In the last five years, open source hardware has moved from being a little-known niche activity to become an essential research vehicle and has even established itself in commercial business plans.

Open source hardware is here to stay

By FRANK K. GÜRKAYNAK

When you only look at the files and descriptions needed, at first sight there is very little difference between developing an operating system (software) and an integrated circuit (hardware). Considering how successful and widespread open source software is, it might even be surprising to see that it took open source hardware a quarter of century to become relevant. However, in 2020, open source hardware has firmly established itself and continues to find a broader user base not only in research but in industry as well.

In this article, I will try to highlight the differences between open source hardware and software development and trace its development over time.

Key insights

- Open source hardware is the key ingredient to allow agile co-operation of partners both in academia and industry, which is essential for modern integrated circuit design.
- While fundamentally similar to open source software, hardware and integrated circuit design in particular involve more stakeholders that need to be aligned.
- There is an opportunity to accelerate the acceptance of open source hardware, which will lead to a competitive advantage for early adopters.

Key recommendations

- Establish a European institution to support open source hardware activities.
- Increased support for integrated circuit design activities in Europe is essential.
- More work is needed to clarify legal aspects and allow closer co-operation in Europe.

I am old enough to remember the beginning of the Open Source Software movement. I was able to experience it first-hand as the Free Software Foundation, the GNU Public License, Linux and many others evolved from humble beginnings and essentially changed how software is developed and used. Roughly a quarter of a century later, what I see now is very reminiscent of those times: we have started to change the way we design and use hardware, as open source principles that have become so common in software are being applied to hardware as well.

But before we go any deeper, what exactly do we mean by hardware and how do we differentiate it from software? After all, hardware is a very broad term; brewing beer, building 3D printed medical equipment, designing printed circuit boards as well as implementing integrated circuits could all be seen as hardware design. Since my specialization is in integrated circuit (IC) design, I will concentrate on open source hardware (OSH) for computing hardware. Personally, I like the following definition by Richard Stallman of Free Software Foundation (FSF) fame [1].

"Software is the operational part of a device that can be copied and changed in a computer; hardware is the operational part that can't be."

Still the distinction is not so easy, especially because recent developments would allow a designer to develop hardware using a subset of a conventional language like C, then use a high-level synthesis (HLS) compiler to translate it into a hardware description language and apply it to a field programmable gate array (FPGA), a process that looks deceptively similar to developing a software application using an embedded platform like Raspberry Pi. This *semblance* has led to many discussions and misunderstandings in recent years. Therefore, in this article I will explicitly concentrate on an even more *restrictive aspect* of OSH, where we make use of it to design ICs that can be used to build better computing systems.

Integrated circuit design differs significantly from software development

No matter how you look at it, getting an IC manufactured is quite different from developing software. First of all, an IC is a physical component; it has to be manufactured through a very complicated process that takes weeks in dedicated factories. These so called *fabs* are operated by technology providers, like TSMC, Intel, Samsung, GlobalFoundries and UMC to name just the major players, which have invested billions of dollars in infrastructure to be able to manufacture ICs which today can have tens or even hundreds of billions of components. I tell my students that making a modern 7nm chip is technologically more complex than sending man to the Moon. It may be an exaggeration, but it is not that far off the mark. IC design involves a very substantial one-time (or non-recurring) upfront cost just to get going, and modern large ICs can only justify this cost through large production volume. Of course, there are *cheaper* ICs that are not that complex, but the fact remains that IC design involves working together with a technology provider as well as substantial investment.

It should be no surprise that designing something so complex also involves a wide range of dedicated software, which is collectively known as electronic design automation (EDA) tools. Over the years, following the pace dictated by Moore's law, ICs grew exponentially in complexity, and the tools had to be developed to keep pace with and manage this growth. Today three major companies (Cadence, Synopsys and Mentor (Siemens)) dominate the EDA tool market. Any serious IC design relies on these commercial tools, which come with significant licensing costs.

The key to managing the complexity is modularity and a substantial part of modern IC design relies on pre-designed and validated sub-systems that are made available through third party providers. They can be as simple as standard cell libraries that contain simple Boolean logic gates, I/O drivers, memories, clocking, interconnection solutions, and even processor cores that can then be combined to make a system-on-chip (SoC).

