Wireless Networking-Lab – A Project-Based Approach for Teaching WSN Fundamentals

Leonard Zurek
zurek@ibr.cs.tu-bs.de
TU Braunschweig
Germany

Jan Schlichter
schlichter@ibr.cs.tu-bs.de
TU Braunschweig
Germany

Felix Büsching
f.buesching@ostfalia.de
Ostfalia - University of Applied Sciences
Germany

Sven Pullwitt
pullwitt@ibr.cs.tu-bs.de
TU Braunschweig
Germany

Lars Wolf
wolf@ibr.cs.tu-bs.de
TU Braunschweig
Germany

ABSTRACT

Wireless Sensor Networks (WSNs) are ubiquitous in everyday life, playing important roles in industrial and civil appliances. Their usage can range from basic monitoring to complex distributed control systems. The growing adoption has led to a great availability of many hardware platforms.

As a result, teaching these topics has become vital for preparing students to operate and develop these platforms. Focusing on WSNs, this paper introduces the experiences gained over 10 years of teaching Internet of Things (IoT) through a dedicated course: the Wireless Networking Lab (WSN-Lab). We highlight its project-based approach, along with gamification techniques that we used to increase students' interest and motivation. The feedback of the students suggests that the open-ended structure of a project-based approach supports self-initiative, allowing students to delve deeper into specific aspects of IoT and WSN.

CCS CONCEPTS

• **Hardware → Wireless integrated network sensors**; • **Networks → Sensor networks**; • **Applied computing → Education**.

KEYWORDS

Wireless Sensor Networks, IoT, Project Based Learning

1 INTRODUCTION

The growing adoption of connected devices and monitoring systems in industry, and therefore, the implementation of Industry 4.0 [14], is a prime example of the growing adoption of the IoT. It is expected that this trend will continue not only in the industrial sector but also in private devices and other areas where more information and automated data processing can enhance process efficiency.

This not only poses new research questions and engineering challenges but also requires the education of new experts who tackle these problems. However, teaching IoT poses many challenges on its own that educators must address.

First of all, as with all teaching material, the courses about IoT need to be updated regularly to reflect the current state of the art. This is especially challenging if the course includes practical tasks that involve embedded hardware that has to be maintained and replaced at some point.

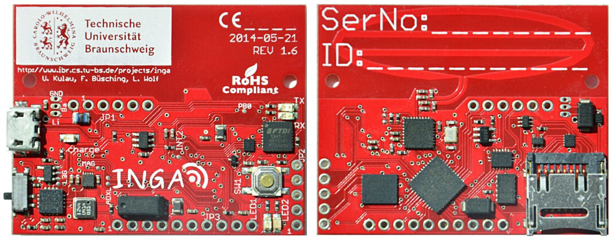
Furthermore, the selection of hardware must be carefully considered, depending on the course's focus. For example, enabling students to benefit from a broad base of resources, platforms like Arduino UNO [2] can be helpful. However, focusing more on teaching specific techniques such as battery-free computing, platforms like Riotee [9] are more suited.

In this paper, we present our experiences of teaching WSNs for over 10 years through the WSN-Lab. It is a practical course that teaches students basic techniques in WSNs before working on a larger practical task with minimal boundaries. This allows them to explore various parts of WSNs aligning with their interests. The course is equivalent to 5 ECTS.

The course began in the summer term of 2014, based on the INGA [5] hardware platform, which features various sensors and an ATmega 1284P Microcontroller Unit (MCU) connected to an AT86RF231 radio. The INGA was used together with Contiki [7] as Operating System (OS) until we switched to RIOT [3] as an OS in 2019. Shortly after, the INGA was retired and replaced by the IBR-Node in 2022. The new hardware platform was developed specifically for research and educational purposes and includes an STM32F411 MCU as a significant upgrade to the previous one.

Apart from the evolving hardware and software stack, the tasks for the students also varied over the years. In all cases, the course started with one or two introductory tasks, allowing students to familiarize themselves with the required tools and stack. For the main task that followed, several different variants were used. These range from allowing students to undertake a self-designed project to special tasks introduced by the supervisors, e.g., focusing on smart farming.

