The Rise of Open Source Hardware: A Sociotechnical Perspective

A Research Paper submitted to the Department of Engineering and Society

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

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

Jason Ashley
Spring, 2021

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

Signature ______________________________ Date 5/4/2021

Jason Ashley

Approved ______________________________ Date 5/4/2021

Hannah Rogers, Department of Engineering and Society

Abstract

Open source hardware is greatly increasing the number of resources available to hobbyists, academics, and individuals for the development of their own projects. Over the past two decades, a large number of such projects have started, with several of these spawning further developments and projects as a byproduct of their open source nature. Through a case studies analysis of recent projects such as Arduino, open source 3D printers, and the RISC-V ISA, the rapid growth of open source hardware as a development paradigm will be analyzed. These technologies exemplify a movement away from restricted access to tools and technology, shifting towards open design with an emphasis on publishing work freely for outside use and modification. The use of these technologies in both physical and virtual communities are also analyzed to determine what role these groups might play in this and continued developments.

The Rise of Open Source Hardware: A Sociotechnical Perspective

Introduction

Few technologies can claim that within the first few years of their introduction, they revolutionized their fields, becoming ubiquitous in academic, industry, and hobbyist settings. However, there are a range of open source hardware projects that quite comfortably fit that description. Open source hardware is a term used to describe devices or specifications with publicly available designs and explicitly granted permissions for users to work with the design, modifying or implementing it to improve the design or implement it in another project (Herrera, 2020). Available resources often include the bill of materials, schematics, CAD designs, and anything else necessary to replicate the project. Notable projects in the field include the Arduino microcontroller platform, many 3D printers, and the RISC-V instruction set for processors. Each of these examples gained support fairly rapidly and are poised to revolutionize their respective field. In fact, some of these tools, once considered far too expensive outside of industry, are now utilized in grade schools to effectively teach basic programming and mathematical concepts because of their low cost and approachability (Martn-Ramos et al., 2017). Even more surprising, many of these open source tools were created just over a decade ago (Kushner, 2011). The meteoric rise of each of these sets of projects relates to the open source nature of their development. Once expensive and difficult to use hardware and processes have significantly lowered in cost. A wider range of individuals have been able to explore hardware development because of this cost reduction. Beyond the lower cost, local communities created with the express purpose of providing access to these technologies have appeared across the world to help increase access further (Hausberg & Spaeth, 2020). Throughout this paper, each of the presented projects and their effects on individuals and communities will be analyzed through the frameworks of technological momentum and paradigm shift to attempt to further understand the goals and impacts of the projects. More specifically, how the rise of open source hardware designs has impacted the available tools and resources available to general consumers and how this has impacted groups and communities will be analyzed. Hardware based on open source principles rapidly gained popularity because these projects expanded access to previously expensive or restricted tools generally available, changing hardware development in the process.

Development and Rise of Open Source Hardware Projects

Electrical hardware-based projects are often costly, making them difficult for a general consumer to get started with. Several factors contribute to this particular problem. One notable issue is prototyping and simulation tool access for hardware developers (Gupta et al., 2017). Such tools are rarely available to the consumer, and those that are often very costly or lack support for common components, making simulation possibilities very limited. Even without software tools to aid in this process, it would be possible to base the design on details in data sheets. However, data sheets rarely follow a standardized format and are confusing to a more general audience due to the knowledge necessary to understand them (Mancini, 2005). Even then, some data sheets are not public or require entering a non-disclosure agreement with the hardware manufacturer to access. Once a design is developed, the next problem arises in prototyping a design. Ordering parts in small quantities can be difficult. It is expected companies will buy large quantities of parts for developing a product in large quantities, not a consumer buying individual components (Nascimento & Pólvora, 2018). If a microcontroller is desired in a project, options include developer kits or project boards with high costs or building the board from prototyping sheets (Kushner, 2011). Even after prototyping a circuit, developing the printed circuit board or proper housing for the electronics could cost tens of thousands of dollars (Archer, 1988). Thus, hobbyists developing a project, if the resources were even available, would be quite expensive. However, more general consumer-friendly tools, hardware access, and literature are now available.

