ESPBoost: A Rapid Prototyping Toolkit for Helping Designers Create the Internet of Tangible Things

Sark P. Xing
Department of Industrial Design
Eindhoven University of Technology
Eindhoven, the Netherlands
sark.xing@gmail.com

Yaliang Chuang
Department of Industrial Design
Eindhoven University of Technology
Eindhoven, the Netherlands
y.chuang@tue.nl

ABSTRACT

Although the applications of IoT are significantly developed in the past decade, most of the user interfaces are designed with the graphical or voice modality provided on another device, such as a smartphone or speaker. Researchers have started to investigate how to promote rich interactions embodied in physical IoT objects namely the Internet of Tangible Things (IoTT). However, most studies mainly focused on the explorations of tangible concepts. Due to the technical bottleneck, very few concepts were actually implemented and validated with their complete idea. There is a need for rapid prototyping toolkits to help designers easily assemble technical components to build tangible prototypes and evaluate the user experiences in real contexts. In this position paper, we present an under-development toolkit called ESPBoost. It has modular hardware featured in four aspects: *connectivity, tangible input, tangible output, and power management*. We explain the details of our toolkit and how it could facilitate the designers' iterative creation process. By applying the fabricated ESPBoost onto a challenging design case, we also surface ESPBoost's limitations and plan an evaluation study for investigating its efficacy. With ESPBoost, we aim to provoke designers' creations of the user experiences of IoTT without spending significant time and effort in trying to configure endless choices of electronic components.

CCS CONCEPTS

• Human-centered computing ~ Human computer interaction (HCI) ~ Interactive systems and tools ~ User interface toolkits

KEYWORDS

Internet of Tangible Things, IoTT, rapid prototyping

1 INTRODUCTION

Internet of Things (IoT) breakthroughs the way we work, live and study. It brings huge of opportunities to augment traditional products to be connected and interactive over Internet. Examples of applications include asking virtual assistant to make a taxi reservation [12], feeding a pet remotely from a mobile application while traveling [14], or having real-time notes or sketches synchronization between an application and a smart pen [12]. In brief, IoT builds connectivity for everything as ecosystems that allow users to leverage and interact with online services [6]. Yet, the dominant interaction types of these services tend to be screen or voice interaction.

To enrich the diversity of interaction styles, introduces the Internet of Tangible Things (IoTT). It suggests connecting tangible interfaces with IoT objects by exploiting the tangible interaction properties. Building upon this vison, IoTT (and -related) researchers have concentrated mainly on two splits: the ideation tool, using cards to generate concepts, such as [2,3,8,15], sets of design cards with divergent types of hints (e.g., tangible interaction properties, human actions, electronic components, etc.), and the prototyping tool, providing toolkits to support prototyping, e.g. [5] a home sandbox equipped with modular controlling interface. However, compared the goal of IoTT with the current state of developed tools, the exploration and evaluation of IoTT in an embodied and tangible way remain sparse, but why?

This might be on account of the IoTT prototyping complexity as it demands interdisciplinary knowledge of prototyping IoT and tangible interaction as presented by [11]. This resembles the dilemma of developing shape-changing interface, as in Alexander's [1] words, "Prototyping shape-changing interfaces requires knowledge of complex electronics and mechanical engineering that go beyond that typically required in other areas of interactive computing — software programming or simple electronics.". As a consequence, the aim of this paper is framed, that is, designing a prototyping solution that lowers the barriers for both IoT and tangible interaction.

With the goal in mind, we took a glance at design projects that have the characteristics of IoTT and selected one that is the most challenging. Knowing the challenges of implementing IoTT, we propose a custom toolkit named ESPBoost, which encapsulates four modular sets: *connectivity, tangible input, tangible output, and power management.* We then developed and fabricated a few samples for a design practice and technical evaluation. Lastly, we describe the next-step development with synthesized design criteria, aiming to involve other designers and engineers into the iteration of *ESPBoost*, and/or to provoke designers and researchers to implement designs in IoTT.