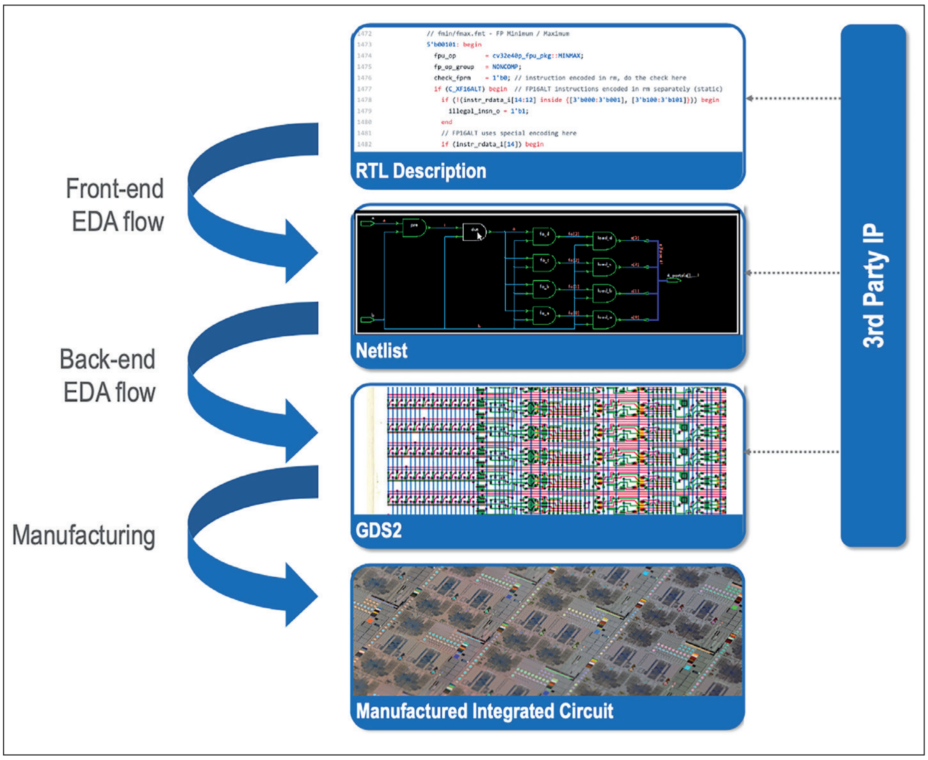


Figure 1: Steps of manufacturing a typical IC, showing interaction of third-party IP, technology and EDA tool providers

As a result, the challenges facing open source hardware are not only the fact that you cannot "copy and change it in a computer" as Stallman stated, but also that there are multiple entities with different commercial interests involved in the process. There are both obstacles and opportunities for open source at all these levels, but the process is taking longer as these different entities (technology, EDA, IP providers) have different goals and concerns. It is important to understand these relationships to develop more sustainable solutions for open source hardware.

There is a second challenge, which is directly related to the complexity of hardware design. There are simply much fewer IC designers than software developers, which reduces the pool of people that can provide open source solutions. This may have contributed to the fact that it took some time for OSH to establish itself but, in the last five years, OSH has started making some serious noise and it will continue to do so.

How does open source factor in?

Even with all the complexities involved, in the end, IC manufacturing relies on a number of computer-generated files that can be stored, transmitted and manipulated electronically. The ultimate description is probably the physical blueprint of the IC, the so-called physical layout captured in a format known as GDS2. This is essentially the only information that the technology provider needs to develop your circuit. Such a file can only be generated with the support of all three entities I have described above. The physical layout discloses information on the capabilities of the technology provider, and is only distributed under strict confidentiality agreements. A physical description that is designed properly needs the support of several EDA tools, and the vendors take issue when such descriptions are made available as it could mean fewer customers will need their tools. This is a particular issue in the case of universities and research institutions that pay only a fraction of the actual licensing costs to the EDA tool vendors. As mentioned, the physical description will most probably feature several different pieces of IP from third parties who may object to their IP being openly distributed as part of a larger design. At the moment, making GDS2 files openly available is still an issue, although in 2020 we saw the first steps to address this problem with the efforts of Google and eFabless in connection with the 130nm Skywater technology [2].

One level below this abstraction is what we call a netlist, a circuit mapped into readily available components of a basic library such as AND, OR gates and Flip-Flops as well as some common blocks like ADCs, PLLs, memory macros etc. Such a netlist has physical properties, you know basically how large the circuit is, how fast it can operate and estimate its power consumption as the functionality is mapped to a technology-specific library. Similar to GDS2, releasing netlists mapped to libraries gives rise to similar problems: third party providers give access to their libraries under strict NDAs and EDA tool companies are not happy to see the output of their tools released.

