2024

Open Source Hardware Guidelines

No. 2: OSH Licensing

Prof. Dr. Dana Beldiman
Dr. Fabian Flüchter
Simon Tannen

Version December 31, 2024

This Guideline has been prepared by the Bucerius Law School Center for transnational IP, Media and Technology Law and Policy (Hamburg) as part of the project “Fab City Hamburg”. Its aim is to provide basic guidance for technical personnel relating to the use of licenses in the context of open source hardware (OSH) projects. This research is funded by dtec.bw – Digitalization and Technology Research Center of the Bundeswehr which we gratefully acknowledge. dtec.bw is funded by the European Union – NextGenerationEU.

Disclaimer: The content of this Guideline is academic in nature and is not intended to constitute or substitute for legal advice. The considerations discussed are subject to the specifics of each individual case, and legal assessments may vary depending on the laws of different jurisdictions.

Content

A. Introduction: The relevance of licensing for OSH........1
B. General remarks regarding OSH licensing........2
I. The Open License Explained........2
II. The OSH License explained........2
1. Hardware with Patent Protection........4
2. Hardware without patent protection........5
III. OSH-licenses: Characteristics, Types and Structure........6
IV. The IP-left effect........6
V. Types of licenses........8
1. Strongly reciprocal / strong IP-left licenses........8
2. Weakly reciprocal / weak IP-left licenses........8
3. Non-reciprocal / permissive licenses........9
VI. Differences between OSH and OSS licenses........10
VII. Typical structure of OSH-licenses........11
1. Preamble........11
2. Definitions........11
3. Scope/Applicability........12
4. Rights and Obligations........12
5. Disclaimer and Liability........13
6. Termination........14
VIII. Why OSH standard licenses should not be modified........14
C. Selected OSH-licenses in detail........16
I. CERN OHL 1.2........16
II. CERN 2.0 licenses........18
1. General Remarks regarding all CERN OHL 2.0 licenses........18
2. CERN OHL 2.0 – s........21
3. CERN OHL 2.0 – w........22
4. CERN OHL 2.0 – p........23
5. How to implement the CERN OHL 2.0 licenses........24
III. TAPR OHL........24
IV. Apache 2.0 and Solderpad 2.1........26
V. GPL 3.0........28
VI. LGPL........29
VII. Mozilla 2.0........29
VIII. CC-licenses........29

D. License compatibility 31
E. Multi-Licensing and Version-Lagging...33
F. Strategic Implications for FabLabs...34
G. Overview – classification of licenses 35
H. FAQ – OSH licensing..36

A. Introduction: The relevance of licensing for OSH

Proper licensing is essential for Open Source Hardware (OSH) as it ensures the OSH project remains open for future users. OSH licensing is complicated by the fact that, unlike Open Source Software (OSS), many components of OSH do not automatically come with exclusive rights and, therefore, do not necessarily support a license. Without a formal license, both the original OSH and any modifications could potentially be claimed by others. This could, in extreme cases, even result in the original creators being barred from using their own project (see “Freedom to Operate” in Guideline No. 1 on IP and OSH). Against this background, this guideline focuses on how to effectively license every aspect of your OSH project to maintain its open-source status.

Please note that this guideline does not cover all possible scenarios for licensing OSH. We merely aim to give an overview of the most important licensing aspects. This guideline is primarily intended for licensing OSH in FabLabs. It does not constitute legal advice.

B. General remarks regarding OSH licensing

When participating in an Open Source Hardware (OSH) project, it is important that you focus on two key intellectual property (IP)-related aspects: First, do you own the necessary rights to proceed with your project without infringing others' rights? Second, how can you effectively grant users of your hardware the requisite rights under conditions of your choice? For guidance on the first question, please refer to Guideline No. 1 on IP Rights and OSH. The second question will be addressed below.

I. The Open License Explained

This section will discuss how to effectively license your OSH project so that it maintains openness without compromising your own rights to the project. To make things simple, we will start out with an OSS license and then move on to the OSH license.

An OSS license is a binding contract between an author and a user of a software component. The “openness” results from two crucial provisions. The first consists of the main functions of the license, which are to give users the right to use, modify and redistribute the software component and to provide the underlying source code needed to allow them to do so.[1] The second crucial license provision is to require that any further re-distribution of code or derivatives to users downstream, must take place under the same license conditions as the license under which the code was received by the user (aka the “viral” provision).

Otherwise stated, a license gives users permission to do which otherwise would be prohibited by copyright laws. Without a license, the acts of using (copying), modifying and redistributing software would infringe the author’s copyright.

II. The OSH License explained

OSH operates on a principle similar to OSS, with regard to open use, modification and sharing. It differs in control over downstream redistribution, because hardware is governed by different intellectual property rules than software.

[1] “*The license grants people rights that they wouldn’t otherwise have and makes clear the conditions under which they can exercise those rights*”, see OSHWA FAQ under “*What is a license and how do they apply to hardware?*”, see https://www.oshwa.org/faq/ (last accessed 31 December 2024).

An OSH project usually involves design documentation and hardware, whereby documentation is usually in digital form, and hardware in physical tangible form.

> **Example:** Plans on how to build a RepRap printer, for instance, are available online.[2] But to build and operate any RepRap printer, you need to assemble the physical materials needed to build it.

Documentation follows the rules of OSS above and is most often licensed under a separate OSS license which grants the right to use, modify and retransmit the documentation.[3]

To understand how hardware is licensed, a bit of knowledge in IP law is required. Because hardware is governed by different IP rules than software, the creator/inventor’s ability to exercise control over downstream transfers is different. Here is why:

- Software[4] is considered a creative work, which falls in the realm of copyright law. Copyright law attaches to “creative” works automatically, as soon as they are created and fixed. In other words, a programmer cannot avoid automatically becoming an author, once software is written. Becoming an author gives the programmer the exclusive rights to copy, modify and distribute the work, and therefore the power to allow (or not) others to perform these acts, i.e., to condition the use of the software.