The contributions by far are two folds. First, we fabricated a custom PCB named *ESPBoost* and use it to prototype a novel design case of IoTT. The developed toolkit is fully open sourced, allowing senior engineers to iterate; designers to explore other implications that the toolkit is capable of; Second, based on the reflective analysis of the prototyping process, we generate a table of taxonomy and design considerations for *ESPBoost.*

2 PROTOTYPING IoTT

2.1 IoTT Concept

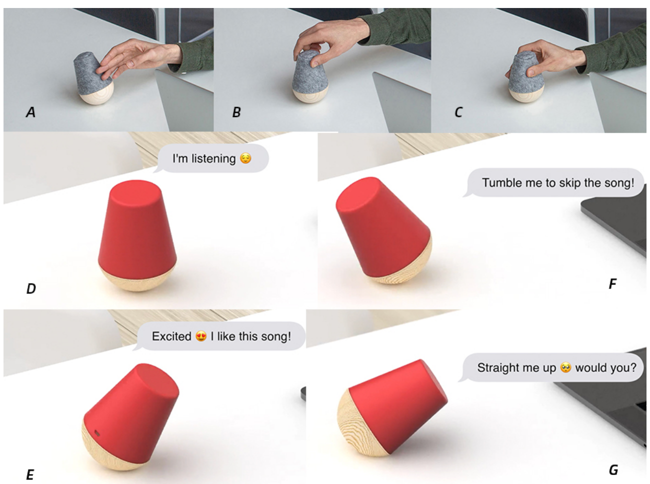

Figure 1 Functions of *Topplr*: A) skipping songs by tumbling, B) changing volumes by rotation, C) pausing/resuming by squeezing. D) Calm (*gently wobbling*): enjoying the music, E) Excited (*vibrantly wobbling*): really enjoying the song, F) unpleasant (*shaking*): asking for tumbling to the skip a song, G) exhausted (*unable to get up*): asking for help.

To navigate our design process, we explored and chose an IoTT design concept called *Topplr*, a roly-poly like tangible music interface that consists of two sets of functionalities: 1) tangible input: *Topplr* featuring in three types of input (see Figure 1 A-C) and 2) tangible output: wobbling itself to express different emotions in response to different contexts (see Figure 1 D-G). Although the design concept has some envisioned functionalities , it hasn't been prototyped in practice for further evaluation. This might be a suitable design case for us to explore and reflect on the challenges of developing IoTT prototypes.

2.2 Prototyping Challenges

By exanimating the concept with the knowledge we possess and references we consult, we synthesized the challenges of prototyping *Topplr* :

Handling large current

Typical tangible interaction prototypes, particularly those shape-changing interfaces, tend to involve motors to create actuation, which usually draws large current. The current prototyping solution (e.g., Tilting Bowl [7]) generally leverages an MCU board paired with driver boards to handle these current. However, looking up the datasheet of each amplifying chip to find the suitable drivers could be a comprehensive and time-consuming task. Not to mention that some of the design novices are not even aware of a driver is a necessity, causing the prototyping failures.

Portability with tangible interaction

Most tangible interaction prototypes (e.g., [4,9,10]) seem to be constrained in a fixed place as they are wired for constant-power supply. However, in this design case, *Topplr* is designed to be running without a cable, as otherwise the cable might accidentally prevent or help *Topplr* tumbles down. Although some marketed esp boards (e.g.,[16]) that come with a Li-Po charger would allow prototypes running without external power supply, their power regulating solution usually does not allow to provide sufficient current to drive motors. .

Compact dimension with sophisticate components

Since *Topplr* contains two motors and one battery, the remaining space for containing other modules (e.g., esp-integrated MCU board, gyro and accelerometer sensor, power regulator, etc.) is considerably limited. It is indeed a challenge to maintain the compact dimension of *Topplr* while encapsulating all necessary components. Meanwhile, such a challenge could be general among IoT projects that incorporate tangible interaction quality and portability.

3 ESPBoost

Knowing the mentioned challenges, we thought that it might be convenient to encapsulate all complementary modules into one custom PCB board and pair it with certain MCU board. As such a kit (or in other words shield) can extend or boost up the capability of esp-chipped MCUs, we name it the ESPBoost. It is as simple as to use other shields, simply attaching the kit onto the MCU board by correctly matching the GPIO pins. Then, the supported motors, pressure sensor and battery shall be wired up via the connectors and no extra datasheet looking up and soldering needed; On the software side, we used OOCSI [6], a programming-friendly platform that regulates the communication between cross-platform clients (e.g., esp32/esp8266, Python programs, Processing, etc.). This would allow *Topplr* to be controlled remotely from different interfaces (see Figure 2 E) and *Topplr* could also react in response.

