In-Situ Assistance for Fluctuating Maker's Expertise in the Continuous Prototyping Process

Yoonji Kim*

Chung-Ang University, yoonji.h.kim@gmail.com

Hye-Young Jo

Chung-Ang University, jhy.vfx@gmail.com

Physical computing prototyping is a complex task that involves both software and hardware. In continuous prototyping scenarios, a diverse range of makers may encounter a variety of obstacles that arise from their differing objectives, expertise levels, and prototyping procedures. Thus, it is essential to provide makers with in-situ assistance during physical computing prototyping to bridge the gap between conception and creation. To address this issue, we have developed systems that aid makers in physical computing prototyping. Our previous research involved formative studies that identified the challenges makers face during the prototyping process. Based on the issues we discovered, we have designed systems that offer in-situ assistance in physical computing, enabling users of varying knowledge levels to prototype their concepts. In this workshop, we introduce our previously proposed systems and discuss how to support makers in different stages of the prototyping process as their expertise levels fluctuate over time.

CCS CONCEPTS • Human-centered computing • Human computer interaction (HCI)

Additional Keywords and Phrases: Physical computing, electronic prototyping, personal fabrication

ACM Reference Format:

Yoonji Kim and Hye-Young Jo. 2023. In-Situ Assistance for Fluctuating Maker's Expertise in the Continuous Prototyping Process. In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany. ACM, New York, NY, USA, 3 pages

This work was presented at the **CHI2023 Workshop [WS2] - Beyond Prototyping Boards: Future Paradigms for Electronics Toolkits**

CHI '23, April 23-28, 2023, Hamburg, Germany

1 FLUCTUATING MAKER'S EXPERTISE

Many researches and tools for physical computing prototyping primarily focused on novice or expert makers, however, most makers are perpetually at an intermediate level. Some are experts in software or hardware but novices in the other fields, and they are rarely experts or novices in all disciplines required for physical computing. The difficulties of designing interaction for physical computing leads to makers' different experience levels. In addition, according to Cooper [1], "Although everybody spends some minimum time as a beginner, nobody remains in that state for long. People don't like to be incompetent, and beginners, by definition, are incompetent. Conversely, learning and improving is rewarding, so beginners become intermediates very quickly — or they drop out altogether." Commonly, user interaction designs come from satisfying these intermediate users with different levels of experience [1], but it is enormously more complicated to unleash interaction design for makers because of the multiple levels of experience that exist in multidimensional areas.

2 IN-SITU ASSISTANCE SYSTEMS FOR PHYSICAL COMPUTING PROTOTYPING

Makers encounter various challenges that stem not only from the constantly shifting and evolving nature of their expertise but also from the complexity of the problems they need to tackle. These problems arise from diverse goals and multiple processes, requiring makers to be proficient in a range of technical and creative skills. To address these difficulties related to the multidimensional nature of physical computing prototyping, researchers have proposed various tools that aim to simplify the involved tasks (software and hardware) for novices [2] or provide functionalities, i.e., showing the inner states of a live circuit [3], for experts. However, those systems have yet fully supported the varying expertise range of users in the corresponding continuous prototyping iteration and situation. Consequently, bridging the gap between idea and prototype requires providing makers with in-situ assistance during physical computing prototyping to help overcome the challenges of fluctuating expertise levels and ensure a smooth iteration process.

To address these challenges, we developed a suite of systems (figure 1) for in-situ assistance in physical computing to enable users with varying levels of knowledge to prototype their ideas. We conducted formative studies to investigate makers' difficulties throughout the exploring, prototyping, and testing/debugging stages of the process. Based on our findings, we developed three distinct systems to address the identified issues.

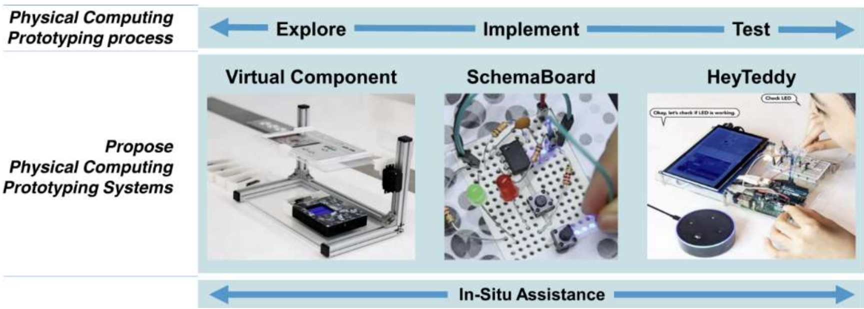

Figure 1: Developed physical computing prototyping systems. VirtualComponent for exploring circuit design, SchemaBoard for circuit implementation and inspection, and HeyTeddy, test-driven assistance system, for developing software of hardware behavior

