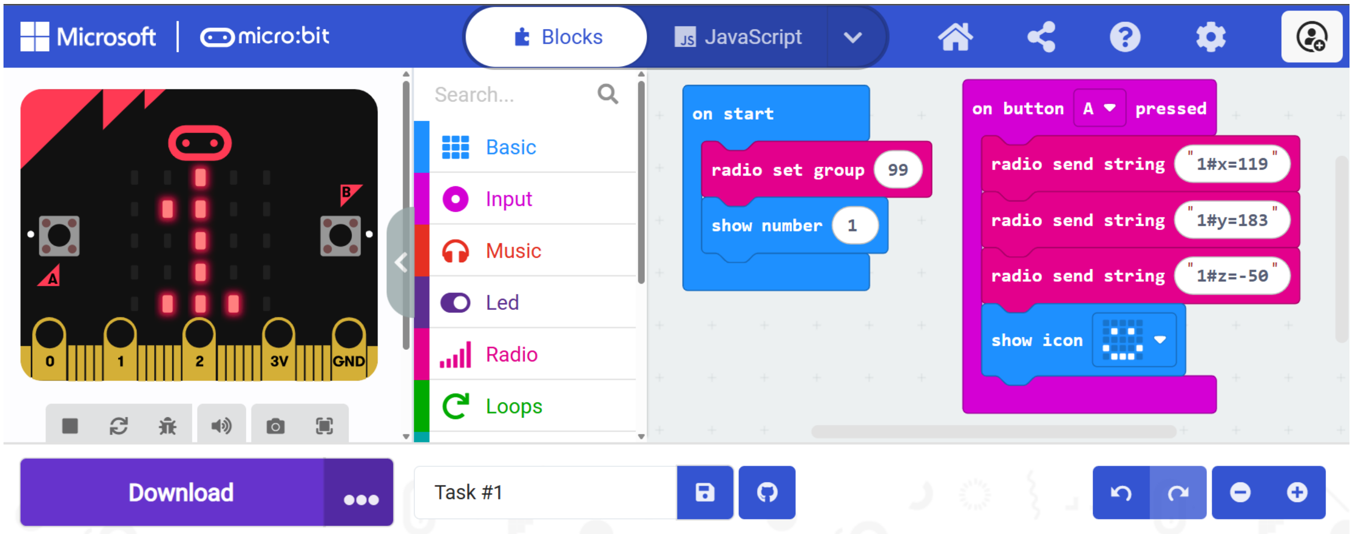

Figure 3: The Microsoft MakeCode environment is used by the participants to program the micro:bit platform in the browser.

MakeCode platform, which allows to program the micro:bit using a visual programming approach in the web browser, as shown in Figure 3. The micro:bit platform allows coding in Python as well, but we only recommend this to more experienced participants that have done this before.

We provide a dedicated micro:bit with the corresponding USB cable for each participant. Depending on the school, each participant is equipped with either a personal laptop or we use the school's computer infrastructure. Programming the micro:bit can be performed through WebUSB with any recent version of popular browsers (e.g., Chrome, Edge). If WebUSB is unavailable (e.g., on Safari or Firefox), users can download the binary and manually move it to the corresponding USB mass storage device provided by the micro:bit.

4.3 Controller & Visualization (GUI)

The controller is running on a laptop and implemented in the Python programming language. It communicates with the hardware, i.e., both robots and the micro:bit serving as the radio receiver, through the USB serial ports. It implements the state machine of our automation system, orchestrating both robot arms and the conveyor belt to perform the automation task. It keeps track of the coordinates of the different key positions, which are received through radio messages sent by the participants. We have implemented a graphical user interface (GUI), as shown in Figure 4. It is available in English as well as translated to all languages in which we provide the Snack-o-Mation module: German, French and Italian. The GUI is shown on the screen in front of the classroom during the hands-on exercises and is used to visualize the current state of the two robots, as well as to keep track of the progress of the different groups. The participants are able to read the coordinates of the current position of the two robot arms. Additionally, the interface facilitates monitoring of the wireless communication by showing incoming radio messages, allowing users to verify that their message was correctly transmitted and formatted. To support collaborative and competitive learning environments, the GUI can track and display the progress of each team, offering immediate feedback and fostering motivation. Completed sub-tasks are highlighted with green background color. Furthermore, the GUI serves as a control panel for initiating delivery tasks, managing orders, updating the amount of sweets in stock, as well as letting the students move the robot to the specified position. These capabilities collectively enhance usability, situational awareness, and engagement, making the GUI indispensable for both educational and operational contexts.