- Hardware, on the other hand, falls in the realm of patent law, which protects works that are useful (also described as functional or critical to the operations of the hardware).[5] Patent law does not attach automatically to a work or invention. Instead, patent rights must be procured by registration. Absent patent protection, hardware inventors do not have the ability to place conditions on the use of hardware. Therefore, the second crucial requirement for imposing “openness” on the hardware license cannot be met. This requires a different treatment of open hardware and standardized specialized OSH licenses are available to try to address most practical problems and intellectual property risks. However,

[2] See https://reprap.org/wiki/RepRap_Options with corresponding links (last accessed 31 December 2024). Note that most of the building instructions seem to be hosted on GitHub.

[3] Please note that OSS licenses may not always be suitable to license Documentation for OSH. The reason is that OSS licenses are designed to license software, which in most cases is not part of the Documentation for OSH. We will discuss this point in further detail below.

[4] We use the term software to include all design documentation, firmware, designs, graphs, documentation, writings, etc. i.e., generally all digital material that contributes to the making of open hardware.

[5] Copyright law does generally not apply to any aspect of hardware use, production or distribution. Although it is conceivable that a completed hardware product displays ornamental features which could potentially be protected by copyright.

it is helpful to understand the possible problems. The following will look at the two likely scenarios: hardware with and without patent protection.

1. Hardware with Patent Protection

The benefit of patent protection for hardware is that it allows to control downstream use of the protected hardware.

Licensing hardware with patent protection works essentially the same way as licensing software. The patent rights allow the licensor to put conditions on the use of the protected hardware. The (OSH) license gives users permission to do what otherwise would be prohibited by patent laws. The licensor may require that any further re-distribution of the hardware or derivatives to users downstream, must take place under the same license conditions as the license under which the hardware was received by the user (aka the “viral” provision).

But there are disadvantages. One downside of patent protection is the cost and time associated with registration of the patent. For instance, filing a European patent on average costs 6,800 EUR[6] and the grant procedure takes about three to five years.[7] This may be incompatible with the development timeline of an OSH project, where information on the project design is usually made available early in the design process to receive as much feedback as possible and to enable design collaboration.

In addition, once a patent is registered, it must be maintained. In contrast to copyright, this requires the patent holder to pay maintenance fees to the patent office and to defend the patent against attacks (i.e. revocation). Failure to maintain a patent means to lose all patent rights. This leads to loss of control over the use of the licensed hardware. Third parties could then make the hardware without permission, assuming they can do it without infringing the copyright on the underlying documentation.

Furthermore, from a practical view, owning a patent may require you to enforce it against parties that do not comply with your OSH license. But patent litigation is very costly. In Germany for instance, a first-instance infringement action may cost between 75,000 EUR and

[6] See EPO FAQs under https://www.epo.org/en/service-support/faq/applying-patent/fees-and-costs/how-much-does-european-patent-cost (last accessed 31 December 2024). Please note that this is just the fee of the European Patent Office (EPO). There may be additional costs (especially attorneys’ fees) for defending the patent application in case of an opposition.

[7] See EPO FAQs under https://www.epo.org/en/service-support/faq/applying-patent/procedure/how-long-does-grant-procedure-take (last accessed 31 December 2024).

375,000 EUR[8] Also, you would have to watch the market for potential infringers and monitor downstream use of the protected hardware. Otherwise, your patent rights would not make any difference.

Finally, registering a patent does not mean you cannot infringe other parties' rights, e.g. other patent rights. This is because the contours of patents are generally imprecise. Patents are open to interpretation and may thus not cover all the hardware you think. You may thus think you have protection, but in fact your use may infringe another's rights. In addition, the more successful or monetarily rewarding your project is, the more likely a third-party challenge will be.

2. Hardware without patent protection

Having no patent protection to the hardware has the benefit of making project administration easier. You do not have to worry about patent registration, maintenance and enforcement (see above).

But there are downsides, too. If used in an OSH project, you have no control over downstream distribution of your hardware and cannot prevent the appropriation of derivative works without a patent. For example, if a third party makes an improvement to your hardware, where the improvement itself is eligible for patent protection, it may register a patent on this improvement. In the worst-case scenario, the third party may thus lock you out of your own hardware. This means that, because of the patent rights in the derivative, you may no longer be allowed to use your hardware. In addition, there is also the risk of infringement of third-party rights (see above).

From a practical view, it is probably best to ensure that your software and hardware travel together. You thus need two licenses: One OSS and one OSH. To ensure wide accessibility of your documentation and any derivates, you should use a strong copyleft open source license.

As a practical matter, most OSH hardware is not patent protected, because it is developed by OSH developers who generally tend to follow an openness ideology, or because it is not

[8] See *Taylor Wessing*, German Patent Cases: Statistics, p. 9, note 11. Available under https://www.taylorwessing.com/-/media/taylor-wessing/files/germany/2021/04/tw_2020_german-patent-cases---statistics.pdf (last accessed 31 December 2024).

patentable (no novelty or inventive step). But keep in mind that the hardware may be subject to utility model and design law (see Guideline No. 1 on IP Rights and OSH).

One possible strategy to avoid third parties from registering patent rights to a piece of OSH is called “defensive publishing”. It means publishing information on an OSH project as early as possible to destroy its novelty and thus make it no longer patentable. For more information on defensive publishing, see Guideline No. 1 on IP Rights and OSH.

III. OSH-licenses: Characteristics, Types and Structure

When it comes to choosing a suitable license to make hardware open source, there is no one-stop-shop solution. What license to choose depends on how much you want to control downstream use of the licensed piece of OSH. You therefore have to ask yourself: What should others be able to do with my ideas? Should any contributions or changes also be Open Source? Should they be used for profit or only non-profit?

The question of what license is chosen is also important for licensees. Licensees should not assume that hardware made available under OSH-licenses can be used free of any obligations. For instance, OSH-licenses generally require the licensee to mention the original creator’s name when using or republishing the licensed material or any derivatives. Also, it is usually required to also publish the license as well as the licensing conditions. Examples of how this is correctly done can be found below in the section C. dealing with particular licenses.

The crucial question for choosing a suitable license is how strongly reciprocal the license in question is. In other words: What does the freedom to use, study, distribute and modify require the licensee to do in return and which parts of a derivative work are affected by those obligations?

Depending on how source material must be made available and how derivative works are affected by a license, a distinction can be made between strong and weak IP-left as well as permissive licenses. You may think of these types as different “flavors” of OSH-licenses. The transition between these types, however, can be fluid.