Open source hardware initiatives are a relatively recent development, with the first proposed definition appearing in 1997 (Barak, n.d.). Relative to similar developments in the software industry, the development of open source in hardware lagged behind by about a decade (Benoit-Barné, 2007; Kushner, 2011). Early work pertained primarily to the development of definitions and desirable traits in a project to be considered open source. Several projects arose slowly over time that would attempt to meet these criteria.

Notable Project: Arduino

One early notable project that appeared in 2005, spun off from a university project, was Arduino (Kushner, 2011). The Arduino project's early goal was to make development of projects using commodity microcontrollers much easier in a number of ways. The developed boards included low cost, low power processors, microcontrollers, for use in a wide variety of situations. They quickly became notable for being used to read sensors, control motors, and performing more simple, single application tasks. The early goals were to build simple and extensible boards that would handle the complexities of power input and output while exposing the pins on the microcontroller that would be useful for interacting with other hardware. The Arduino project also sought to develop relatively easy to use, free software tools to write applications for the microcontroller (Kushner, 2011). These tools would compile, verify compatibility, and upload code to the microcontroller. Over time, the Arduino ecosystem grew to include more sensory tools and applications. Individuals found more interesting use cases which, in many cases, found their way to the mass market. One of the most interesting examples of this was the realization that, with the precise controls of motors and heating elements available via these microcontrollers, plastic could be melted and laid in predefined shapes.

Notable Project: 3D Printers

Proper housing, protection, and other specially designed components are often important in an electronics project. It adds some amount of credibility and rigidity to the project, as well as allowing for it to be more portable or extensible. Thus, some approach for prototyping these components is necessary. One possible material for this purpose is plastic. The material itself can be cheap and its use cases quite flexible, though several problems arise when considering the cost to develop molds based on a model (Archer, 1988). Estimates for even simple designs range into the thousands or tens of thousands of dollars for low quantity injection molded components. Ideally, if the molding process could be avoided, only the material cost would be the major factor for a part. Additive manufacturing, a process by which materials are added layer-by-layer to create a part, would be ideal for this (B. Hughes & Wilson, 2015). Though this process has roots dating back as far as 1890 and some commercial implementations became available in the 1980's, major advancements occurred after 2005 when open source software and hardware designs were developed by the University of Bath in a project called RepRap.

The basis for many cheaper and widely available 3D printers is open source microcontrollers and off-the-shelf hardware for motors, heating elements, and any relevant sensors that might be used in the varying designs (Oltean, 2017). Open designs have been implemented by several manufacturers, sometimes with minor modifications or improvements to the hardware, leading to low costs for these printers. RepRap itself saw many projects spin off of its original specification in an attempt to compete in the market, but one of the most notable examples of this is the open source Prusa series of printers which, despite their popularity, are available at a relatively low cost (B. Hughes & Wilson, 2015; O'Connel, 2021). This printer series has inspired other printers, such as the Ender series, which makes cuts to the original Prusa designs to lower the cost further. This, in turn, has led to a wider adoption of such printers and their designs. With the designs available providing the ability to control and modify all of the hardware, coupled with the rapid growth in the ownership numbers of these 3D printers, communities rapidly developed around these 3D printers. Members in these communities have been able to design, prototype, and manufacture mechanical, electrical, and software modifications to these devices (Oltean, 2017). Using the currently available 3D printers as a basis, more 3D printer design iterations have been designed, prototyped, and implemented in practice (Simons et al., 2019). 3D printers rapidly cut the cost of prototyping and manufacturing parts (Oltean, 2017). A once many thousand dollar endeavor in molding, casting, and curing that might have been required was cut down to simply building a model and running that model into a processing application that would generate positional code. This code could then be used to direct the motors and control the heating element of any 3D printer, reducing the overall costs to just that of the material used in many cases. This completely changed the way that consumers could develop and manufacture their own designs.