In the following, the complete history of the course, including the lessons learned, is shown in Section 2. Our general WSN-Lab teaching approach is presented in Section 3. The current state of the course is presented in Section 4, with a detailed description of the hardware used and an evaluation of student feedback in Section 5. The paper concludes with an outlook on future developments in Section 6, a discussion of related work in Section 7, and the conclusion Section 8.



(a) INGA Front View **(b) INGA Back View**

Figure 1: INGA Hardware Platform. INGA's Physical Dimensions are 50 mm x 39 mm

(a) Self Built Vehicle **(b) Gesture Controller**

Figure 2: Example of a Student Project from the WSN-Lab Summer Term of 2019 Based on INGA

2 HISTORY OF THE WSN-LAB

In 2001, our teaching already included educational labs for embedded systems, which featured several aspects of today's IoT. While these early lab iterations focused on the intersection of electrical engineering and computer science (Computer and Communication Systems Engineering), new topics and interests have emerged throughout the years. Getting all participants to a common level, these educational labs focus on the basic understanding of the 8-bit architecture and bus communication, as well as the integration of peripherals in embedded systems. To offer a course on more aspects of wireless communication in embedded systems, we established a new lab with a special focus on WSNs: the WSN-Lab. With the summer term of 2014, our WSN-Lab was successfully introduced.

2.1 Teaching with INGA

We introduced INGA as a teaching platform alongside the introduction of the WSN-Lab in 2014. With this inexpensive and low-power wireless sensor node, students learned how to monitor the environment efficiently. Equipped with an accelerometer, gyroscope, temperature & pressure sensor, SD-card slot, and an IEEE 802.15.4 radio, the node can store and transmit measured data. An in-depth description of INGA can be found in [5].

Initially developed for the GAL project [8], INGA filled the gap of an open-source wearable sensor node that met the project's requirements. As the nodes were to be used for gait analysis and fall detection on people, the sensor technology was evaluated intensively in advance [4]. Throughout the years, INGA was also used in other research projects, such as GINSENG [6], and the idea of using INGA more intensively in teaching emerged.

For teaching, three aspects were particularly conducive to including INGA: *1) Comprehensible processor architecture*, *2) Open Source Operating System & toolchains*, *3) Inexpensive components*.

By utilizing an 8-bit ATMega microcontroller, we already had a Central Processing Unit (CPU), which is easy to understand and not overloaded. Students with an interest in computer or electrical engineering who are familiar with the basics of computer architectures generally get to grips with the 8-bit computer quickly and comprehensively. For this target group, programming in C and examples in assembler do not usually present a significant challenge.

The support for OSs such as Contiki and the use of existing libraries made it interesting for students coming from the field of classical computer science, as it was possible to produce results that could be presented and experienced quickly. Being open-source, Contiki was a good match, as students could easily examine any part of the system and modify it if required.

With the open toolchain and debugging option JTAG, getting started is inexpensive and uncomplicated, as no licenses or special hardware are required. INGA's bootloader allows it to be programmed and put into operation via USB.

All in all, we decided that INGA builds a solid teaching platform for our courses. Unfortunately, throughout the years, Contiki development stopped, and hence, we switched to RIOT OS. It delivers the same flexibility due to its open-source structure while providing a broader range of supported hardware and, nowadays, a larger community.

3 WSN-LAB TEACHING APPROACH

Our teaching approach for the WSN-Lab is split into four main phases: *1) theoretical lecture*, *2) practice-oriented programming*, *3) self-study and development*, and *4) group project*.

Starting the first phase with 8-bit architectures, students familiarize themselves with basic operations on MCUs and utilize simple logics, such as interrupt routines, Pulse-Width Modulation (PWM), and serial communication, including Inter-Integrated Circuit (I^2C), Serial Peripheral Interface (SPI), and Universal Synchronous/Asynchronous Receiver-Transmitter (USART). Afterwards, they learn to work with registers and memory regions to operate peripheral devices.

Having gained this basic understanding, we proceed to phase 2, where students can explore the entire feature set of the MCUs, thereby strengthening their knowledge and gaining more experience on IoT devices. After a quarter of the semester, a colloquium assesses each student's acquired knowledge, and if passed, groups are formed to proceed with the course.