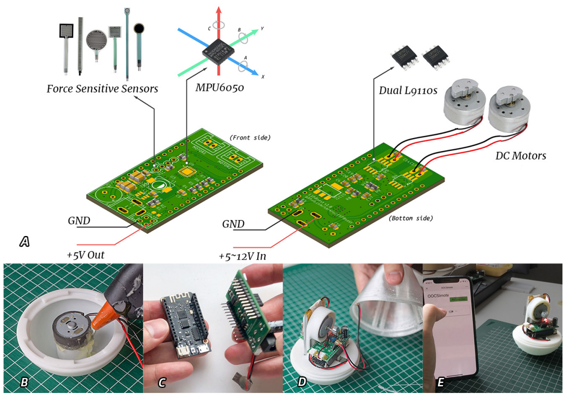

Figure 2 A) ESPBoost has built-in motor drivers (*L9110s*), gyro and accelerometer sensor (*MPU6050*); Force sensitive sensor connectors wired with 10K pull-down resistors; C) easy assembly of ESPBoost onto an MCU board; D) Giving force to send commands to PC via Bluetooth BLE (*ESP32*); E) Sending commands from a Web interface (*OOCSImote*) to manipulate *Topplr*.

4 DISCUSSION

Based on our first-person prototyping experience with ESPBoost, we recognized that the developed toolkit: ESPBoost can indeed simplify the prototyping workflow and guarantee reliable performance. Also, such an elegant prototyping solution could help designers replicate connected IoTT prototypes easily, particularly in the context of Internet of Things, where objects are usually plural. Specifically, imagine if we want to create *multi-Topplr* interaction, it would be handy to use ESPBoost for the duplication.

However, we also learned about its limitations, that is, firstly, the supported peripherals of ESPBoost are insufficient to cover all prototyping scenarios; Secondly, current ESPBoost uses a pinout definition explicitly for one specific MCU board, whereas there are tons of MCU boards with different pinout map on the market. To address that, we generalized some suggestions for both engineers and designers to evolve ESPBoost.

4.1 ESPBoost refinement

For engineers, we hereby highlight some aspects that may demand extra attention when refining ESPBoost:

Portability

To achieve portability, power management is the key. ESPBoost needs to be able to regulate large (de)charging current, at the same time, providing selectable regulated power voltages for different electronic components.

Extendibility

Normalizing all tangible in(out)put modules as much as possible. For instance, sensors and drivers that communicate in *i2c* could use a unified JST connector.

Usability

In parallel with leveraging advanced technologies, the solution should retain user-friendliness to IoTT practitioners who have non-expert technical background. For example, we suggest using unified jst connectors to for all modules that communicates in *i2c* instead of asking the user to find the proper GPIO pins from the breakout board.