IV. The IP-left effect

Regarding the terms of OSH-licenses, one of their most important aspects to be observed by licensees is the “IP-left” effect. With the help of this effect, the IP-owner can use her exclusive control of protected material to help keep it open-source in accordance with no. 4 of the OSHWA open-source-hardware definition ("derived works").[9]

Originally, the term "IP-left" derives from the term "copyleft". It is a play on words intended to illustrate the "reverse use" of exclusive IP rights in open-source licenses, in the sense that the rights are not used in the traditional "exclusionary" way, i. e. to prevent others from use, but, instead, in an "open" way, to make the rights available to as many users as possible.

Traditionally, IP rights are used to create (temporary) monopolies. This means that with the help of IP rights such as copyright or patents, any owner can exclude everyone else from using, distributing and often even improving on the protected material. Broadly speaking, the owner of an IP right is in control of how the protected material may be used (For more information on IP rights, see Guideline No. 1 on IP Rights and OSH).

In other words, the owner's exclusive right to copy, publish, distribute and edit a work, for instance in the case of copyright, is used not to exclude, but to allow copying, publishing, distributing and editing by any interested party, as long as users adhere to the applicable licensing conditions. This usually entails that every contribution to or change of the original source material is also made open source. For this reason, the term "viral effect" is often used in connection with IP-left licenses: It works like a virus that impacts all enhancements and derivatives.

> **Example:** The licensed piece of OSH is a device for testing the quality of drinking water. You, the licensee, modify the original design by adding sensors to make it more accurate. Under the IP-left effect, you are required to make your modified design also available under open-source conditions. In this way, other users may benefit from your design and may further improve on it.

[9] Available at https://www.oshwa.org/definition/ (last accessed August 03, 2022).

V. Types of licenses

The following will describe the different “flavors” of OSH licenses.

1. Strongly reciprocal / strong IP-left licenses

Strongly reciprocal licenses (or strong IP-left licenses), grant the licensee the freedom to use, study, distribute and modify the licensed material in return for preserving the same freedoms in all derivative works. Hence, what has been released once under a strongly reciprocal open-source license (usually) remains accessible with the same freedoms downstream, including all its derivatives. Think of it as a snowball effect (or viral effect).

“Strongly reciprocal” means in this context, that when making a product, you, as licensee, have to provide each recipient with a copy of all information necessary to make the product or parts of it themselves. This does not only include information on how to assemble the product, but also information on where to get the parts and tools and - if applicable – how to assemble each of them. In particular, software used to make the product must usually be made available under the same or under an OSS-license. This ensures that, if necessary, every recipient will be able to repair the product by herself.

> **Example:** The licensed piece of OSH is a clothes drying rack. A strongly reciprocal license will probably not only require you, as licensee, to publish the schematics, but also information on what tools are needed, where to get or even build them and how to cut the wood, etc.

Popular strongly reciprocal licenses include CERN OHL 2.0 – s for hardware, GPLv3 for software or CC BY-SA 4.0 for other copyright-protected works.

2. Weakly reciprocal / weak IP-left licenses

Weakly reciprocal (or weak IP-left licenses), similar to strongly IP-left licenses, grant the licensee the freedom to use, study, distribute and modify the licensed material in return for preserving the same freedoms in all derivative works. Hence, what has been released once under a weakly reciprocal open-source license (usually) remains accessible with the same freedoms downstream, including all its derivatives. As in strongly IP-left licenses, think of it as a snowball effect (or viral effect).

In contrast to strong IP-left licenses, however, the scope of the material licensees must publish in compliance with the license is less comprehensive. “Weakly reciprocal” means that when making a product, you, as licensee, have to provide each recipient with a copy of all information necessary to make the product, or parts of it, themselves. This does not only include information on how to assemble the product, but also information on where to get the parts and tools. In contrast to strongly reciprocal licenses, some parts, such as software used to make the product, may be proprietary. A compatible open-source-license may not be necessary.

> **Example:** The licensed piece of OSH is a 3D-printer. A weakly reciprocal license will probably not only require you, as licensee, to publish the schematics, but also information on what tools are needed, where to get or even build them and how to cut the wood, etc. But it may not include the software that is operating the printer. This software may even be proprietary.

Well known examples for such licenses include CERN OHL 2.0 – w for hardware.

3. Non-reciprocal / permissive licenses

Non-reciprocal (or permissive licenses) do not require reciprocal, i. e. cooperative behavior. Commercial use of OSH products is possible under permissive licenses. Therefore, the licensee is granted the same freedoms as in strong or weak IP-left licenses but is not required to grant those freedoms for its own derivative work. This means, that for instance, commercial use of OSH is possible under permissive licenses. This means that you, as licensee, can make a product using material licensed under a permissive license, but do not have to make it open source yourself. For this reason, permissive licenses are quite short, but still make sure that the licensor cannot unilaterally withdraw her permission. Permissive licenses have no viral effect.

Permissive licenses may still require you, as licensee, to name the original author and keep notices (e. g. copyright, acknowledgement or trademark notices) intact.

Popular permissive licenses include CERN OHL 2.0 – p for hardware, MIT or Apache 2.0 for software.

VI. Differences between OSH and OSS licenses

For a long time, open-source innovation has mostly been successful in the software field, i.e. in form of OSS. OSS is an intangible digital product that consists solely of code. This is also the reason, why licensing OSH differs from licensing OSS. You should not assume that applying a standard OSS-license to hardware will work. Here is why.

Copyright arises automatically upon creation of a work. Because source code is a work subject to copyright, the creator of source code automatically has the power to license its exclusive right (copyright) in the source code by means of an OSS-license. This makes open-source licensing easy. A programmer of OSS can apply a copyleft or a permissive license to her software, and thereby entitle users to use her software.

The same mechanism does not necessarily apply to OSH. This is due to the fact that OSH consists of tangible hardware or a combination of hardware and software. Making an OSH product therefore not only requires documentation, but also tools and materials, as well as possibly, software.