This closes phases 1 and 2, opening the possibility for each student group to prepare and present a project they will be working on for the rest of the semester, which forms phases 3 and 4. Regular meetings with each student group ensure that the problems faced are addressed at an early stage and progress is made. Additionally, these meetings oversee that phase 3 is covered properly and that the group project is finished by the end of the semester.

One of these student projects is shown in Fig. 2. The group chose to utilize two INGA nodes to realize a gesture-controlled vehicle. One node is responsible for reading the direction and sending steering commands, while the other uses them to control servos and motors inside the vehicle.

(a) INGA Node with Battery and Housing (b) Deployed Base-Stations in the Potato Field

(c) Base-Station at the End of the Potato Season

Figure 3: PotatoNet Special Edition WSN-Lab Deployment

In the summer term of 2017, we introduced a special edition in conjunction with the PotatoNet project [12]. Students monitored environmental data of one season on a potato field. Fig. 3 shows the housing for the nodes and the solar-powered base stations used for communication to a backend. The task was to program INGA nodes to collect as many environmental parameters as possible, with a trade-off between battery lifetime and the number of collected samples per day. Students had the opportunity to deploy the solar base stations they built and programmed together with the INGA nodes on a real potato field and test their approaches.

Providing space for ideas and embracing self-initiative in group projects has proven to be an efficient way to increase students' interest in the topic of IoT and WSNs.

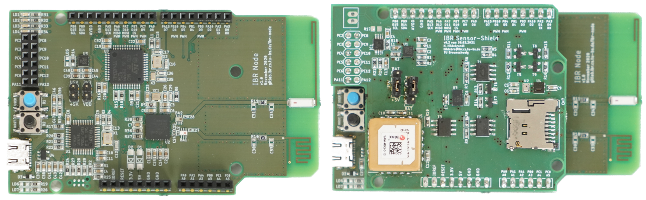
4 CURRENT STATE OF THE WSN-LAB

Throughout the years, rising demands for greater expandability and advances in hardware development have resulted in a new generation of WSN nodes for our lab: the *IBR-Node*. This feature-rich platform consists of a 32-bit MCU with a wealth of periphery and a dual-band IEEE 802.15.4 radio. Although the underlying hardware changed, the OS chosen for the nodes remains RIOT.

In the following, we will describe the IBR-Node in detail and present a special type for the project phase: the Industrial Communication Challenge (ICC), which were both introduced in summer term 2022.

4.1 Teaching with IBR-Node

Due to a higher demand on computational power and memory, as well as peripheral extendability, a new generation of sensor nodes



(a) IBR-Node Front View (b) IBR-Node & Sensor Shield

Figure 4: IBR-Node Hardware Platform Currently Used for the WSN-Lab. The Physical Dimensions are 53 mm x 90 mm

as teaching platform was developed. With the IBR-Node, a powerful yet lightweight node is created, moving from an 8 to 32-bit architecture. In contrast to INGA, the node is mainly designed for teaching purposes. However, due to the vast amount of periphery provided by the STM32 CPU, it is well-suited for many more scenarios.

As MCU, we chose the STM32F411RE, which provides an Arm Cortex M4 at 100 MHz, 512 kB Flash and 128 kB SRAM. Its periphery includes, among others, 16 channel 12 Bit Analog to Digital Converter (ADC), 16 & 32 Bit timers, a Real Time Clock (RTC) and several USART, I^2C and SPI ports.

Serial communication is enabled via the USB-C connector and a Black Magic Probe (BMP) programmer [13]. BMP is an open-source embedded programmer providing a serial bridge and a GNU Debugger (GDB) server for debugging.

Wireless connectivity is provided by the AT86RF215 dual-band transceiver. This radio module is IEEE 802.15.4 compliant and features a wide range of PHY modes. Drivers for this radio are available in RIOT, which facilitates teaching, as students can adapt and learn low-level functionality. Furthermore, it can be used to learn how to operate heterogeneous links and multiple connections simultaneously, enabling concepts such as Multi-Connectivity (MC) [15]. By utilizing RIOT's Generic Network Stack (GNRC), students can use this radio in conjunction with IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) to apply prior knowledge in communication and networks. Nevertheless, they can solely work on the MAC layer or exchange the medium access to incorporate Time Division Multiple Access (TDMA) protocols, if prior knowledge of these techniques exists, which we teach in other courses.

