Robotany: An Environmentally-Aware, Autonomous Plant-Hybrid
(Technical Paper)

An Exploration into Trusted Execution Environment Vulnerabilities
(Technical Paper)

The Rise of Open Source Hardware: A Sociotechnical Perspective
(STS Paper)

A Thesis Prospectus Submitted to the

Faculty of the School of Engineering and Applied Science
University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements of the Degree
Bachelor of Science, School of Engineering

Jason Ashley
Fall, 2020

Technical Project Team Members
Robotany:
Eleanor Ozer
Zach Hicks
Noelle Law

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Introduction

Behind the veil of maker spaces, programming courses in schools, and individual projects that are “just for fun,” a silent revolution stirs. This revolution is the result of a series of innovations that rely immensely on openly sharing and modifying other people’s work, compared to the once more common model of proprietary works (Nascimento & Pólvora, 2018). Hardware-based innovations utilizing this sharing are often referred to as “open source hardware” (Herrera, 2020). Open source hardware allows a broader range of people to explore fields traditionally locked behind high cost and experience requirements (Nascimento & Pólvora, 2018). As a result, such technologies are now available in a much more widespread capacity, enabling their use in education, in individual projects, and within community spaces, further widening access to and knowledge of each of these technologies. Understanding the rise of this technology, its impacts, and where this technology will move related fields in the future is crucial to ensuring segments of the population are not left behind. This discussion will specifically analyze projects such as Arduino, 3D printers, and RISC-V.

Beyond this discussion, the first technical project will work towards developing an environmentally-aware, autonomous plant-hybrid system based on expansions to previous, open systems. This system will utilize the Internet of Things to report details and statistics related to the growth and health of some plant within some contained vehicle, allowing a user to view the growth progress of the plant via a secure mobile application. This system will have the ability to measure action potentials produced by the plant through connected electrodes, such as those caused by wounding, movement, or shifting (Aditya et al., 2011). These action potentials are interpreted as electrical signals and measured to a highly accurate degree with inexpensive and

open hardware designs. One of the primary goals will be for the system to reposition itself to optimize the amount of light the plant receives based on these action potentials on a daily basis, serving as a look into the future of plant-based sensing applications at a small scale.

Finally, for the second technical project, research into Trusted Execution Environments on certain processors will be performed to understand what security vulnerabilities exist within these systems and how they can be mitigated, likely through analysis of execution patterns. Virtually every CPU, whether in a phone, laptop, or embedded system, open source or not, utilizes a Trusted Execution Environment (TEE) to run code that handles privileged details, such as user passwords, biometrics, security keys, and more (Cerdeira et al., 2020). With the rise of inexpensive, accessible hardware, devices are now deployed everywhere, many of which contain sensitive information protected by TEEs. However, each of variation of TEE deployed to this point exhibit some sort of security vulnerability that leaks sensitive information in some way or another. These vulnerabilities could compromise accounts on a phone, lead to the misuse of deployed sensors, and more. Thus, it is important to secure these environments.

Technical Topic – Robotany

Bioelectric sensing is a rapidly growing field that utilizes the latent signals within living organisms as an inexpensive, resilient, and effective sensory device. Biosensor technology has applications ranging from security to smart agriculture systems (Aditya et al., 2011). The Robotany is an early exploration into this field, presented as an environmentally-aware, autonomous plant-hybrid that tracks moisture levels, monitors growth, and moves according to the plant's lighting needs. Equipped with a moisture sensor and camera, the device allows the plant to notify the user via a mobile application when it should be watered and updates about its

growth. Additionally, two electrodes, attached to leaves on either side of the plant, will measure the plant's bioelectrochemical signals and give the plant the ability to move around based on perceived need. When a plant is exposed to variation of temperature or light, drought, soil pollution, or any other perturbation, the plant produces electrical potentials to communicate between plant tissues and organs (Volkov & Ranatunga, 2006). These potentials, known as bioelectrochemical signals, resemble nerve impulses in humans and animals, and can be measured using electrodes (Gage, 2017). These signals are relatively weak, typically in the range of 1-50 mV, depending on the stimulus and the plant (Cai & Qi, 2017). Thus, a system for amplifying these signals to be read by low-cost hardware must be developed.