Given that fact that OSH combines both tangible and intangibles, licensing a piece of OSH consists of two parts: (1) licensing the documentation that instructs the user on how to build the OSH-product (intangibles) and (2) licensing the making of the product itself (tangible). Whereas the documentation (i. e. texts, drawings, code, etc.) may or may not be copyrightable, this is not necessarily the case when it comes to the product itself. For example: It is questionable if utilitarian products, such as simple spare parts, reach a certain level of intellectual creation that is copyrightable, even if they are 3D-printed. In addition, the issue whether the transfer of an OSH-design from a digital file that is copyright protected to a tangible product in the real world through 3D-printing would be seen as a copyright infringement, has to our knowledge until now not been raised in German courts. Consequently, when it comes to OSH, the product itself may not serve as a trigger for copyright. In addition, hardware can be subject to other IP rights such as patents, utility patents, designs and trade secrets. These rights generally arise upon registration. For an OSH license to be effective, the inventor must register its rights.

As a result of the structural differences between OSS and OSH, OSS-licenses generally do not sufficiently cover the rights in (i) the documentation as well as (ii) the physical hardware that may be created based on the documentation. A separate license for licensing the making of the hardware is thus required.

VII. Typical structure of OSH-licenses

In this section, we will take a closer look at the general structure of OSH-licenses.

OSH-licenses generally consist of the following parts:

- Preamble
- Definitions
- Scope / Applicability
- Rights and Obligations
- Disclaimer and Liability
- Termination

Individual OSH-licenses may differ from each other and may contain different terms.

1. Preamble

The Preamble says what the general purpose of a license is.

> **Example:** According to the Preamble of the CERN 1.2 license, "CERN wishes to provide a tool to foster collaboration and sharing among hardware designers" by way of the license.

In case of OSH-licenses, the purpose usually is to promote free access and distribution of ideas and pieces of OSH. While a Preamble may be helpful to understand a license, it is not formally necessary to have a Preamble for the license to be legally binding.

> **Example:** The Apache-license 2.0 for instance, does not have a Preamble.

2. Definitions

This is a license's own dictionary.

Terms that are important to the license most often have a special meaning. To avoid repetitions and make a license easier to understand, this meaning is defined in the "Definitions" section of a license. Defined terms usually appear with capital initial letters throughout the license.

Example: The well-known Creative Commons License BY SA 4.0 defines "*Licensed Material*" as "*artistic or literary work, database, or other material to which the Licensor applied this Public License*".

Sometimes, Definitions can be more difficult to grasp, especially when they refer to each other or overlap.

Example: In the CERN OHL 2.0 – s, the term "Source" – which is similar to the "Licensed Material" in the CC BY-SA 4.0 - is defined as "*information such as design materials or digital code which can be applied to Make or test a Product or to prepare a Product for use, Conveyance or sale, regardless of its medium or how it is expressed.*" In order to fully understand the meaning of "*Source*", it is therefore necessary to also understand the definitions of "*Make*", "*Product*" and "*Conveyance*".

3. Scope/Applicability

This section describes the place and time to which the license applies.

Usually, OSH-licenses include the grant of a perpetual, worldwide license. This means that the license is unlimited in time (perpetual) and works everywhere in the world (worldwide). Sometimes, licenses also mention the type of use to which a license applies.

Example: For instance, the CERN 2.0 – s license provides that it applies to the "*use, copying, modifying and Conveying of Covered Source and Products and the Making of Products*". In its essence, this means that the license will apply to any use, change or publication of the licensed material as well as the making of any hardware.

4. Rights and Obligations

This section is the core of any OSH-license since it governs what the licensee can and cannot do with the licensed piece of OSH.

Usually, OSH-licenses include the grant of a non-exclusive, no-charge, royalty-free, irrevocable license to specific rights such as copyright or patent. This means that anyone can take a license (non-exclusive), without having to pay for it (no-charge; royalty-free) and the license cannot be taken back by the licensor (irrevocable).

In exchange for this grant, if you, the licensee, publish the licensed material or part of it or if you publish any modifications thereof, OSH-licenses usually require you to give notice. In particular, this can include the following obligations:

- You must give recipients of the licensed material or your modifications thereof a copy of the license.
- You must indicate the parts you modified or changed.
- In case of modifications, you must describe how to make these modifications and where to get the necessary material.
- You must retain any legal notice (e. g. copyright, attribution, trademark) previously in place and pass it on to the next recipient.
- You must retain any other notice and pass it on to the next recipient.

In addition, and this is crucial, some OSH-licenses will require you to make any modifications you publish available under the same license.

> **Example:** For instance, if you download and modify a design in a CAD-file licensed under CC-BY-SA 4.0, the license requires you to use the same CC-BY-SA 4.0 license, a later version of it or a license that is compatible with the CC-BY-SA 4.0, i. e. that has the same standards.

As described above, this is often called the “viral effect” of open source licenses and works especially well with copyright, because copyright arises automatically.

5. Disclaimer and Liability

This section usually tries to limit the licensor’s liability as far as permitted by law.

> **Example:** The TAPR OHL states that “IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW WILL ANY LICENSOR BE LIABLE” (emphasis added).

The current set of Open Source Hardware (OSH) licenses reflects the fact that for many developers, OSH projects are driven by passion rather than by financial motives. Typically, developers work on these projects in their spare time without the intention of profiting financially. Furthermore, OSH is often available at no cost. Consequently, developers generally avoid assuming legal responsibility for the quality of the OSH they develop or for any other damages.

The extent to which liability can be excluded varies by law, which differs significantly across countries. In most countries, liability for certain damages such as personal injuries cannot legally be excluded (so-called “strict liability”).

Under German law, a clause excluding liability "to the extent permitted by law" (so-called “limitation of liability clause”) may not be valid. The reason is that such a clause may be considered to be a General Term which requires transparency. This means that the contractual partner must be able to determine her rights simply and clearly without outside assistance, so that she is not deterred from enforcing them. This may not be possible in the case of a severability clause such as the one mentioned above. Therefore, the licensee may be fully liable under the statutory provisions. For more on OSH and liability under German law, see Guideline No. 3 on Liability.

6. Termination

The “Termination” section in OSH-licenses stipulates, under which conditions a license will terminate.

As mentioned before, OSH licenses are usually not limited in time. However, especially in case of IP-left-licenses, if you, as licensee, do not comply with the terms of a license, it will automatically terminate.