With Printed Circuit Board (PCB) antennas, the node is ready to use and lowers costs for additional parts and connectors. M3 mounting holes are included on the board for securing the node in a potential housing. The pinout design follows the Arduino UNO I/O header, which enables the use of many extension shields. For our use cases, we built a custom sensor shield carrying common modules such as temperature & humidity, air pressure, GPS receiver, and an SD card slot. Fig. 4 shows the IBR-Node and the sensor shield attached to it. This sensor-shield can also be equipped with a LiPo battery, which is charged and monitored by the sensor shield. Alternatively, the IBR-Node's onboard power management can be turned off and an energy harvesting shield can be used to power the node directly, covering most WSN scenarios.

Although we utilize custom designs, skills acquired in our lab can be easily applied to other platforms, as the IBR-Node functions very similar to boards offered by chip manufacturers, such as the

STM32 Nucleo with an additional radio. We deliberately chose the STM32 chip over other platforms, such as the ESP32, due to the very low power requirements and comprehensive documentation for the STM chips. Furthermore, the radio for our board is connected via SPI instead of a register mapping inside the MCU. This provides more control over the drivers, allowing curious students to dive deeper into radio communication, as it is not an abstract class/binary firmware. With RIOT as a basis, any part of the OS can be investigated and practically taught. Now, being in the 4th iteration of our course utilizing the IBR-Node, it has been well established as a learning platform.

4.2 Industrial Communication Challenge

To further enhance learning and attitude towards our WSN-Lab, we have integrated gamification techniques. We decided to introduce them in an alternative to the group project, as it has shown to be improving students' accomplishments and attitude [16]. This alternative has formed the Industrial Communication Challenge (ICC) and is now in alternation with the group project every year.

To implement gamification, we embedded a task within a fictitious industrial environment. The task consists of monitoring environmental data, such as humidity and temperature, while maintaining a multi-hop network. The groups are faced with requirements regarding the sample rate and artificially induced interference. Key challenges to solve are meeting timing constraints to realize real-time monitoring, forwarding data from distant nodes to a central sink, and remaining operative if the band is disrupted. Thus, this task incorporates several key aspects for resilient and precise monitoring of industrial scenarios.

The actual challenge involves measuring the performance of all groups. Each group assigns a sink node, which collects measurement data from all other nodes on an SD card. Each measurement point includes a timestamp, a temperature, and a humidity reading. After a 5-minute run, the data is retrieved and checked for timing inconsistencies and the overall temporal resolution.

With regular status meetings, student progress and problems are supervised and addressed. Three months into the course, a dry-run for the ICC is held. One week later, all groups are concurrently running their WSN and a winner is determined by comparing the sample rates, message throughput, and stability. To increase ambition, students can additionally compete against a team of PhDs. After all WSNs have been evaluated and a winner has been determined, the course finishes by presenting solutions and a relaxed exchange at a barbecue.

The underlying task of the challenge was chosen to isolate specific aspects of WSN development while simultaneously dedicating comprehensive attention to the remaining components. In the current scenario, students focus on efficiency and performance, diving deeper into data structures, threading, and MC. Time-synchronization is realized through the Global Positioning System (GPS) module on the sensor-shield and energy consumption is not considered as a performance metric. Thus, no complex time-synchronization mechanisms have to be integrated and handling of deep-sleep states can be neglected.

By deliberately eliminating the power consumption and time synchronization, albeit fundamental aspects of most WSNs, the students can concentrate on the performance aspect. Therefore, students gain the opportunity to have a more in-depth look at one specific aspect of WSNs without the complexity of balancing the partly conflicting design goal of such networks.