At least for digital designs, there is an even higher level where we describe the functional behaviour of the circuit using dedicated languages such as SystemVerilog, VHDL, Bluespec or Chisel. In this form that we call RTL, the complete IC description, together with supporting information for verification, is available. However, it needs a front-end design flow to first produce the netlist and then the back-end design flow to generate the GDS2 which can be used to manufacture the integrated circuit. It is exactly these RTL descriptions that fuel current open source hardware success. At this abstraction level, there are still no EDA tools directly involved, and no technology-specific information is disclosed. If you do not look closely, a SystemVerilog description of a processor will look quite similar to the C++ code of a display driver. The difference of course is that good open source RTL descriptions are those that have some pedigree: they have been used as part of actual implementations and working integrated circuits. As an example, there are already forty highly-successful integrated circuits that have been manufactured based on the Parallel Ultra-Low-Power (PULP) platform [3], an extensive collection of optimized implementations of energy-efficient RISC-V based computing systems in SystemVerilog by ETH Zürich and the University of Bologna.

Eventually the success of open source RTL descriptions will also pave the way for open source releases in lower levels of abstraction, as there is nothing that fundamentally limits the distribution of open source GDS2 files once companies embrace open source principles.

How did it all start?

Open source hardware is enjoying a fair amount of the spotlight at the moment, but there were many products with OSH components long before people started to take notice. In the beginning, similar to open source software, most of the contributions came from volunteers and enthusiasts, people that were both passionate and had time on their hands. One of the most well-known early repositories was accessible under Opencores.org (the current www page is not maintained by the same group that originated it). While there were many smaller and simpler projects, as early as 2000, one of their key projects was OpenRISC [4] an open source processor which found serious use in many applications. In fact, the early versions of our PULP platform [3] used customized versions of the OpenRISC. The user group around OpenRISC later ended up founding the Free and Open Source Silicon (FOSSi) Foundation [5] and organized a small meeting called ORConf in 2012. The first three editions attracted only a small group, but I can safely say that OrConf 2015 in Geneva was a key event in OSH history. If you take a look, you will identify most of the key people active in OSH today among the 100+ attendees. The key change was the involvement of major academic groups in these meetings. In addition to our group in ETH Zürich and the University of Bologna, the University of Cambridge, IIT Madras, TU-Munich and UC Berkeley were present at OrConf 2015, held at the premises of CERN. This clearly marked a change in the OSH world, as the initial OSH volunteers were now joined by well-known research centres and universities.

While members of universities were also contributing to early work on open source hardware, involvement at the institutional level allowed longer term projects, and more people to work on them. This also had a direct effect on the output: larger and better supported projects started to became available.

At around the same time RISC-V started to have a noticeable impact. Developed by UC Berkeley, on its own RISC-V is not directly OSH. But the instruction set architecture (ISA) provided a contract between the software and the underlying hardware that was fresh, clean and was made openly available. A well accepted ISA is impor-

Figure 2: Group photo of OrConf2015 in CERN, Geneva (from https://orconf.org/2015/). One of the key milestones of OSH.

tant to allow both the supporting software (compilers, libraries, operating systems) and hardware to be developed independently. It is important to note that RISC-V was not the first open ISA. Open SPARC and the aforementioned OpenRISC were available long before RISC-V but, while the other two still continue to exist, RISC-V has enjoyed far more success. A large part of this success lies in the work put in by the RISC-V Foundation [6], which nurtured the ISA and was able to attract many high-profile companies to support the effort.

If OrCONF 2015 in Geneva was the coming out party of larger universities joining the OSH movement, 2019 (and perhaps the RISC-V Workshop we organized in Zurich) marked the time when OSH received serious industrial backing. Not one, but two non-profit organizations, the Chips Alliance [7] (Google, Western Digital, SiFive) and OpenHW group [8] (NXP, Thales, Silicon Labs) as well as the OpenTitan [9] initiative by Google and LowRISC were announced in 2019. All three of these efforts committed significant resources to develop, curate and improve open source hardware. Today, practically all major companies have significant involvement in these organizations and many see real benefits from using OSH.

How does the public, academia and industry benefit from open source hardware?