Notable Project: RISC-V

One final project that constitutes open source hardware and makes a previously inaccessible element of hardware and hardware design more widely available is RISC-V. RISC-V is an instruction set architecture (ISA) for general purpose processors that may be used in all ranges of computing devices, from microcontrollers to high performance compute clusters (Engadget, 2021). An ISA is simply a set of technical specifications for the range of inputs and the expected output a processor design should produce to be compliant with tools designed to write software for a series of processors. The “language” a processor understands, and how it will respond, are defined by the ISA. The designers of a processor can specify its architecture however they may like, but choosing an established architecture allows code to be more portable and ensures the processor behaves as expected since it may be checked against the ISA’s specifications. Architectures are costly to develop and maintain and, as a result, most processor development up to this point involved a licensing process for the ISA that was both time and cost prohibitive. RISC-V, in being open source, intended to reduce many of these drawbacks of processor design while also giving community members more of a say in the development of the ISA itself. RISC-V-based processors, within just a few years of the ISA being available for development, are already competing with microcontrollers in the low-end, thanks to the ISA promoting simple processor designs that are inherently energy efficient, and could be poised to appear in more consumer and high performance electronics in the near future (Legenvre et al., 2020). Some groups are proposing processors with thousands of small, energy efficient cores to allow for rapid parallel processing, a difficult task with modern, closed architectures (Engadget, 2021). As this continues, more companies may work towards designing and developing their own processors (Gupta et al., 2017). New approaches for handling processor design are one benefit, but an open ISA opens up the possibility for more competition in a space that has been dominated for many years by locked down designs and few major players, while also providing space for more specialized designs that were previously not considered reasonable.

Each of these projects already have or are poised to lower the barrier of entry into their respective fields. Arduino projects made microcontroller development much more accessible, open 3D printers made prototyping and manufacturing products in small quantities attainable to consumers, and RISC-V has the potential to change how processors are developed. As each of these projects developed, their respective communities grew, leading to more continued growth of these projects to date. This interplay between development of open source hardware projects and communities developing to support these projects, as well as using them as a basis for further projects, leads to gathering momentum behind open source hardware and is actively changing what the general public has access to.

Propagation in Communities

Several notable communities have arisen around the rise of open source hardware, contributing to the popularity of certain hardware projects. By using such hardware, these communities are leading to a greater push for companies to offer such hardware, in turn leading to more momentum being put behind these projects. One notable example might be that of makerspaces rising across the world. Makerspaces are centralized locations in some part of a wider community where individuals can meet and work together on their own or group projects (Nascimento & Pólvora, 2018). For many, it is not reasonable to possess a wide range of tools that may be useful in building and prototyping a project. It simply is too much of a time and space investment. However, if a group of individuals worked together to buy the tool, or it was loaned to them, it would encourage the use of those tools. Makerspaces often provide access to

3D printers, laser cutters, electronic measuring probes of many sorts, a wide variety of mechanical tools, and much more (Hausberg & Spaeth, 2020). Individually, building such a collection could be completely untenable, but would be quite reasonable for a community with greater combined resources. These spaces have appeared in many different forms across the world, but, within the United States, there are both private and public entities creating these spaces, most notably public libraries (Good, 2013).

Makerspaces provide a collaboration space for makers, the individuals developing these projects, to learn more about how to effectively utilize the tools. As a result, they grow more accustomed to them and, in turn, might wish to utilize them outside of the makerspace. Notably, this could simply include mechanical tools, such as drills and lathes, but could also include open source 3D printers, Arduino boards or similarly open microcontrollers, and more. The use in these spaces not only increases the overall use of a project, but also will promote the use outside of these spaces due to increased familiarity.

Academia often has also been a hotspot for open source hardware use and development. Higher education benefits from the use of open source hardware in research, as suggested by open 3D printers designed for labs, Arduino as a platform for automated data collection and processing, and safety-critical applications built for industry in academic research labs based on such hardware (Dobrilović et al., 2016; Oltean, 2017; Simons et al., 2019). The use of these technologies in grade school is also a significant boon to students and educators alike (Heradio et al., 2018). A study of students introduced to Arduino in fourth grade found that it was a useful tool for promoting and retaining lessons in mathematics and computer science over more traditional teaching methods, leading to a growing use of this and similar technologies in the classroom as a teaching aid. The study would also suggest that students were significantly more likely to continue learning about Arduino and its applications, which would, in turn, lead to a greater, more widespread use of the platform or other open source hardware projects. Open source hardware is a great fit for this education use case because, when compared to similar methods of teaching these concepts on proprietary platforms, each board is significantly cheaper and the software used for development is often free, easily accessible, and well documented, mitigating the risks and confusion of using such devices.