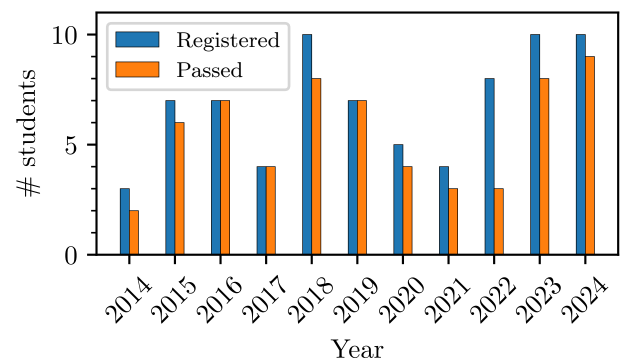


Figure 5: Student Success Rate over Past 10 Years

5 STUDENT FEEDBACK

Regular assessment of courses through student evaluation is mandatory to safeguard good lecturing practice. Although we can address intermittent problems during the course and mitigate them in future versions, we rely on student feedback. In the following, we will discuss some of the feedback throughout the years of our WSN-Lab. The questionnaire given to students consists of 7 different categories. Forming a mixed-method approach, they include inter alia, a self-assessment, content, teamwork, overall rating, and a comment section. Each section is rated on a 5-point Likert scale, reaching from 1 (very good) to 5 (bad). Although the questionnaires can provide valuable insights, students are not obliged to complete them. Furthermore, the evaluation is not returned to the lecturers if fewer than 5 sheets are returned to preserve anonymity.

To assess our WSN-Lab based on student feedback, we can only consider those semesters for which we received evaluations. However, to obtain qualitative feedback on the ICC and IBR-Node, we asked students from their first iteration to comment on the structure and organization of the course, as only 3 students finished the course. Additionally, we measure demand based on course registrations compared to actual successful participation.

Fig. 5 depicts this metric over 10 years, starting with the summer term of 2014. Due to examination regulations, registration for the course at our institute solely reserves a slot and is not binding. To receive a grade, students must sign up for the course at the examination office within a 4-week window towards the end of the semester. Unfortunately, we have no data regarding the point in time when students stopped participating in the course, whether they did not attend the kickoff or dropped out during the semester.

Several points of interest can be observed in Fig. 5. First, only 4 students registered in 2017, which is almost half compared to 2015, 2016, and 2018. This was due to the special edition of the course, which could only offer 4 slots as a result of an increase supervision overhead for the real-world deployment and available hardware.

Secondly, we can observe a decrease from 2020 to 2021, which was closely related to the Corona pandemic, as we had to limit the course slots. Lastly, in 2022, half of the students stopped participating in the course. Although one might suspect that this was due to the introduction of the IBR-Node and the ICC, results of 2023 and 2024 show that the overall demand increased.

The overall student rating since introducing the IBR-Node shows an average of 1.64 with 14 votes for the WSN-Lab over 3 semesters. According to the 5 point Likert scale reaching from 1 (very good) to 5 (bad), this suggests that the course design is well accepted by students. With an average rating of 1.54, students stated to recommend the course to others. When asked about their opinion on working in a group of unknown people, the feedback showed an average of 2.8, with 11 votes and 3 abstentions, suggesting a somewhat unsatisfactory condition. However, students reported getting along with their team quite well, with an average rating of 1.3, which mitigated the initial doubts about working in an unknown group.

Regarding the ICC, students overall liked the challenge concept, with one student stating: "*The challenge really motivated us to not only get the task done, but also to try to get the best solution possible*". Another student reported: "*I appreciate the flexibility of the course structure. It is open-ended, allowing us to explore various solutions using the provided microcontroller*". Especially the first phases were highlighted as a good preparation for the group phase. This suggests that our course is well suited for teaching IoT and WSN fundamentals.

6 FUTURE OF THE WSN-LAB

With the current WSN-Lab setup now in its 4th iteration, we have established a stable teaching routine for working with the IBR-Node and alternating between group projects and the ICC. However, we aim to expand the possibilities for the course by providing a broader set of assignments to form additional challenges for students to choose from.

The current ICC focuses on efficient data acquisition and networking. Students do not have to solve the difficult task of time-synchronization between nodes, as we impose the usage of the GPS modules. We are currently working on another evaluation criteria regarding power consumption. By utilizing power profilers and logic analyzers, we aim to incorporate energy consumption as an additional metric in the ICC.

Furthermore, the ICC will be held in a real industrial environment, used for research, as opposed to the outdoor scenario currently employed. This will increase the challenges through signal reflections and interference and provide students with more accurate insight into real-world deployments. However, this adds the overhead of time-synchronization, if we move to an indoor environment due to the absence or poor quality of GPS signals.