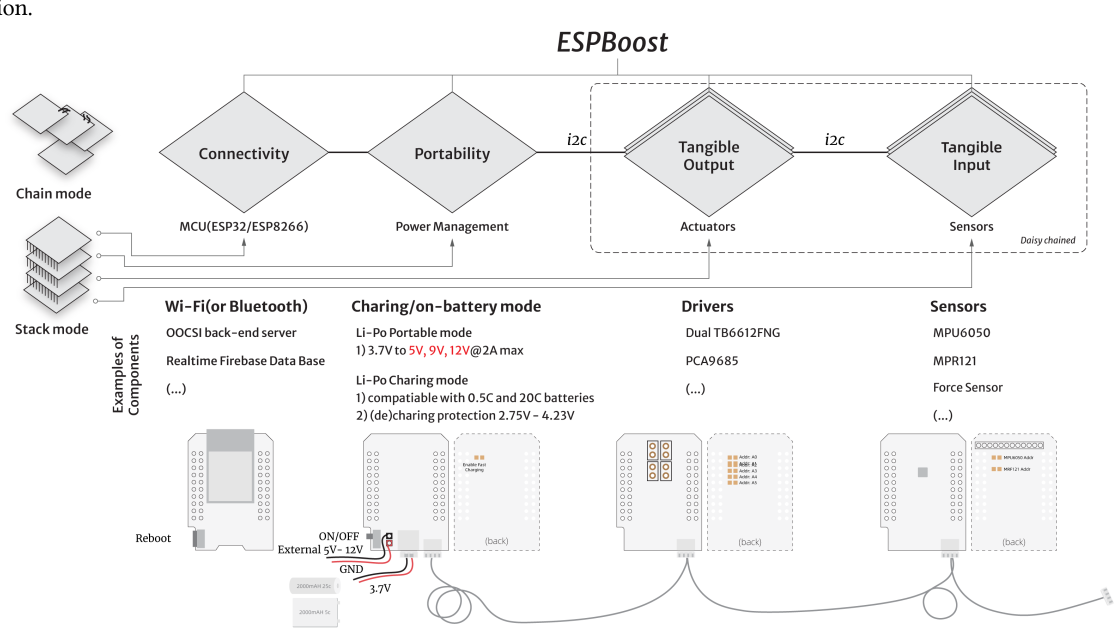

Figure 3 We split ESPBoost into four sub-catagory: *Connectivity, Portability, Tangible output and Tangible input*; The tangible in(out)put category may contain multiple stackable modules with addressable *i2c* pads at the back; all these modules share a unified *i2c* connector for great extendibility, allowing them to be wired up with an *i2c cable.*

4.2 Usability and efficacy evaluation study

For designers, we encourage them to evaluate the usability and efficacy of ESPBoost and return constructive feedback. To foster the development of IoTT, we hereby provide our suggestions to conduct user studies with ESPBoost.

The study will be conducted among (*n=10*) participants with ideally even prototyping competences and sufficient IoT and tangible interaction knowledge. All participants will be required to complete a questionnaire regarding practical questions for prototyping, followed by the introduction of IoTT and ESPBoost. Then, both groups of participants will be given complementary components (e.g., variants of sensors and actuators) while only the experimental group (*n=5*) will be distributed ESPBoost boards. The controlled group will explore their own way of prototyping. All participants will be assigned to a design task: designing a tangible interface for Philips Hue [13] in a given time span of two weeks. After which, all participants will be interviewed individually for reporting experiment feedback. Lastly, the participants will be informed about requirements of the designed interfaces as follows:

- Connectivity: allowing the user to control a light bulb (e.g., on/off, brightness) at home over the **Internet**.
- Tangibility: allowing the user to leverage their **bodily movement** to manipulate the above-mentioned parameters.
- Actuation: incorporating at least one kind of **tangible output** for pragmatic purposes.
- Portability: allowing the interface to run **without an external cable**.

The study will be evaluated based on the following criteria:

- Degree of completion: Are participants able to finish tasks and meet the design requirements? Is the prototype functioning well for evaluation? And how far do they reach?
- Aesthetics of interaction: Is the created interaction intriguing or intuitive?

4.3 Future of ESPBoost

The long-term vision of this study is to support designers in implementing their IoTT concepts. If the usability and efficacy of ESPBoost are validated, further study that evaluates whether it can foster creativity can be carried out. Since ESPBoost has been open sourced on GitHub [17] under the MIT license, it can serve as a hardware fundamental for the future development of IoTT.

5 Conclusions

We identified a crucial need of implementing IoTT in practice. We accordingly developed and reported an under-development toolkit to prototype IoTT concepts elegantly. By applying the fabricated ESPBoost onto a challenging design case, we surfaced ESPBoost's limitations and synthesized considerations for the next-step implication and evaluation studies. We provoke senior engineers to evolve the toolkit, and/or to encourage designers and researchers to practice prototyping IoTT with ESPBoost.