2.1 VirtualComponent

VirtualComponent [5] is a modular mixed-reality tool that allows users to construct a breadboard circuit with a combination of physical and virtual components. Using an augmented-reality interface, users can place virtual components onto a physical breadboard and adjust their values and properties within the software. These modifications are instantly reflected in the electrical characteristics of the physical breadboard. For instance, the resistance between the terminals of a virtual resistor can be measured in physical form, and its value corresponds to the value set in the software.

2.2 SchemaBoard

SchemaBoard [6] is an augmented breadboard that supports the transition from on-screen schematic representations of a circuit to its corresponding physical instantiation on a breadboard. SchemaBoard facilitates the assembly and inspection tasks of circuit design by maintaining a dual representation of circuit diagrams and providing in-situ visualization using an LED matrix underneath the breadboard. This enables users to correctly place components on the breadboard and quickly check whether the layout is consistent with the schematic diagram.

2.3 HeyTeddy

HeyTeddy [4] is a conversational agent designed to facilitate the programming and execution of code on an Arduino device in real time through natural language dialogue. The system enables users to operate the device without the need to write actual code. HeyTeddy can be accessed through either a voice-based interface or a web chat. Spoken or written commands are parsed, interpreted, and executed in real-time, resulting in physical changes to the hardware. For instance, the "write high" command configures an I/O pin to behave as a digital output with its internal state set to high, thereby enabling driving an LED. Thus, the user is relieved from the task of writing, compiling, and uploading code on the hardware. Moreover, HeyTeddy provides TDD (Test Driven Development) functionalities, supervises the user's choices, and guides them through each step needed to assemble the circuit, preventing incorrect logic (e.g., writing an analog value to a digital pin). The system offers users the option of testing individual components through separate unit tests without interrupting their workflow. Finally, HeyTeddy allows the user to export the issued commands as written code for Arduino, thereby providing an Arduino sketch in C++ ready for upload.

ACKNOWLEDGMENTS

This research was supported by the Chung-Ang University Research Grants in 2021.

REFERENCES

[1] Alan Cooper, Robert Reimann, and David Cronin. 2007. About face 3: the essentials of interaction design. John Wiley & Sons.

[2] Fraser Anderson, Tovi Grossman, and George Fitzmaurice. 2017. Trigger-Action-Circuits: Leveraging Generative Design to Enable Novices to Design and Build Circuitry. In Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17). ACM, New York, NY, USA, 331–342. DOI: https://dl.acm.org/doi/10.1145/3126594.3126637

[3] Will McGrath, Daniel Drew, Jeremy Warner, Majeed Kazemitabaar, Mitchell Karchemsky, David Mellis, and Björn Hartmann. 2017. Bifröst: Visualizing and checking behavior of embedded systems across hardware and software. In Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology. ACM, New York, NY, USA, 299–310. DOI: https://dl.acm.org/doi/10.1145/3126594.3126658

[4] Yoonji Kim, Youngkyung Choi, Daye Kang, Minkyeong Lee, Tek-Jin Nam, and Andrea Bianchi. 2020. HeyTeddy: Conversational Test-Driven Development for Physical Computing. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 3, 4, Article 139 (sep2020), 21 pages. DOI: http://dx.doi.org/10.1145/3369838

[5] Yoonji Kim, Youngkyung Choi, Hyein Lee, Geehyuk Lee, and Andrea Bianchi. 2019. VirtualComponent: A Mixed-Reality Tool for Designing and Tuning Breadboarded Circuits. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19). Association for Computing Machinery, New York, NY, USA, 1–13. DOI: http://dx.doi.org/10.1145/3290605.3300407

[6] Yoonji Kim, Hyein Lee, Ramkrishna Prasad, Seungwoo Je, Youngkyung Choi, Daniel Ashbrook, Ian Oakley, and Andrea Bianchi. 2020. SchemaBoard: Supporting Correct Assembly of Schematic Circuits Using Dynamic In-Situ Visualization. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20). Association for Computing Machinery, New York, NY, USA, 987–998. DOI: http://dx.doi.org/10.1145/3379337.3415887