> **Example:** You do not attach a copy of the license for subsequent users while publishing your modifications to a design. Your own license will terminate immediately, and you will automatically be subject to copyright infringement.

As a result, you should take special care to fulfill the requirements of any OHS-license.

VIII. Why OSH standard licenses should not be modified

Sometimes, you may feel that a standard license does not fit the needs of your project. In this case you may want to take a standard license and modify this license to your needs.

However, we would strongly discourage modifying an existing standard license for the following reasons:

a. Adding or modifying provisions risks foreclosing, explicitly or implicitly, one of the four freedoms, which would render the license “non-open” and ineffective as OSH license;

b. From a practical standpoint, community resistance may be expected, as the community is used to standard licenses.

Therefore, our recommendation is to draft a new, project-specific license if needed.

C. Selected OSH-licenses in detail

The following chapter will give an overview of the details of selected OSH-licenses.

Since the CERN 1.2 and Creative Commons licenses seem amongst the OSH-licenses most commonly used, they will be the focus of the following analysis. Special attention will also be given to the CERN 2.0 licenses. Theses licenses are relatively new, but they are an excellent example of the different flavors of OSH-licenses as described above.

I. CERN OHL 1.2

CERN OHL 1.2 is a popular Open-Hardware-License (OHL) developed by the European Organization for Nuclear Research (CERN).

This license is applicable to the use and distribution of documentation as well as the manufacture and distribution of physical products. The license is probably not well suited for licensing software on its own. Indeed, Section 2.3 of the license clarifies that the license is applicable to software, firmware or code as part of hardware, if it is explicitly made available under the license. This means the licensor must explicitly state that the license also applies to the software in its product. However, when licensing only software, there usually is no corresponding documentation, which the license requires to make available to any recipient of the licensed material.

CERN OHL 1.2 is one of the first licenses to distinguish between “Documentation” and “Product”. In terms of the license, “Documentation is basically any explanatory material made available under the license, while “Product” is any device or part thereof built using the Documentation.

From this distinction, it follows that CERN OHL 1.2 explicitly grants a patent and design license (Section 3.5) and can therefore be used for defining a patented or design protected hardware as open source. Please note, however, that the license does not seem to apply to utility patents, since they are not mentioned in Section 3.5.

CERN OHL 1.2 allows you, as licensee, to do the following (not limited to non-commercial activities):

- Copy, publish and distribute the licensed Documentation (Section 3.2).
- Modify the Documentation or any part thereof (Section 3.3).

- Publish and distribute the modified Documentation (Section 3.4).
- Make and distribute the Product (Section 4.1).

CERN OHL 1.2 requires a licensee in turn to do the following:

- Keep intact all copyright, trademark, Documentation Location and other notices that refer to the disclaimer of warranties included in the Documentation.
- Make available / publish any modified Documentation from a location that can be easily located.
- Apply prominent modification notices to modified Documentation as well as a new Documentation Location notice.
- Make available modified Documentation in a similar format compared to the original (should be readable with software licensed with OSI-approved license).
- Make available modified Documentation under CERN OHL 1.2.
- When making a Product, provide to each recipient an easy means of accessing the corresponding Documentation.

In addition, a licensor (who publishes the design documentation) can express the wish to be informed about any production of the hardware. However, compliance by the licensee (who builds the licensed product) in this case is voluntary.

CERN OHL 1.2 also tries to limit the licensor's liability for defects or damages. Disclaimers of warranty and liability (Sections 5.1 and 5.2) stipulate that all risk associated with the Documentation (e. g. infringement of patents of third parties) and with the product (e. g. fitness for a particular purpose) is with the licensee and not the licensor. Accordingly, the licensor shall not be held liable for any possible damages occurring in conjunction with the Documentation or the product. This clause, is, however, probably void, as in many jurisdictions – for instance under German law – you cannot legally exclude your entire liability, especially not for defects of physical products.[10]

CERN OHL 1.2 automatically terminates, if the licensee violates the license conditions or takes legal action against the licensor (Section 6.4). However, the licensor must give written notice first.

[10] See for instance § 309 Nr. 7 BGB, § 14 ProdHaftG.

The latest version of the license can be obtained from the Open Hardware Repository.[11]

For examples of how to implement the CERN 1.2 OHL, please see the corresponding how-to.[12]

II. CERN 2.0 licenses

The CERN Open Hardware Licenses Version 2.0,[13] released in February 2020, are providing another legal basis for the sharing of documentation and products by giving licensors different options for sharing mechanisms: a strongly reciprocal, a weakly reciprocal and a permissive license.[14]

1. General Remarks regarding all CERN OHL 2.0 licenses

Although people may still tend to associate CERN OHL 2.0 licenses especially with electronic devices and their designs at circuit-board level, all CERN OHL 2.0 licenses are applicable to the use and distribution of documentation as well as the manufacture and distribution of physical products. Just as CERN OHL 1.2, the CERN OHL 2.0 licenses are probably not well suited for licensing software on its own. Indeed, according to Section 1.3 and 1.4 (CERN 2.0 – s), material covered by the license must always be information which can be applied to make or test a product. Although it is possible to understand the term “Product” (CERN 2.0 – s, Section 1.5) as (digital) work, the licenses rather seem to be targeted at making physical objects. Against this backdrop, under all the CERN OHL 2.0 licenses, “Documentation” basically means any information used to make a Product and “Product” any device, component, work or physical object made based on the corresponding Documentation.

The CERN OHL 2.0 licenses contain an implied license to any IP-rights necessary to copy, modify and distribute the Documentation or to make and distribute the corresponding Product. With regard to patents, the licenses also include an explicit license. Please note that all CERN OHL 2.0 licenses also contain a patent retaliation clause: Licensees who sue the licensor for patent infringement automatically lose their rights under the license.

[11] See https://ohwr.org/project/licences/wikis/cern-ohl-v1.2. Last accessed 01 September 2022.

[12] Available at https://civiclaboratory.files.wordpress.com/2016/06/cern_ohl_v_1_2_howto.pdf. Last accessed 01 September 2022.

[13] Available at https://ohwr.org/project/cernohl/wikis/Documents/CERN-OHL-version-2. Last accessed 01 September 2022.