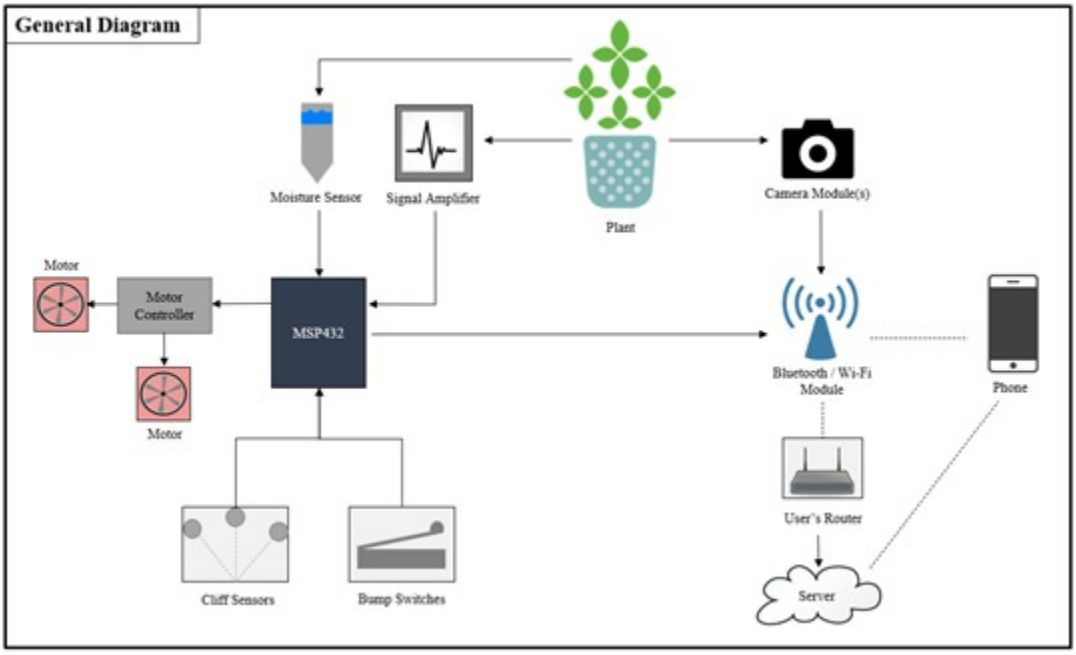

Figure 1. System block diagram of the proposed project.

As demonstrated by Figure 1, the plant is connected to a signal amplifier circuit that filters and amplifies the signal. Any signal from the plant will then be clean and readable for the MSP432, a low-cost, readily available microcontroller that is capable of reading varying input

voltages. This microcontroller will then determine whether or not to move the plant, signaling the motor controller. The MSP432 will also be connected to a series of safety devices, notably cliff sensors and bump switches, that will prevent the mobile plant from tumbling over a ledge or running into a wall. A camera, mounted above the plant, will capture the growth of the plant over time, filtering out the background to prevent unintended data collection. A WiFi module (ESP32, another inexpensive microcontroller) will process, format, and send the data from the camera and MSP432 to a server. This information will be shared with a user using a mobile application built on React Native. The system will be developed by a research team of four undergraduate engineering students and advised by Professor Harry Powell and Professor Adam Barnes.

Throughout the development of this project, the goals are to learn about the field of biosensing, expand knowledge of the Internet of Things, and utilize affordable, widely available hardware to create integrated systems. Beyond the technologies previously described, 3D printing and free CAD software will be utilized to rapidly develop and prototype casing for the robotic core, sensors, and plant pot. Thus, this will explore many areas that are rapidly developing, while also utilizing many well-established technologies. The final deliverables will consist of the prototype, a demonstration of its complete functionality, and a report detailing how the prototype may be replicated with suggestions for future development. Early development started in September of 2020 and has been making steady progress. Initial hardware development work should finish by early November of 2020, with software following shortly thereafter, likely late November. The final report and demonstration will be available in early December.