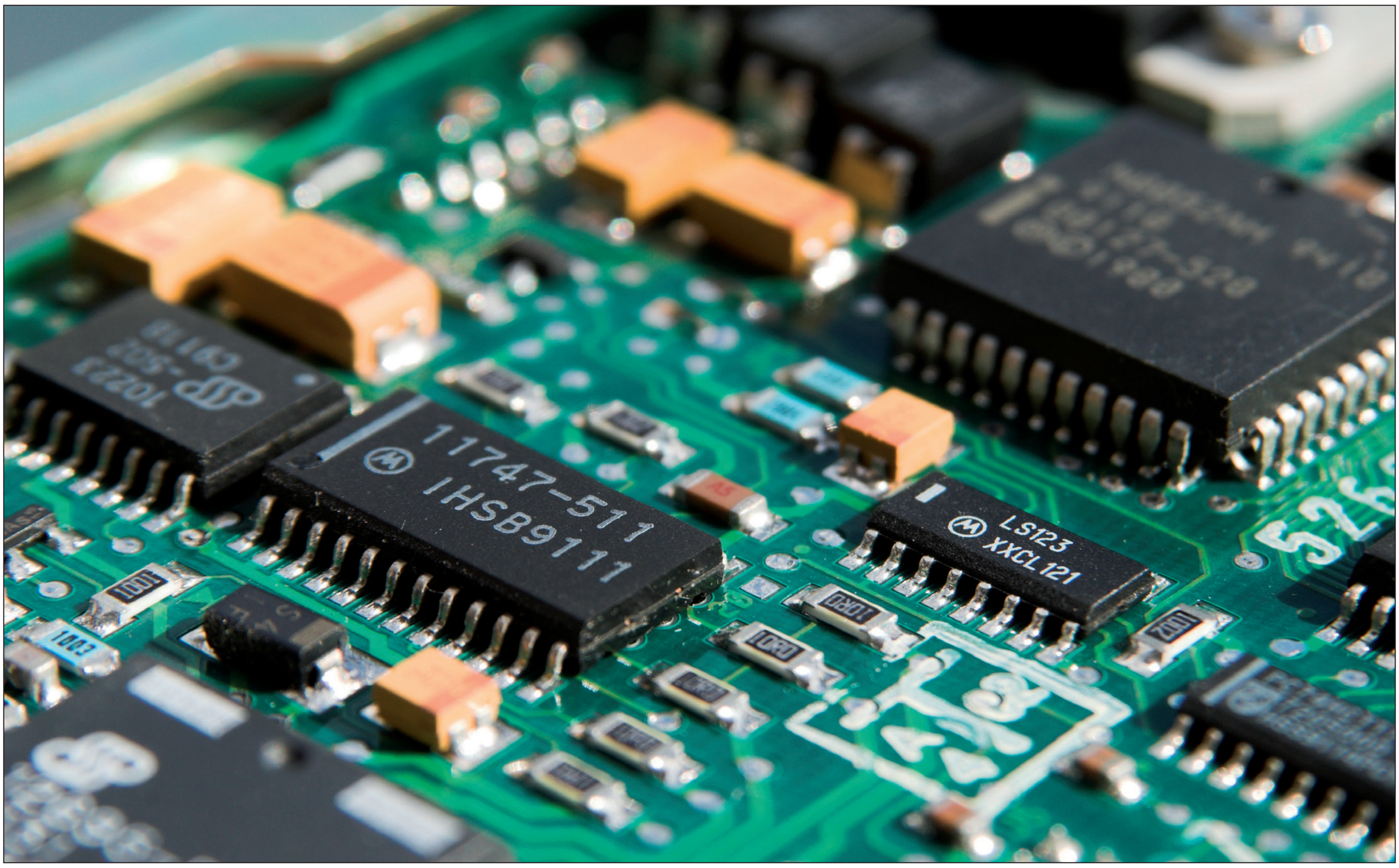
In all reality, if OSH did not have real benefits, it would not survive on its principles alone. In my opinion, the main reason why it took so long for OSH to have an impact has more to do with Moore's law and the associated growth in complexity in integrated circuits. Twenty-five years ago, the development of an efficient 32-bit microprocessor was the pinnacle of integrated circuit design. The Intel Pentium MMX from this era had the complexity of about 1 million gates and could be clocked at 233 MHz. Today Masters students at ETH Zürich where I work, regularly design integrated circuits that contain several processors. What was once considered special has now become a commodity, a building-block to make larger and more capable systems. Having such basic building blocks freely available as open source has been a very attractive proposition for many companies. In fact, Greenwaves, a startup from Grenoble, was able to base about 90% of their GAP8 (and follow-up GAP9) [10] IoT processors on OSH that the PULP project made available. This allowed the company to concentrate their efforts on differentiating their product by adding specialized accelerators and modifications. Especially for SMEs, the cost saving associated with procuring a proven processor infrastructure and peripherals can be substantial, and opens a faster path for innovation.