[14] See https://ohwr.org/project/cernohl/wikis/Documents/CERN-OHL-version-2. Last accessed 01 September 2022.

All CERN OHL 2.0 licenses therefore allow a licensee to do the following (not limited to non-commercial activities):

- Copy, publish and distribute the licensed Documentation.
- Modify the Documentation or any part thereof.
- Publish and distribute the modified Documentation.
- Make and distribute the Product.

In turn, the requirements of CERN OHL 2.0 licenses depend on which license you choose. For more information on these requirements see the below sections on each specific license.

Regarding warranties and liability, all CERN OHL 2.0 licenses exclude warranties and liability to the maximum extent permitted by law. This means that, depending on the applicable law, liability for infringement or any defects may not be completely excluded.

> **Example:** Under German law, for instance, this clause may be invalid for lack of transparency.

For more information on liability under German law, see Guideline No. 3 on Liability.

In addition to being one of the few available full OSH-licenses, CERN OHL 2.0 are the first licenses to introduce the concept of so-called “Available Components”. The concept of Available Components means that under the CERN 2.0 licenses, it is possible to license pieces of OSH which are built with the help of generally and readily available components, even if they have to be made by the licensee, be paid for or the licensee has to obtain a license to get hold of them. The advantage from the licensor’s point of view is, that the licensed Documentation does not need to include any information on how to build these components (CERN OHL 2.0 – s, Section 1.8). As long as the any downstream user has access to this component, any component can potentially be an available component.

> **Example:** The licensed piece of OSH is a clothes-drying-rack. To build it, a licensee needs the design schematics as well as nails and a hammer. As nails and hammers are readily available in every hardware store, you do not need to include information on how to make the nails and the hammer in the licensed Documentation. You can still license the schematics as OSH under CERN 2.0. However, you will need to indicate

what nails need to be used, where to buy them and how to use them to make the drying rack.

Example: The licensed piece of OSH is a motor. To build it, a licensee needs a circuit board the schematics of which are available on the internet under the TAPR 1.0. As the circuit board can be treated as Available Component, you do not need to include the information on how to build it in your motor's overall Documentation. However, you will need to indicate where to find the information on how to build the circuit board.

Example: The licensed piece of OSH is a heat sensor. To make it fully functional, a licensee needs firmware licensed under the GPL 3.0. As the firmware can be treated as Available Component, you do not need to include the information on it in heat sensor's overall Documentation. However, you will need to indicate where to find the information on how to build the circuit board.
Please note that this example only applies to the CERN OHL 2.0 – w and not the CERN OHL 2.0 – s (for more on this see below).

With the help of Available Components, CERN 2.0 therefore allows you as licensor to make your product open source, even if you don't have the complete Documentation of all components or if some components are licensed under an incompatible license but are still available. Even closed components can thus be incorporated into a piece of OSH under the condition that they are readily available, and all the relevant interface information is provided.

Please note that the concept Available Component works slightly differently depending on which variant of CERN OHL 2.0 is used. You will find more information on this below at the relevant license. Please also note that, if a component is included in a piece of OSH licensed under CERN OHL 2.0 and it is NOT an Available Component (e. g. when it is not readily available), the information on building that component must usually be made available as part of the complete information on the Product. In this regard, problems may arise when a component of a piece of OSH was readily available when the piece of OSH was first licensed, but later becomes unavailable. This may lead to a violation and thus automatic termination of the license.

Example: The piece of OSH is a circuit board. A licensee modifies the design adding readily available semiconductor chips to improve on its performance. Imagine there is a global economic crisis, and the chips needed to build the circuit board including your modifications are no longer available on the open market. This may lead to the

Documentation being incomplete. The license to the circuit board may therefore automatically terminate.

Example: The piece of OSH is a motor. A licensee integrates the design for a circuit board and treats it as an Available Component, since it is published under the TAPR 1.0 license in an online repository. Later, the original author deletes the repository, and another licensee cannot find the design anywhere else on the internet. This may lead to the Documentation being incomplete. The license to the circuit board may therefore automatically terminate.

With regard to choosing one of the CERN OHL 2.0 licenses, please note that, if the licensor does not choose one the different flavors of CERN OHL 2.0, the CERN OHL 2.0 – s is deemed to be chosen (see for instance, CERN OHL 2.0 – p, Section 7.3).

Example: The licensor publishes the piece of OSH under “CERN 2.0”. In this case, only the CERN OHL 2.0 – s is applicable.

2. CERN OHL 2.0 – s

The CERN OHL 2.0 – s license is a strongly reciprocal OSH-license. This means that it requires you as licensee to strongly cooperate to make the licensed piece of OSH as accessible as possible.

Against this backdrop, the CERN OHL 2.0 – s requires you to:

- Keep all notices intact (Section 3.1)
- When modifying the Documentation, add a corresponding notice (Section 3.3 b).
- When modifying the Documentation, license the modifications under the terms and conditions of CERN OHL 2.0 – s (Section 3.3 d).
- When distributing a Product, make available any modified Documentation (Sections 3.2; 4).
- When modifying the Documentation and distributing the Product, add a source location notice on where to find the modified Documentation (Section 3.3 c).

Please note that CERN OHL 2.0 – s automatically terminates if you do not comply with one of these requirements (Section 8.4). Prior written notice is not necessary. However, you can be reinstated if you cease all breaches of the license (Section 8.5).

In contrast to CERN OHL 1.2, the CERN OHL 2.0 – s no longer requires you to publish modifications of the Documentation in any case. You just have to do this if you also make and distribute the modified Product. Private use is therefore possible under the CERN OHL 2.0 – s.

Regarding the concept of Available Components in the strongly reciprocal CERN OHL 2.0, only physical components may be Available Components in the sense that the licensee can make them herself (CERN OHL 2.0 – s, Section 1.7 b i) or use software licensed under an incompatible license. This means that digital components will either have to be available under a compatible license (CERN) or be part of the normal distribution of tools. The possibility to integrate third-party software into CERN OHL 2.0 – s - licensed software are therefore limited compared to CERN OHL 2.0 – w.

3. CERN OHL 2.0 – w

The CERN OHL 2.0 – w license is a weakly reciprocal OSH license. This means that it requires you as licensee to cooperate less to make the licensed piece of OSH accessible compared to the CERN OHL 2.0 – s.