The Effect of Open Source on the Consumer

The general consumer also reaps many benefits of open source hardware, whether they realize it or not, and these benefits may encourage a continued use of these tools and technologies outside of just the hobbyist and academics fields. One of the biggest advantages is the cost of developing, deploying, and supporting open source hardware projects in medical research and government settings is often lower than that of proprietary solutions (Heikkinen et al., 2020). These reduced costs in technology find its way to the consumer through lower medical costs and taxes. How much savings can be expected? In a case study in Finland on national policy to fund open source hardware over “proprietary tools” in government-funded laboratories across the country, researchers noted that spending on this tooling would be reduced 90%, saving up to approximately 28 million Euros each year (Heikkinen et al., 2020). In just this particular small segment of the government, a significant amount is already saved. The researchers outline an approach to effectively fund the adopted projects so as to ensure continued development while also highlighting the necessary considerations for successful implementation. This would free portions of the budget to be applied elsewhere or allow for reduced taxation. Consumers may also find implementations of commercialized products that were prototyped with or even include open source hardware, allowing these products to be understood (Herrera, 2020). Such open designs may be useful in allowing for faster, cheaper repair of broken devices, such as computers or phones. Further, some hobbyist projects may push the industry further by developing useful ideas.

Perhaps the biggest risk to consumers from open source development is ensuring such designs “include safety, security, and trust” as goals of their design and verification (Mutschler, 2020). Closed systems often derive some of their security from having an unknown design. Ideally, if no one knows how the device actually works, there is no easy way for the security of the device to be compromised. Security can still be compromised in closed designs, though it takes a significant amount of work and understanding for an attacker. With open source designs, there is no way to hide security flaws. If not taken into the consideration during the design process, the consumer is left more at risk to vulnerable designs.

Comparison to Open Source Software

The most relevant discussion to compare these particular developments of these cases to would be similar software projects. Drawing parallels between the two is inevitable. While open source software has experienced unprecedented growth and a stacking of project upon project to build new software with open source licensing, this has been a more difficult task in the hardware industry. The cost of software is often limited to the human time for development and the compute infrastructure behind it due to most necessary tools (beyond a computer) being free or relatively low cost. A similar cost analysis of hardware will find that the physical components, especially if any part is manufactured in low quantities or must be custom made, are quite expensive relative to software development (Gupta et al., 2017). There are often specialized tools required for working with the hardware, too. As such, the explosive growth of software cannot be mirrored in the hardware industry, right? Such an argument discounts the rapid drop in prices recently for many common components and tools, as well as expanded availability outside of channels commonly used to mass order parts, making small quantity, individual purchases much more viable. Further, the need for specialized tools is reduced in more modern projects and specialized tools themselves are often offered at reduced costs for academic and hobbyist use. As with the software industry, taking an open source hardware approach on a project also lends itself to future project modifications and further development by different groups. Communities also crop up around larger hardware projects, just as in software. Though only a few particular hardware projects or categories were discussed in length, these are important milestones in the open source hardware arena that could be comparable to major projects in the software arena, such as Linux, each hardware project similarly lowering the barrier of entry into their respective areas and building massive communities (Benoit-Barné, 2007; Nascimento & Pólvora, 2018). While hardware does lag behind software in terms of openness and ease of access currently, this gap is constantly being bridged by new projects and developments.

Sociotechnical Perspective

The growing momentum behind open source hardware represents a shift from previous closed approaches to hardware development. Coupled with the subsequent communities built around them, this process is providing more motivation for future projects to be more open themselves, while making previously inaccessible tools available to general audiences. Such growth highlights the momentum gathering behind such a development process, essentially defining a subset of technology within the general electrical hardware field (T. Hughes, 1993). However, it also represents a shift in the industry. In Thomas Kuhn's *The Structure of Scientific Revolutions*, an outline for such a process is proposed (Kuhn, 1962). Quite simply, the "paradigm" of the hardware industry prior to the mid-2000's revolved around hardware projects and devices being reserved for a select few groups to develop a product for the general consumer, not something the general consumer could use to develop a product, especially in relation to integrated circuits. Such development was simply too cost prohibitive. Individuals who were trying to get into hardware development as a hobby would likely need to already be established in the industry or academia to get access to the tooling necessary, as well as have the knowledge to find and use components they might need.