Technical Topic – TEE

Trusting computers is hard. Every day, news articles about the latest hacks appear in the news. Thus, it would seem important to ensure the information within a computer is safely stored. Trusted Execution Environments (TEEs) are tiny, segmented operating systems that run in parallel to the primary operating system the user interacts with that, ideally, is only supposed to run trusted code and hide secret information from the outside world (Sabt et al., 2015). Functions of a TEE could include methods for authenticating users with biometrics, storing keys for accounts, or even ensuring a malicious actor has not taken control of a device. It is composed of several major components necessary for starting and running consistently, notably the secure boot, secure scheduling, inter-environment communication, secure storage, and I/O path (Guilbon, 2018). The secure boot is responsible for loading trusted code, ensuring it has not been modified, and starting it from a known, safe start point, while the secure scheduling is necessary to ensure trusted and untrusted code do not run at the same time, an issue that could lead to information leaking. Utilizing a layer of the environment called the separation kernel, calls to the trusted and untrusted world (inside and outside of the TEE, respectively), are isolated, while an interaction layer is permitted and controlled through this system, allowing for the inter-environment communication specified. Secure storage focuses on the idea of ensuring the data the TEE protects is guarded in some manner and that, even with physical access, it cannot be easily stolen. Finally, the secure I/O (input/output) path ensures that data sent, say, to a biometric scanner, cannot be stolen by a malicious application in the untrusted world.

Throughout the use of a TEE and trusted code, trust is evaluated, both against a set of requirements and against the current state of the system running the TEE. However, this is a

rather difficult problem for a myriad of reasons, most notably due to the behavior of undefined interactions between the trusted and untrusted worlds (Cerdeira et al., 2020). TEEs must interact with the untrusted world, where user code should run, in order to share the information they are responsible for managing. This interface is often large, giving malicious attackers many points to attack. As such, numerous security vulnerabilities have been identified, often by passing bad inputs to the communication interface. Rather than return an error or denying access to this information, there have historically been many cases where the TEE happily provides this information to the malicious application, allowing for vast amounts of secure data to be stolen from the system. Certain TEEs run on top of processors that also handle user requests. If a trusted application performs an operation at the same time the malicious actor performs one, the malicious actor can time its own operations, occasionally giving it useful insight into what the trusted world is doing. Beyond this, several boot exploits have been discovered, bypassing secure boot, allowing arbitrary, untrusted code to run even in the TEE. Such exploits break the idea of separation and lead to the whole system being untrustworthy.

Research is being performed on one implementation of a TEE, ARM's TrustZone, to first replicate known exploits and work towards finding methods to detect active use of these exploits on a system. From here, further work will look into how to implement systems to detect active attempts to exploit a known vulnerability in TrustZone implementations. A method for determining if a particular piece of code is untrustworthy, as well as a paper documenting the development process and results, are the chosen deliverables. Early research started in October of 2020, experimentation with known exploits will occur later within 2020 into 2021. Development of the methods and paper will start in early 2021 and will go into the late spring.

STS Topic

Several open source hardware projects with recent, rapid growth will be studied so as to understand their known or potential social impact. Microcontrollers and single board computers, most notably Arduino products and designs, will be discussed first. Though a relatively new project, Arduino boards made an impact in several key areas (Kushner, 2011). Most notably, Arduino boards offered significantly lower cost and expandable microcontrollers with an easy to use toolchain. As a result, utilizing a microcontroller in a hobbyist or personal project became relatively inexpensive and easy compared to proprietary industry tools previously available. Arduinos also opened opportunities to teach embedded hardware and software programming to a much younger age group (Heradio et al., 2018). 3D printers will be the second case, though there will be some significant overlap as many of these systems utilize inexpensive, open microcontrollers as well (Simons et al., 2019). Studying the impacts of 3D printers will be useful for understanding the impacts of open source hardware outside of embedded systems, changing prototyping and manufacturing by significantly reducing costs and turnaround time (Siddique et al., 2019). Many 3D printer designs were developed and made open source, allowing others to replicate them, creating a self-propogating technology (Oltean, 2017). Further, the rise of 3D printers (among other fabrication tools) led to the rise in makerspaces, collaborative spaces where individuals or small groups can work on projects with specialized tools for a low cost (Nascimento & Pólvora, 2018). Finally, a more future-looking case study will focus on the RISC-V instruction set architecture (ISA) for computer processors. RISC-V, pronounced “risk-five,” RISC meaning Reduced Instruction Set Computer, is an open standard for building a computer processor (Legenvre et al., 2020). RISC-V supports “extensions” on the basic