Against this backdrop, the CERN OHL 2.0 – w requires you to:

- Keep all notices intact (Section 3.1)
- When modifying the Documentation, add a corresponding notice (Section 3.3 b).
- When modifying the Documentation, license the modifications under the terms and conditions of CERN OHL 2.0 – s (Section 3.3 d).
- When distributing a Product, make available any modified Documentation (Sections 3.2; 4.1).
- When modifying the Documentation and distributing the Product, add a source location notice on where to find the modified Documentation (Section 3.3 c).

Please note that CERN OHL 2.0 – w automatically terminates if you do not comply with one of these requirements (Section 8.4). Prior written notice is not necessary. However, you can be reinstated if you cease all breaches of the license (Section 8.5).

In contrast to CERN OHL 1.2, the CERN OHL 2.0 – s no longer requires you to publish modifications of the Documentation in any case. You just have to do this if you also make and

distribute the modified Product. Private use is therefore possible under the CERN OHL 2.0 – w. In addition, the CERN OHL 2.0 – w license allows you to license Products under the terms of your choice; they need not be licensed under CERN OHL 2.0 – w terms (Section 4.3).

In contrast to CERN OHL 2.0 – s, CERN OHL 2.0 – w contains terms that make it easier for you to use third party material, which might not be licensed under CERN or even any open-source license. For instance, regarding the concept of Available Components in the weakly reciprocal CERN OHL 2.0, any component, physical or digital, can be an Available Component in the sense that it can be made by the licensee herself (CERN OHL 2.0 – w, Section 1.7 b i). In addition, physical and digital components count as Available Components, if they are available under a compatible license or if they are part of the normal distribution of tools. In contrast to CERN OHL 2.0 – s, CERN OHL 2.0 – w therefore allows the interfacing of third-party software, even if it is not licensed under a compatible (CERN) license nor available as part of a standard tool.

In addition to this understanding of Available Components, the CERN OHL 2.0 – w introduces the concept of "External Material". The term External Material means material which is combined with the licensed material using a documented interface (Section 1.8). Documentation of this material also does not need to be included in the licensed Documentation (Sections 3.3 d; 4.2). The requirements for an interface in terms of the license, however, have to our knowledge not been tested yet. They will probably depend on the individual case.

4. CERN OHL 2.0 – p

The CERN OHL 2.0 – p license is a permissive OSH license. This means that the licensed piece of OSH is basically free. As licensee, you do not have as many obligations compared to the other CERN OHL 2.0 licenses.

Against this backdrop, the CERN OHL 2.0 – p requires you to:

- Keep all notices intact (Section 3.1)
- When modifying the Documentation, add a corresponding notice (Section 3.3 b).
- When distributing modified Documentation, provide a copy of the CERN OHL 2.0 – p to any recipient (Section 3.4 b).
- When distributing a Product, ensure that any recipient has access to the corresponding notices (Section 4).

In contrast to the other CERN OHL 2.0 licenses, the CERN OHL 2.0 – p license seems to only terminate in case of patent retaliation (Section 6.2). Otherwise, no termination clause is included. This probably makes extraordinary termination due to serious breach of duty the only other possibility to terminate the license.

5. How to implement the CERN OHL 2.0 licenses

For information on how to implement the CERN OHL 2.0 licenses, please see the corresponding user guides.[15]

III. TAPR OHL

The Tucson Amateur Packet Radio (TAPR) Open-Hardware-License,[16] drafted by *John Ackermann* in 2007, was probably the first of its kind.

Like the CERN OHL 1.2, the TAPR OHL is applicable to the use and distribution of documentation as well as to the manufacture and distribution of physical products. Unlike it, however, the TAPR OHL is not applicable to software, firmware or code, even if it is part of a "Product" (Section 1.6). Although the license was originally designed to be best suited for licensing circuit boards, it probably works well with all other pieces of OSH (see the Preamble).

Like CERN OHL 1.2, the TAPR license is one of the first licenses to distinguish between documentation and Product. In terms of the license, "Documentation" is basically any explanatory material made available under the license in tangible or intangible form (e. g. data), while "Product" is any physical device or part thereof based in whole or in part on the Documentation (Sections 1.2 and 1.3). In contrast to CERN 1.2, TAPR grants the licensee licenses to all Intellectual Property Rights attached to the licensed material and with regard to making a Product (Sections 2.1, 4.1 and 4.2).

TAPR allows you as licensee to do the following (not limited to non-commercial activities):

- Copy, publish and distribute the licensed Documentation (Section 4.1).
- Modify the Documentation or any part thereof (Section 3).

[15] Available at https://ohwr.org/project/cernohl/wikis/Documents/CERN-OHL-version-2. Last accessed 31 December 2024.

[16] Available at https://files.tapr.org/OHL/TAPR_Open_Hardware_License_v1.0.pdf. Last accessed 31 December 2024.

- Publish and distribute the modified Documentation (Section 4.2).
- Make and distribute the corresponding Product (Section 5.1 and 5.2).

TAPR requires you as licensee in turn to do the following:

- Keep intact all copyright, trademark and other notices included in the Documentation you received (Section 4.1).
- Apply prominent modification notices in an open format to modified Documentation (Section 4.2 a).
- When distributing Documentation, the whole Documentation, including new, original, and modified files, must be sent (Section 4.2 b).
- Include the TAPR license agreement when distributing Documentation (Section 4.1).
- When making and distributing a Product, include in the Documentation all elements reasonably required to permit others to make Products, including any files used for manufacturing (e. g. CAD files). The Documentation must be distributed in an open format.
- When making and distributing a Product, keep intact all notices applied to the Product.

As you can see from the list of obligations above, the requirements of the TAPR license may be difficult to meet in in practice. This is especially the case considering it is unclear if the Documentation made available needs to be sent in an electronic format or if it is sufficient to simply post it online. In addition, the obligation to make available all Documentation "*reasonably required to make a Product*" is vague and thus might be subject to future dispute. Furthermore, under the TAPR license, the licensee is subject to further obligations not included in any other hardware license. According to the TAPR license, the Documentation must not only be passed on downstream, but also upstream to earlier developers, if a contact e-mail was provided along with the original material (Section 3).