REFERENCES

[1] Jason Alexander, Anne Roudaut, Jürgen Steimle, Kasper Hornbæk, Miguel Bruns Alonso, Sean Follmer, and Timothy Merritt. 2018. Grand Challenges in Shape-Changing Interface Research. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, ACM Press, Montreal QC, Canada, 1–14. DOI:https://doi.org/10.1145/3173574.3173873

[2] Leonardo Angelini, Elena Mugellini, Omar Abou Khaled, and Nadine Couture. 2018. Internet of Tangible Things (IoTT): Challenges and Opportunities for Tangible Interaction with IoT. *Informatics* 5, 1 (March 2018), 7. DOI:https://doi.org/10.3390/informatics5010007

[3] Massimiliano Dibitonto, Federica Tazzi, Katarzyna Leszczynska, and Carlo M. Medaglia. 2018. The IoT Design Deck: A Tool for the Co-design of Connected Products. In *Advances in Usability and User Experience* (Advances in Intelligent Systems and Computing), Springer International Publishing, Cham, 217–227. DOI:https://doi.org/10.1007/978-3-319-60492-3_21

[4] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. 2013. inFORM: dynamic physical affordances and constraints through shape and object actuation. In *Proceedings of the 26th annual ACM symposium on User interface software and technology - UIST '13*, ACM Press, St. Andrews, Scotland, United Kingdom, 417–426. DOI:https://doi.org/10.1145/2501988.2502032

[5] Joep Frens, Mathias Funk, Bastiaan van Hout, and Joep Le Blanc. 2018. Designing the IoT Sandbox. In *Proceedings of the 2018 on Designing Interactive Systems Conference 2018 - DIS '18*, ACM Press, Hong Kong, China, 341–354. DOI:https://doi.org/10.1145/3196709.3196815

[6] Mathias Funk. 2019. *OOCSI.* Zenodo. DOI:https://doi.org/10.5281/zenodo.1321220

[7] Henry Lin, Ron Wakkary, and Doenja Oogjes. 2019. The Tilting Bowl: Electronic Design for a Research Product. In *Proceedings of the 2019 on Designing Interactive Systems Conference - DIS '19*, ACM Press, San Diego, CA, USA, 345–357. DOI:https://doi.org/10.1145/3322276.3323701

[8] Simone Mora, Francesco Gianni, and Monica Divitini. 2017. Tiles: A Card-based Ideation Toolkit for the Internet of Things. In *Proceedings of the 2017 Conference on Designing Interactive Systems* (DIS '17), Association for Computing Machinery, New York, NY, USA, 587–598. DOI:https://doi.org/10.1145/3064663.3064699

[9] Ken Nakagaki, Daniel Fitzgerald, Zhiyao (John) Ma, Luke Vink, Daniel Levine, and Hiroshi Ishii. 2019. inFORCE: Bi-directional `Force' Shape Display for Haptic Interaction. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction - TEI '19*, ACM Press, Tempe, Arizona, USA, 615–623. DOI:https://doi.org/10.1145/3294109.3295621

[10] Majken Kirkegård Rasmussen, Timothy Merritt, Miguel Bruns Alonso, and Marianne Graves Petersen. 2016. Balancing User and System Control in Shape-Changing Interfaces: a Designerly Exploration. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction - TEI '16*, ACM Press, Eindhoven, Netherlands, 202–210. DOI:https://doi.org/10.1145/2839462.2839499

[11] Imran A. Zualkernan, Fadi Aloul, Eilaf Algebail, Mohamed El Refaay, Abdulkarim Ali, and Omar El Sabaa. 2020. Little Genius: An Experiment in Internet of Tangible Learning Things. In *2020 IEEE 20th International Conference on Advanced Learning Technologies (ICALT)*, 69–71. DOI:https://doi.org/10.1109/ICALT49669.2020.00028

[12] 2017. Ordering Ubers with an Amazon Echo. Retrieved October 15, 2020 from https://www.youtube.com/watch?v=o26xCTP6n2Q

[13] 2020. Philips Hue. Retrieved February 4, 2020 from https://www2.meethue.com/en-us

[14] Interactive Pet Camera. Retrieved October 15, 2020 from https://shopus.furbo.com/

[15] Learn. Play. Collect. *Know Cards.* Retrieved October 16, 2020 from https://know-cards.myshopify.com/

[16] Adafruit HUZZAH32 ESP32 Feather. Retrieved February 1, 2021 from https://www.adafruit.com/product/3405

[17] ESPBoost. Retrieved February 1, 2021 from https://github.com/sarkrui/ESPBoost