Moving to the proposed environment also increases the possibilities for future tasks, as students can utilize the available infrastructure, such as area-wide WiFi and a 5G campus network. Even though the use of both transmission technologies requires additional hardware, this presents a great opportunity to teach students about the trade-off between using these technologies in IoT.

7 RELATED WORK

With the increasing popularity of smart-home and wearable devices, IoT hardware has been widely adopted in teaching. Many platforms, such as Arduino, ESP, and Raspberry Pi, have made acquiring hardware and accessing knowledge easy. However, provided examples are often limited and can not replace the supervision and knowledge gained in a course.

Being familiar with programming MCUs is only one part of IoT. Integrating it into a cloud infrastructure for persistent monitoring and remote access poses another key challenge.

An undergraduate course that features all aspects of an IoT-edge-cloud continuum is introduced in [11]. In this course, students familiarize with basic Arduino hardware, reading sensor values, and controlling actuators. They proceed with an integration to an edge device, which awaits sensor values from the Arduino via MQTT and, in turn, forwards it to a cloud server that handles storage. Although this course teaches valuable insights into IoT, its structure is strictly predefined, with little room for students' own ideas.

A more open course with an urban gardening project is introduced in [10]. With this project-based approach, the authors wanted to increase student motivation and support self-learning in a rapidly developing area such as IoT. However, their course was split into two phases, with students from different years working on each phase. Nevertheless, valuable insights into student perceptions are gained from a questionnaire given to course students, where 41 % stated that working on a realistic scenario and 21 % chose self-learning as the best aspect of the course.

A more comprehensive review of 60 papers showcasing the usage of IoT in teaching can be found in [1]. Many of the analyzed papers utilize Commercial Of-The-Shelf (COTS) hardware for their courses, emphasizing the advantage of many IoT devices due to their comparatively low price and easy extendability. While many courses favor project-based learning approaches, they often do not focus on low-power technologies such as IEEE 802.15.4, in contrast to our hardware. With environmental monitoring and home-automation being the top categories of student projects, they emphasize the use of real-world scenarios for teaching.

8 SUMMARY & LESSONS LEARNED

Having gained over 10 years of teaching experience with our WSN-Lab, we have established a reliable and easily extendable learning platform. Our IBR-Node delivers high computational power for wireless sensor nodes and in conjunction with our sensor-shield, it can be used to monitor environmental conditions, persistently saving or forwarding data in a multi-hop network. However, problems arise when nodes malfunction or break due to incorrect handling, as a custom board is more expensive to replace. This is mainly a result of incautious transportation or removal of shields, where some pins can break. Although this rarely occurs, due to small production sizes, costs can reach up to 100€despite base components being comparably inexpensive. Additionally, software has to be updated regularly to keep up with upstream changes in RIOT.

By utilizing RIOT, most of the software parts are written in C, which requires students to have a basic understanding of this programming language. Although C++ or Rust could be used on supported CPUs and modules, it is highly encouraged to complete the course in C. Throughout the years, this has never shown to be a problem for students with little knowledge of C.

Throughout the years, students have rated the course to be improving their skills in the area of embedded devices and networking. Student feedback evaluation yields a positive response, strengthening the use of a project-based approach in teaching IoT. This structure allows students to focus deeply on their own projects, embracing self-initiative and being encouraging. As this course is a group project, one of the most important factors is group dynamics. Furthermore, gaining experience in software projects and working in a team is a key lesson for students to learn. Feedback shows that although many students are initially reluctant to work in a group with unknown people, they generally get along quite well.

ACKNOWLEDGMENTS

This work was partially funded by Land Niedersachsen (NBank) under grant number *ZW 1-80162367*.

REFERENCES

[1] Pramod Abichandani, Vidhusheidhaa Sivakumar, Deepan Lobo, Craig Iaboni, and Prateek Shekhar. 2022. Internet-of-Things Curriculum, Pedagogy, and Assessment for STEM Education: A Review of Literature. *IEEE Access* 10 (2022), 38351–38369. https://doi.org/10.1109/ACCESS.2022.3164709

[2] Arduino. 2025. Arduino UNO R3 User Manual. https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf. Accessed: 2025-05-26.