processor, allowing for processor designers to add what is necessary for their purpose, such as for working with floating point numbers, modifying the processor to fit their needs. Anyone designing and utilizing a RISC-V processor can use a wide-range of software that already supports the architecture and its extensions (Gupta et al., 2017). In this manner, many different groups and companies can design processors or utilize off-the-shelf designs without the overhead of licensing an ISA or processor design (normally at a substantial cost). With the openness of the ISA, it has already found its way into education, and it has the potential to significantly lower the barrier for working with custom computer architecture with a very modern and well-supported architecture specification. However, there is still some necessary work that will aid in its adoption, making it a very young case of open source hardware.

Throughout the analysis, these cases will be analyzed through the lens of technologies accelerating due to a continued increase in its use cases and societal impact, technological momentum. Technological momentum is the flow of a technology from that of one that starts small and is impacted by society and history, but grows to a point where that technology itself impacts society (Hughes, 1993). The rise of open source hardware, especially in the selected cases, demonstrate various characteristics of a technology with momentum, making it a very useful framework for analyzing these cases and helping understand current and future growth, as well as the impacts that may result. One potential point of discussion is that not all technology is subject to generating momentum, so current analyses and discussion that demonstrate the aspects indicating momentum will be highlighted, refraining from speculation. Thus, a paper on this topic will be produced as a deliverable and will focus on first defining and providing background details for each of these cases, describing their impacts, then highlighting where that technology

fits in the more general idea of open source hardware. Research on the topic began in October of 2020, the paper on this sociotechnical subject will be prepared for the end of the Spring 2021 semester.

Research Question and Methods

With these cases in mind, the research will focus on one particular question. How has the rise of open source hardware designs impacted the knowledge of the related fields for consumers? Based on their widening use in education and by the general public, it is clear that these tools have an impact, but how so? Research will be collected about published literature, including knowledge surveys, novel applications, and analyses of the various feature sets of each case will aid in understanding why these tools are popular and how they are being used. This research will be combined with recent discourse of benefits and pitfalls open source hardware in various settings (either the term in general or for one of the cases mentioned), while also determining where there is a current lack of complete research. As the goal is to understand the impact of these technologies on the individual, research will also encompass some of the noted effects, such as hobbyist projects and makerspaces. Thus, the range of terms will include “open source,” “open source hardware,” “OSHW,” “Arduino,” “3D print,” “RISC-V,” “makerspace,” and many similar terms. Searching for literature and discussions based on these terms will be useful for determining the impact on individuals, both those in education and those working on individual projects, by explaining where research on the topic focuses and showing the perception of these technologies in discussion.

Conclusion

Three projects have been outlined, each exploring a budding research area in their respective fields. The Robotany will explore biosensory and how it may be used to understand the needs of a plant, adjusting position and alerting users as necessary to the needs of the plant the robot cannot fulfil for the plant itself while also providing an easy way to monitor its effectiveness. Research into Trusted Execution Environments will help understand what makes systems trustable, how even trusted systems can leak information or run untrusted code, and how those systems can be modified or extended to catch a malicious actor attempting to break the layers of trust. Finally, a look into the current world of open source hardware through several case studies, each at a different level of maturity, will provide a glimpse into the impacts of hardware with openly shared designs on the general consumer and students.