5 Survey Results

For each TecDay event, the organizers conducted a post-event survey amongst the participants of our modules. This survey is voluntary. In this section, we summarize the feedback received by 134 of the 450 participants from 12 TecDay events. Each participant was asked to rate our module along 4 dimensions (interactivity, entertaining, informative, difficulty), as well as to give an overall impression of the module.

5.1 Overall Rating

We show the overall ratings for the module for each of the 12 TecDay events in Figure 5. Overall, 59 out of 134 of participants (44%) rated our module with good, 39 (29.1 %) with very good, while 30 (22.3%) perceived it as average and 6 (4.4%) as poor. We can observe that the distribution of ratings changes between the different events, which could be explained through different facilitators, local setup, and differences in the background of the participants (e.g. age, focus of their studies). In particular, we noted worse ratings given as feedback for the second event, which were attributed to some technical issues we encountered with the robot during this particular event.

Furthermore, 94% of the participants perceived our module as interactive and 86% rated it as an entertaining experience. In total, 101 out of 134 participants (76%) indicated that the module was informative to them. In terms of the difficulty level, it was rated as too easy by roughly 20%, exactly right by 71%, and as too difficult by 9% of the participants.

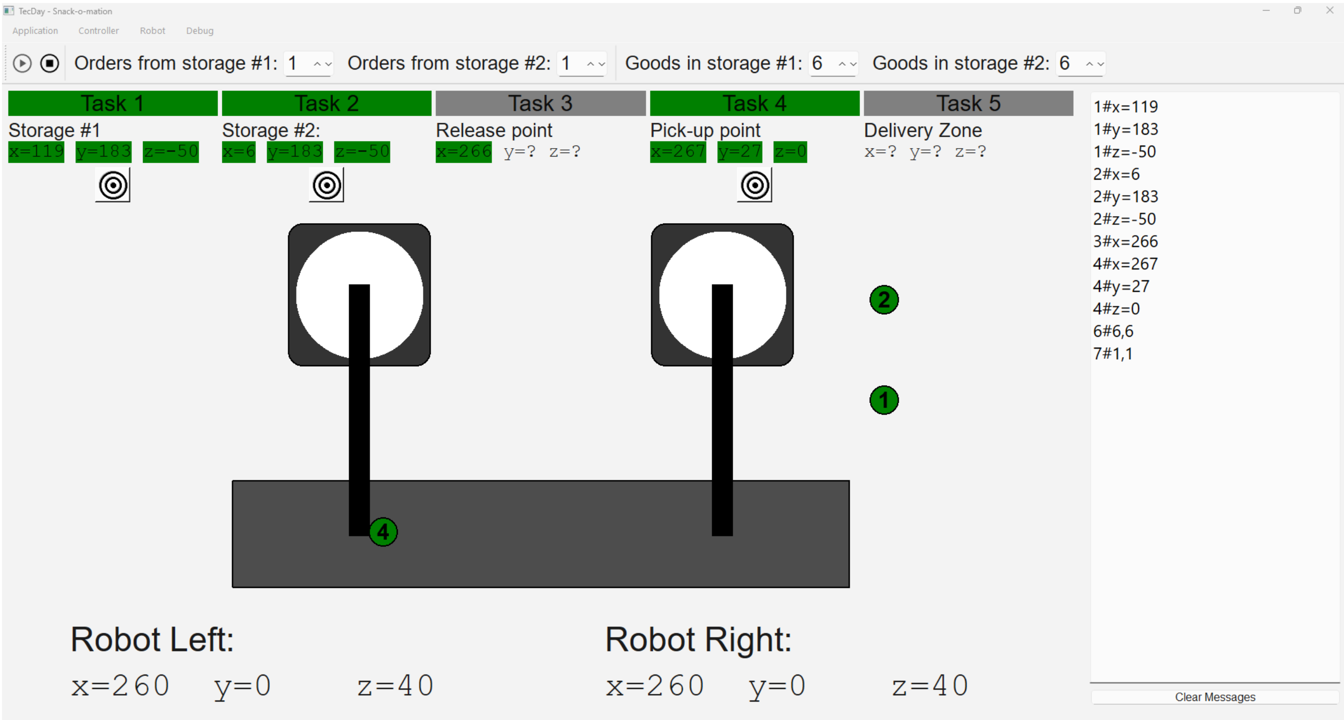

Figure 4: GUI: Visualization of the Snack-o-Mation system shown on the screen during the session. Green background colors indicate already completed tasks. Radio messages sent by the participants are visible on the right side of the screen.

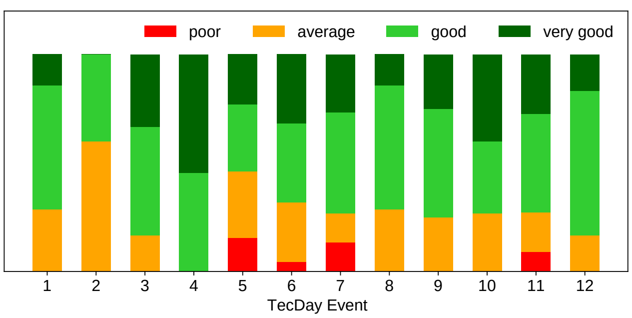

Figure 5: Survey results based on feedback from 12 TecDay events. Participants were asked to select an overall rating for the module from the given categories.

5.2 Free-text Comments

Besides the categorical ratings, students were asked to provide free-text comments regarding *(i)* what aspects they particularly liked and *(ii)* what could be improved.