For those of us in the research field, the main enabler and driver has been the ability to co-operate with both industry and other academic partners freely. Modern integrated circuits have become so large and complex that creating innovations in this field is virtually impossible if you need to do everything on your own. The ability to share and co-operate with partners is essential if projects are to have impact. Without lengthy discussions on NDAs and access regulations, we can quickly get started on projects at a scale that is relevant and concentrate on innovations instead of spending inordinate amounts of time on what is essentially commodity infrastructure. Simply put, the ability to share our work with different partners has become part of our research infrastructure, which is greatly simplified under an open source model.

We have also come to realize that we get more co-operation opportunities and more dissemination as a direct result of our open source activities. There are also some additional benefits that get often overlooked. The ability to have fair and well-controlled benchmarks in integrated circuit design and the widespread use of OSH designs in training and teaching activities even by commercial EDA companies are just two examples.

In a time when people are increasingly worried about privacy, and popular attacks like Spectre and Meltdown have grabbed the headlines, having access to the complete inner workings of the hardware for public scrutiny presents yet another opportunity for OSH. Just having access to the RTL description is certainly not sufficient, but removes an important hurdle for an auditable system. In a time when digital systems control large aspects of our life, getting more secure and auditable solutions will be

Figure 3: Major industry backed open source initiatives and their backers as of November 2020.



one of the most important contributions of open source hardware to society. There are already several well-funded projects, like the Posh Open Source Hardware [11] from DARPA, to address these concerns.

A few words on open source licences for hardware

As OSH becomes increasingly popular, the licensing aspects have also come under scrutiny. I had to learn it the hard way that working with open source licensing is a bit more involved than simply keeping all the rights for yourself. Our first OSH release was delayed by several months until we could understand and sort out all the issues.

The first point is to understand that there are fundamentally two separate families of open source licence. What we call permissive licences (Apache, MIT, BSD) basically allow your users to take what you have provided, use it, modify it and even sell it. They do not even have to tell you what they are doing with it. Most annoyingly, they can take what you have started and, when they make something better out of it, they do not have to share it with anyone else. Particularly at the beginning of the open source movement, this was seen as a major problem, and so called reciprocal licences were developed (GPL, LGPL). This second family of licence asks the user to make systems built using what they have received openly available under the very same licence.

Traditional open source contributors and volunteers prefer and advocate the reciprocal licences as much as possible. On the other hand, industrial users have to be careful to be able to protect the extent that open source components they use penetrate the overall ownership and rights of their products and therefore will generally only work with permissively licensed OSH. What makes everything more difficult is that solutions in software that set practical boundaries on how far the influence of the reciprocal licence will reach (like the Lesser GPL license, LGPL) do not translate well to OSH. Recent efforts by CERN on the Open Hardware License [12] have been an attempt to bring more clarity. The fact remains that, until they are challenged in court, we will not know for sure how well licences used for OSH will hold up. For example, is Apache good enough as a permissive OSH licensee or do you need the additional clarification that the Solderpad [13] licence brings on top of Apache? Will companies be more susceptible to patent lawsuits if they use OSH, or will the combined strength of the industrial interest groups like RISC-V International, OpenHW Group and Chips Alliance that support these OSH be sufficient to deter such suits? There is still a lot to learn in the coming years.

What can Europe do to lead in this area?

There is no denying that Europe has a keen interest in OSH activities and, as a result, it has become one of the most important players in the OSH movement, the recent move of RISC-V international to Switzerland being just another example of this trend.

In simple terms, OSH allows more people to work and innovate on IC design, and it is important to support these efforts and encourage the re-use of common building blocks to develop designs of much higher complexity and significance. A key issue is supporting OSH at lower abstraction levels, in addition to RTL descriptions, as well as paving the way to distribute ready-to-manufacture GDS2 files as well. This is especially important for analog components that need to be designed specifically for a given technology as well as providing OSH components that have already been manufactured and proven to work as advertised. As described earlier, this aspect is still facing some challenges from stakeholders (technology providers, EDA tool companies, third party IP providers) that are comfortable in their established practices and realize that additional effort will be needed on their part for a change. Europe is home to several companies that are involved as stakeholders, and active encouragement to support OSH activities will accelerate these changes.