References

Aditya, K., Udupa, G., & Lee, Y. (2011, December 20). *Development of Bio-Machine Based on the Plant Response to External Stimuli* [Research Article]. Journal of Robotics; Hindawi. https://doi.org/10.1155/2011/124314

Cai, W., & Qi, Q. (2017, October 10). *Study on Electrophysiological Signal Monitoring of Plant under Stress Based on Integrated Op-Amps and Patch Electrode* [Research Article]. Journal of Electrical and Computer Engineering; Hindawi. https://doi.org/10.1155/2017/4182546

Cerdeira, D., Santos, N., Fonseca, P., & Pinto, S. (2020). SoK: Understanding the Prevailing Security Vulnerabilities in TrustZone-assisted TEE Systems. *2020 IEEE Symposium on Security and Privacy (SP)*, 1416–1432. https://doi.org/10.1109/SP40000.2020.00061

Gage, G. (2017, April). *Transcript of "Electrical experiments with plants that count and communicate."* TED. https://www.ted.com/talks/greg_gage_electrical_experiments_with_plants_that_count_and_communicate/transcript

Guilbon, J. (2018, June 19). *Introduction to Trusted Execution Environment: ARM's TrustZone*. https://blog.quarkslab.com/introduction-to-trusted-execution-environment-arms-trustzone.html

Gupta, G., Nowatzki, T., Gangadhar, V., & Sankaralingam, K. (2017). Kickstarting Semiconductor Innovation with Open Source Hardware. *Computer*, *50*(6), 50–59. https://doi.org/10.1109/MC.2017.162

Heradio, R., Chacon, J., Vargas, H., Galan, D., Saenz, J., De La Torre, L., & Dormido, S. (2018). Open-Source Hardware in Education: A Systematic Mapping Study. *IEEE Access*, *6*, 72094–72103. https://doi.org/10.1109/ACCESS.2018.2881929

Herrera, A. (2020). The Promises and Challenges of Open Source Hardware. *Computer*, *53*(10), 101–104. https://doi.org/10.1109/MC.2020.3011080

Hughes, T. (1993). Technological Momentum. In L. Marx & M. Roe Smith, *Does Technology Drive History? The Dilemma of Technological Determinism* (pp. 101–113). Massachusetts Institute of Technology. https://collab.its.virginia.edu/access/content/group/6ac8ef1f-6b15-4912-a488-d8c7468046db/Readings/Hughes%20-%20Technological%20Momentum.pdf

Kushner, D. (2011, October 26). *The Making of Arduino—IEEE Spectrum*. IEEE Spectrum: Technology, Engineering, and Science News. https://spectrum.ieee.org/geek-life/hands-on/the-making-of-arduino

Legenvre, H., Kauttu, P., Bos, M., & Khawand, R. (2020). Is Open Hardware Worthwhile? Learning from Thales' Experience with RISC-V. *Research Technology Management*, *63*(4), 44. https://doi.org/10.1080/08956308.2020.1762445

Nascimento, S., & Pólvora, A. (2018). Maker Cultures and the Prospects for Technological Action. *Science and Engineering Ethics*, *24*(3), 927–946. https://doi.org/10.1007/s11948-016-9796-8

Oltean, S.-E. (2017). A Practical Approach to the Design and Implementation of a Low Cost 3d Printer Using Open Source Technologies. *Scientific Bulletin of the Petru Maior University of Targu Mures*, *14*(2), 5–10.

Sabt, M., Achemlal, M., & Bouabdallah, A. (2015). Trusted Execution Environment: What It is, and What It is Not. *2015 IEEE Trustcom/BigDataSE/ISPA*, *1*, 57–64. https://doi.org/10.1109/Trustcom.2015.357

Siddique, T. H. M., Sami, I., Nisar, M. Z., Naeem, M., Karim, A., & Usman, M. (2019). Low Cost 3D Printing for Rapid Prototyping and its Application. *ArXiv:1911.10758 [Cs]*. http://arxiv.org/abs/1911.10758

Simons, A., Avegnon, K. L. M., & Addy, C. (2019). Design and Development of a Delta 3D Printer Using Salvaged E-Waste Materials. *Journal of Engineering*, *2019*, 1–9. https://doi.org/10.1155/2019/5175323

Volkov, A., & Ranatunga, D. R. (2006). Plants as Environmental Biosensors. *Plant Signaling and Behavior*, *1*(3).