Positive Aspects. They particularly enjoyed the hands-on experience of programming the robots and the opportunity to work collaboratively in groups. They appreciated the practical approach, the engaging and interactive sessions, and the ability to experiment and see their projects come to life. The competent instructors and the balance between instruction and active participation were also highlighted as positive aspects.

Improvement Suggestions. The students also suggested several improvements for the module. They felt that the programming tasks could be more challenging, especially for older students, and that the overall difficulty level should be increased. Additionally, they recommended more detailed explanations and introductions to the tools and concepts, as well as a clearer focus on the topic of sweets automation. A few students also suggested extending the duration of the module and incorporating more interactive and engaging activities.

6 Lessons Learned & Outlook

Several insights emerged from the facilitation of the module that inform best practices for instructional design in hands-on computing education.

Be Prepared For the Unknown. As the module is taught at different schools, we are usually not fully aware of the details of the facilities available in the classroom. Therefore, we always bring spare extension cables for power, USB and HDMI, as well as different adapters with us.

Concise Instructional Materials. We also observed that participants frequently overlooked detailed written instructions, indicating a need for more concise materials supplemented by verbal guidance and live demonstrations.

Structured Learning Environment. Time management was another critical factor. While students benefited from the opportunity to experiment with the micro:bit platform, examples shown on screen by the facilitators and strict timekeeping were necessary to maintain progress across groups. The presence of two facilitators proved advantageous, enabling one to provide support to individual groups while the other managed the overall session flow.

Accessible Programming. Interestingly, many students did not initially recognize that they were engaging in programming. This suggests that the activity design effectively abstracted complex concepts, thereby lowering the barrier to entry. However, certain technical elements—particularly the message format encoding (i.e. in what format to provide the coordinates) were perceived as difficult.