An important arena in which OSH is expected to play a key role is the realization that technologies that everyday life increasingly relies on (computers, data centres, communication infrastructure) are being developed, manufactured and also controlled by a very limited number of companies (and countries). Recent efforts of the European Processor Initiative [14], which has significant contributions from OSH, is part of the push for digital sovereignty for Europe. It is clear that individually, the member states and their research centres, universities and companies will have a hard time competing with established powerhouses in IC design unless they are able to pool their resources effectively and work in close co-operation. OSH can be an effective tool to facilitate just such a co-operation, but more work is needed to establish it within Europe.

The Europractice service [15] has been the key enabler facilitating access to both EDA tools and IC manufacturing services for SMEs and academia for more than two decades. An obvious step would be to bolster and extend these services in such a way as to allow members to be more active in OSH. Such a European institution (Europractice-OSH if you will) could take a leading role in opening discussions with stakeholders and creating an environment that not only provides an infrastructure for sharing OSH but also helps to establish legal framework and clarify licensing discussions around OSH usage for member states.

When it comes to designing high-performance ICs, especially for computing hardware, it is very important to realize that these are very costly projects, due not only to personnel costs but also to those associated with manufacturing. These include EDA tools and the necessary third-party IP, even when significant elements of it are being realized using OSH. If Europe wants to take a role of leadership in OSH, it also needs to support activities for supporting the manufacturing of designed ICs. Most of the current funding schemes are not compatible with the costs of modern IC manufacturing. The aforementioned institution could also serve in this capacity, as an interface to negotiate third-party IP for use in European-sponsored projects, educate European decision makers on the costs and feasibility of such projects and provide the necessary technical and legal framework to allow project partners access to and to share jointly developed projects. Note that Europractice already provides excellent service to fabrication (through Europractice-IC) and EDA tools (through Europractice Software Service), which represent two of the three stakeholders identified; expanding this service to allow design enablement through OSH principles seems like a logical next step.

It is important to note that commercial entities (big or small) will equally benefit from a more dynamic environment enabled by a wider influx of OSH in IC design. The entry barriers to SMEs designing their own ICs will be reduced, resulting in more designs that will be manufactured, requiring additional EDA licences, and increased need for both open source and commercial third-party components as well as services and businesses around these opportunities. While there is a good chance that these changes will happen organically over time in line with market demands, there are opportunities to accelerate this process within Europe, to allow the Union to move further ahead through government support to improve the acceptance of open source hardware among all stakeholders.

Conclusion

Within the last five years OSH has already made a significant impact, which is only going to increase as more and more stakeholders realize that the opportunities that it presents outweigh the concerns they have over quality and potential loss of revenue. This is not to suggest that all future ICs will be 100% open source but, as the open source software example has shown us, for components that everyone needs (think of GCC, Linux), taking advantage of the collective experience and effort of an open source approach allows everyone to benefit from solid building blocks and concentrate their energy into further innovation.

References

[1] Richard M. Stallman, "Free Hardware and Free Hardware Designs", https://www.gnu.org/philosophy/free-hardware-designs.en.html
[2] Github, "SkyWater Open Source PDK", https://github.com/google/skywater-pdk
[3] PULP Platform, https://pulp-platform.org
[4] OpenRISC, https://openrisc.io/
[5] The Free and Open Source Silicon Foundation, https://fossi-foundation.org
[6] RISC-V, https://riscv.org
[7] Chips Alliance, https://chips-alliance.org
[8] OpenHW Group, https://openhwgroup.org
[9] OpenTitan, https://opentitan.org/
[10] Greenwaves Technologies, https://greenwaves-technologies.com/gap8_gap9/
[11] DARPA, "Posh Open Source Hardware", https://www.darpa.mil/program/posh-open-source-hardware
[12] CERN Open Hardware Licence, https://ohwr.org/cernohl
[13] Solderpad Hardware License, http://solderpad.org/
[14] European Processor Initiative, https://www.european-processor-initiative.eu/
[15] EuroPractice, http://www.europractice.stfc.ac.uk/europractice_com/

Frank K. Gürkaynak is a Senior Scientist in the Digital Circuits of Systems Group of ETH Zürich, in Switzerland. He has been part of the open source PULP platform effort since its beginning in 2013.

This document is part of the HiPEAC Vision available at hipeac.net/vision.

This is release v.1, January 2021.

Cite as: F. Gürkaynak. Open source hardware is here to stay. In M. Duranton et al., editors, HiPEAC Vision 2021, pages 192-197, Jan 2021.

The HiPEAC project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 871174.