Hardware development paradigms started shifting in the early 2000s and likely will continue to move in this current direction as a result of more accessible tooling (Barak, n.d.). Ever since the introduction and popularity of the microcontroller and surface mounted devices in the 1980's, electronics have become increasingly sophisticated and small. This discouraged experimentation without the use of a printed circuit board and proper tooling. With easy to work with microcontroller platforms in the form of development boards such as Arduino, this restriction was almost entirely mitigated (Kushner, 2011). Those involved in these early projects sought to publish their work so others could expand it, hoping to create a viral cycle where one project leads to an advancement of ideas and understanding, which paves the way for the next project which may push this advancement further. While an individual publishing a project could simply be trying to "show off" some interesting work, they also propagate a cycle of more advanced developments leading to even more advanced developments.

Limitations and Counterarguments

The hardware projects chosen are not fully representative of the entire open source hardware movement. They were chosen to highlight progression and the interaction between projects. What is the real state of open source hardware? Is it truly achieving goals of making more widely accessible projects and gathering communities around them? There are many projects that have failed for one reason or another. Some because they fail to reach technical goals, some because they fail to gather a community and support base. However, as a whole, these projects do still benefit those who are involved and could still be used in the future as a basis for other projects since the details of the project should be publicly available. They still further open source hardware and human knowledge as a whole, even if they "fail" by traditional standards.

A further point against this analysis is that many of these projects are still rather niche and might simply represent a fad. The development of projects based on older projects and the communities cropping up around them, as well as the number of companies, academics, and educators that have thrown support into open source hardware projects suggests that the rapid and increasing growth is likely to continue, it is not simply a fad. Even considering the hobbyist settings, makerspaces are still being developed in many major cities, with an ever expanding range of audiences being invited to participate and use the tools and technologies available. For instance, libraries host events for all age groups, artists are welcomed into the space to develop their work or experiment with new artforms, and tools available (as well as sponsorships for public spaces to get these tools) are still increasing in many areas. Though some such spaces do struggle to determine an economic model for staying afloat, others still are quite successful and gather communities of makers in the same area.

Some companies have found great advantages to using open source hardware in development, but have trouble releasing something that embodies those principles (Legenvre et al., 2020). While development using open source hardware is often easy due to ease of access to information about the project, releasing an open source project is considered dangerous as a competitor may easily enter the space by copying these designs. Admittedly, this is a possibility, though there are solutions that could prevent this from being an issue. One would be utilizing dual licensing. The basic premise is to offer a paid, supported version of the project under one license that allows use in commercial settings, but another version that has less or no support and under a more restrictive license preventing the use in a commercial setting, a practice known as selling exceptions (Barak, n.d.). The major problem with this is a legal team may need to be willing to handle violations of these agreements. Many companies, especially those focused on software development, utilize this method to sell software that is still open source. In this way, the company still contributes to open source hardware development while being able to reasonably enter the market competitively. However, some companies allow for copycats. In the case of Arduino, they dealt with many companies developing their own boards that copied the designs of Arduinos, but continued to offer low cost, well-documented hardware and software, never enforcing a policy that would restrict access to the software if using a “counterfeit” device, all while remaining commercially viable (Banzi, 2015). As such, there are options for commercial viability even if the intellectual property of the design of the product is publicly distributed and easily accessible.

Despite the increasing popularity of hardware projects and communities around these projects, development tool availability is not evenly distributed in different parts of the world, country, or even city (Nascimento & Pólvora, 2018). Just because a project is open does not mean everyone has the ability to work with it. A developer working on open source projects should consider this, especially if mass adoption or accessibility is intended. While the barriers to entry have dropped dramatically in the previous twenty years, they are not equal around the world. This is an area where initiatives to alleviate cost and accessibility restrictions to some extent will make significant impacts moving forward. Further, continued development of communities that provide free access will help.