[3] Emmanuel Baccelli, Cenk Gündoğan, Oliver Hahm, Peter Kietzmann, Martine S. Lenders, Hauke Petersen, Kaspar Schleiser, Thomas C. Schmidt, and Matthias Wählisch. 2018. RIOT: An Open Source Operating System for Low-End Embedded Devices in the IoT. *IEEE Internet of Things Journal* 5, 6 (2018), 4428–4440. https://doi.org/10.1109/JIOT.2018.2815038

[4] Felix Büsching, Ulf Kulau, Matthias Gietzelt, and Lars Wolf. 2012. Comparison and validation of capacitive accelerometers for health care applications. *Computer Methods and Programs in Biomedicine* 106, 2 (2012), 79–88. https://doi.org/10.1016/j.cmpb.2011.10.009 Design of Environments for Ageing.

[5] Felix Büsching, Ulf Kulau, and Lars Wolf. 2012. Architecture and evaluation of INGA an inexpensive node for general applications. In *SENSORS, 2012 IEEE*. 1–4. https://doi.org/10.1109/ICSENS.2012.6411295

[6] Felix Büsching, Wolf-Bastian Pöttner, Dieter Brökelmann, Georg von Zengen, Robert Hartung, Karsten Hinz, and Lars Wolf. 2012. A demonstrator of the GINSENG-approach to performance and closed loop control in WSNs. In *2012 Ninth International Conference on Networked Sensing (INSS)*. 1–2. https://doi.org/10.1109/INSS.2012.6240572

[7] A. Dunkels, B. Gronvall, and T. Voigt. 2004. Contiki - a lightweight and flexible operating system for tiny networked sensors. In *29th Annual IEEE International Conference on Local Computer Networks*. 455–462. https://doi.org/10.1109/LCN.2004.38

[8] Marco Eichelberg, Andreas Hein, Felix Büsching, and Lars Wolf. 2010. The GAL middleware platform for AAL. In *The 12th IEEE International Conference on e-Health Networking, Applications and Services*. 1–6. https://doi.org/10.1109/HEALTH.2010.5556589

[9] Kai Geissdoerfer and Marco Zimmerling. 2024. Riotee: An Open-source Hardware and Software Platform for the Battery-free Internet of Things. In *Proceedings of the 22nd ACM Conference on Embedded Networked Sensor Systems* (Hangzhou, China) *(SenSys '24)*. Association for Computing Machinery, New York, NY, USA, 198–210. https://doi.org/10.1145/3666025.3699332

[10] Javier Hormigo and Andrés Rodríguez. 2019. Designing a Project for Learning Industry 4.0 by Applying IoT to Urban Garden. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje* 14, 2 (2019), 58–65. https://doi.org/10.1109/RITA.2019.2922857

[11] Admela Jukan, Xavi Masip-Bruin, Jasenka Dizdarević, and Francisco Carpio. 2023. *Network of things engineering (NoTE) lab* (2023 ed.). Springer International Publishing, Cham, Switzerland. https://doi.org/10.1007/978-3-031-20635-1

[12] Ulf Kulau, Sebastian Schildt, Stephan Rottmann, Björn Gernert, and Lars Wolf. 2015. *PotatoNet – Outdoor WSN Testbed for Smart Farming Applications*. Technical Report. Erlangen, Germany. https://www.ibr.cs.tu-bs.de/papers/kulau-fgsn2015.pdf

[13] Black Magic Probe. 2025. Black Magic Debug Repository. https://github.com/blackmagic-debug/blackmagic. Accessed: 2025-05-26.

[14] Klaus Schwab. 2016. *The fourth industrial revolution*. Crown Currency.

[15] Marie-Theres Suer, Christoph Thein, Hugues Tchouankem, and Lars Wolf. 2020. Multi-Connectivity as an Enabler for Reliable Low Latency Communications—An Overview. *IEEE Communications Surveys & Tutorials* 22, 1 (2020), 156–169. https://doi.org/10.1109/COMST.2019.2949750

[16] Ibrahim Yildirim. 2017. The effects of gamification-based teaching practices on student achievement and students' attitudes toward lessons. *The Internet and Higher Education* 33 (2017), 86–92. https://doi.org/10.1016/j.iheduc.2017.02.002