Adaptive Learning. Student groups are typically quite diverse in their level of prior programming experience. Task flexibility also contributed to the module's success. Optional extension tasks were provided for faster groups, allowing them to explore advanced topics without delaying others. For example, one additional task was to provide the start signal for a new delivery by sending a specially crafted radio message. In case no team was able to complete the additional task, we could start the delivery by pressing a button on the GUI instead.

Potential Extension - Distributed Systems. Future extension tasks could involve implementing decentralized control logic on the micro:bit, enabling devices to respond to events received via radio communication. This would not only introduce event-driven programming, but also encourage exploration of distributed systems concepts. However, we believe additional time during the session would be required to introduce these concepts.

7 Conclusion

We described *Snack-O-Mation* where students collaboratively solve a programming riddle to retrieve candy from a robot-based vending machine. It was met with great enthusiasm. Student feedback indicates that they enjoyed the practical aspect of the course, as well as the active participation. During the evolution of the course we were able to improve the setup for full reliability and quick reproducibility. Especially the exploration phase of the course is a considerable risk. We were however pleasantly surprised by the level of creativity of the students, and their sparked interest in IoT technology. The subsequent instruction-based phase of the course finally facilitates a positive and satisfactory outcome for the participants. It remains to say that while considerable effort was put into the preparation of this course, we think that it is worthwhile to motivate students early in their education to consider the STEM field as a future path.

8 Acknowledgments

We would like to thank the Swiss Academy of Engineering Sciences (SATW) and the many hosting schools throughout Switzerland for their support during the TecDays events. Furthermore, we would like to thank Felix Sutton for his support with the hardware setup and all our colleagues who helped to facilitate our past TecDay events: Andreas Port, Antoine Riaud, Simon Eicher, Enrico Fregnan, Christian Goettel.

References

[1] Timo Hynninen and Antti Knutas. How Well do Students Understand the All-Encompassing, Ubiquitous, and Interconnected Nature of IoT? Evaluating Student Capstone Projects. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education-Volume 1*, pages 244–250, 2022.

[2] Justus Bamert. What Makes a Role Model? STEM Participation and Exposure to Female STEM Experts. In *IZA/ECONtribute Workshop on the Economics of Education*, 2024.

[3] KOF Swiss Economic Institute. Can stem events attract more pupils to technical degree programmes? http://tiny.cc/ao3o001, 2024.

[4] Swiss Academy of Engineering Sciences (SATW). https://www.satw.ch/en/, 2025.

[5] Swiss Academy of Engineering Sciences (SATW). Nachwuchsförderung mit wirkung, tecdays. https://mint.satw.ch/de/tecdays, 2025.

[6] Nicholas Barendt, Nigamanth Sridhar, and Kenneth A Loparo. A new course for teaching internet of things: a practical, hands-on, and systems-level approach. In *2018 ASEE Annual Conference & Exposition*, 2018.

[7] Bill Siever and Michael P Rogers. Micro: bit magic: Engaging k-12, cs1/2, and non-majors with iot & embedded. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, pages 1237–1238, 2019.

[8] Yvonne Rogers, Venus Shum, Nic Marquardt, Susan Lechelt, Rose Johnson, Howard Baker, and Matt Davies. From the bbc micro to micro: bit and beyond: a british innovation. *interactions*, 24(2):74–77, 2017.

[9] Dobot. Magician robot. https://www.dobot-robots.com/products/education/magician.html, 2025.

[10] Dobot. Conveyor belt kit. https://www.dobot-robots.com/products/conveyor-belt/conveyor.html, 2025.

[11] Jonny Austin, Howard Baker, Thomas Ball, James Devine, Joe Finney, Peli De Halleux, Steve Hodges, Michał Moskal, and Gareth Stockdale. The bbc micro:bit: from the uk to the world. *Communications of the ACM*, 63(3):62–69, 2020.