Conclusion

Open source hardware as a development process is a monumental shift from the closed source and restrictive approaches that often appear in the hardware development industry. Of the projects implementing such practices, some of the notable ones include Arduino microcontroller boards, 3D printers, and the RISC-V ISA. While these projects exemplify several different aspects of open source hardware project design and development, notably making details of designs available to and modifiable by the public, they each are changing the way in which hardware is developed, notably in hobbyist and academic settings, creating rippling effects across the industry and dependent fields. It is not surprising these projects have been used as an element or basis for other projects, whether that be in a hobbyist setting, academic work, or even general consumer products, pushing each of these areas further. For this reason, it seems fair to recognize their constantly gaining momentum as such projects are further integrated into scenarios requiring microcontrollers around the world. Further, it is well within reason for the reasons outlined throughout this work that these developments are representative of a much larger shift in hardware development, moving from a more closed, industry-specialized field to one that even early grade school students can get into, with proper guidance. Whole communities are built around developing these projects, and the ideas of those in the community are ingrained into the projects they develop. These spaces are not simply for individuals to build a project for it to disappear without an impact, but to further human knowledge through interactions and publication of their work, strengthening both that individual community and the hardware development community as a whole, wherever they may be located. With few downsides, even to commercialization, it is likely that more projects will open up their designs to the public, continually shifting the world towards more open hardware development.

References

Archer, D. (1988). *Early Cost Estimation of Injection Molded Components* [University of Rhode Island]. https://doi.org/10.23860/thesis-archer-david-1988

Banzi, M. (2015, March 19). *Fighting for Arduino*. Make: DIY Projects and Ideas for Makers. https://makezine.com/2015/03/19/massimo-banzi-fighting-for-arduino/

Barak, S. (n.d.). *How "Open" is Open Source Hardware?* Retrieved April 6, 2021, from https://www.mouser.com/applications/open-open-source-hardware/

Benoit-Barné, C. (2007). Socio-Technical Deliberation about Free and Open Source Software: Accounting for the Status of Artifacts in Public Life. *Quarterly Journal of Speech*, *93*(2), 211–235. https://doi.org/10.1080/00335630701426751

Dobrilović, D., Stojanov, Ž., Čović, Z., Simon, J., & Petrov, N. (2016). Model of data center temperature monitoring system with the use of open source hardware. *2016 IEEE 14th International Symposium on Intelligent Systems and Informatics (SISY)*, 221–226. https://doi.org/10.1109/SISY.2016.7601501

Engadget. (2021, January 11). *RISC-V and the future of open-source computing at CES 2021*. https://www.youtube.com/watch?v=iDvst1yhs7Q

Good, T. (2013, February 6). *Manufacturing Makerspaces*. American Libraries Magazine. https://americanlibrariesmagazine.org/2013/02/06/manufacturing-makerspaces/

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

Hausberg, J. P., & Spaeth, S. (2020). Why makers make what they make: Motivations to contribute to open source hardware development. *R&D Management*, *50*(1), 75–95. https://doi.org/10.1111/radm.12348

Heikkinen, I. T. S., Savin, H., Partanen, J., Seppälä, J., & Pearce, J. M. (2020). Towards national policy for open source hardware research: The case of Finland. *Technological Forecasting and Social Change*, *155*, 119986. https://doi.org/10.1016/j.techfore.2020.119986

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, B., & Wilson, G. (2015). 3D/additive printing manufacturing: A BRIEF HISTORY AND PURCHASING GUIDE. *Technology and Engineering Teacher*, *75*(4), 18–21.

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

Kuhn, T. (1962). The Priority of Paradigms. In *The Structure of Scientific Revolutions* (p. 264). University of Chicago Press.

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

Mancini, R. (2005, April 14). How to read a semiconductor data sheet. *EDN*. https://www.edn.com/how-to-read-a-semiconductor-data-sheet-2/

Martn-Ramos, P., Lopes, M. J., Lima da Silva, M. M., Gomes, P. E. B., Pereira da Silva, P. S., Domingues, J. P. P., & Ramos Silva, M. (2017). First exposure to Arduino through peer-coaching. *Computers in Human Behavior, 76*(C), 51–58. https://doi.org/10.1016/j.chb.2017.07.007

Mutschler, A. S. (2020, January 8). *Open Source Hardware Risks*. Semiconductor Engineering. https://semiengineering.com/open-source-hardware-risks/

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

O'Connel, J. (2021, January 9). *Should I Buy a Prusa i3 Clone or the Original in 2021?* All3DP. https://all3dp.com/2/buy-prusa-clone-original/

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.

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