Regarding warranties and liability, the TAPR license probably does a better job at excluding these than CERN OHL 1.2, since warranties and liability are only excluded "*to the extent permitted by applicable law*" (Section 7.1) or unless required otherwise by applicable law (Section 7.2). Please note that consequently, depending on the applicable law, liability might still not be excluded. For more on liability under German law, see Guideline No. 3 on Liability.

In contrast to CERN OHL 1.2, the TAPR license automatically terminates if any of the licensing conditions is not met. Giving notice is not required under the TAPR license.

With regard to license compatibility, TAPR 1.0 declares any following version compatible, but only if this is explicitly specified in the licensed Documentation. As of now, we are not aware of any subsequent version of the TAPR license.

IV. Apache 2.0 and Solderpad 2.1

The Apache 2.0[17] and Solderpad 2.1[18] licenses are two permissive licenses, where Apache 2.0 is a permissive software license and Solderpad 2.1 acts as a wraparound for associated hardware projects.

Apache is a Software Foundation founded in 1999, that develops, supports and incubates software projects in categories such as Artificial Intelligence, Big Data, Cloud Computing etc. According to its mission statement, one of its missions is to provide OSS for the public good.[19] To this end, Apache created the Apache-Open.Source-Software-license. The Solderpad license was later developed by the Free and Open Source Silicon Foundation (FOSSi), a non-profit organization with the mission to promote and assist free and open digital hardware designs and their related ecosystems.[20]

The current version of the Apache license, Apache 2.0, covers copyright and patent rights to the licensed material. While it can primarily be seen as a software license, its scope is not limited to source or object code. Rather, according to its definitions, “source form” is the preferred form for making modifications, while “object form” is the form resulting from mechanical transformation or translation of source form.

Being a permissive license, the Apache License does not require you as licensee to make your modifications available under the same conditions as the licensed material. However, if you send your modifications back to the licensor for inclusion into the licensed material intentionally, the original conditions will also apply to your modifications, unless you expressly object or have a separate agreement with the licensor (see Apache 2.0, no. 5).

Otherwise, Apache 2.0 essentially requires you as licensee to (Apache 2.0, no.4):

- Give a copy of Apache 2.0 to any recipient of the licensed material or any modification.
- Give notice that you modified the licensed material if you did.

[17] Available at https://www.apache.org/licenses/LICENSE-2.0. Last accessed 31 December 2024.
[18] Available at http://solderpad.org/licenses/SHL-2.1/. Last accessed 31 December 2024.
[19] See https://www-archive.fossi-foundation.org/licensing. Last accessed 31 December 2024.
[20] See https://www.apache.org/. Last accessed 31 December 2024.

- Retain any copyright, patent, trademark or attribution notices.

In addition, in case patents are part of the licensed material, you must not start a patent invalidity action, otherwise the license automatically terminates.

Solderpad 2.1, as a wraparound, is a very short license that simply adds to, or changes the wording of, Apache 2.0 (see Solderpad 2.1, no. 1).

> **Example**: Solderpad 2.1 clarifies that “source form” also means “board layouts, CAD files, documentation source, and configuration files”. In addition, “object form” also means the application of a source form to physical material and specifically includes “instantiation of a hardware design or physical object or material and conversions to other media types, including intermediate forms such as bytecodes, FPGA bitstreams, moulds, artwork and semiconductor topographies (mask works)”.

In contrast to Apache 2.0, Solderpad 2.1 covers copyright, design rights, rights in semiconductor topographies and database rights, excluding patents and trademarks.

Considering that Apache was originally designed for software projects, it is probably not best suited for OSH-licensing. One reason is that its definitions, for instance the division between source form and object form, are clearly tailored for software. In case of hardware, this division is not necessary and may lead to confusion in the interpretation of the license.

> **Example:** In case of hardware, the license does not clarify, what “object form”, meaning “any form resulting from mechanical transformation or translation of a Source form” would look like.

Solderpad 2.1 on the other hand might be a good approach regarding projects involving semiconductors or reprogrammable logic chips (FPGAs), since the license was originally developed to address this technology.

One downside in using Solderpad 2.1 for other pieces of OSH, however, is that it probably does not cover patent rights. In this regard, the exclusion of patent rights in the rights’ definition in Solderpad 2.1 is contradictory. On the one hand, Apache 2.0 already contains a license grant for patents and according to Solderpad 2.1, no. 1, the definition for “Rights” in Solderpad is an “new definition”. This could mean that the original patent license in Apache 2.0 stays in effect. On the other hand, the rights’ definition in Solderpad 2.1 specifically mentions copyright, which would not have been necessary if the original copyright license of Apache 2.0. was left in place. In addition, regarding the “license grant” in Solderpad 2.1, no. 2, the license does not clarify whether this license grant replaces the original copyright and patent licenses in Apache 2.0.

In summary, Apache 2.0 and Solderpad 2.1 are permissive licenses that probably work best applied to Software or semiconductor and FPGA-related OSH. The benefit of Solderpad 2.1 is that it additionally grants a license in design rights, rights to semiconductors and database rights. With regard to patent rights, however, it seems likely that these rights are not included in Solderpad 2.1.

From a licensor point of view, the lack of clarity regarding patent rights would not necessarily be problematic, since Solderpad 2.1 is a permissive license. Using this license, a licensor should generally have little interest in restricting use of the licensed material. Lack of clarity could, however, deter licensees from using material licensed under Solderpad 2.1. This is because a licensee will have to rely on the license’s grant of rights and might feel at risk for infringement, if the scope of the license is not sufficiently clear.

V. GPL 3.0

The GNU General Public License, Verison 3 (GPL 3.0)[21] is one of the most commonly used copyleft licenses for making available open-source software. The Free Software Foundation also recommends its use for physical objects such as design of protected circuits.[22]

Despite this recommendation, the GPL 3.0 may not be best suited for OSH-licensing. The license does not distinguish between documentation and product, which may have different licensing needs (e. g. distribution of documentation must be non-commercial, but not the distribution of products). In addition, the license grant depends on copyright (Section 2). Although corresponding patent rights may also be licensed (Section 11), other important rights such as designs and utility patents, are not covered by the license. Finally, the wording of the license is clearly targeted at licensing software and may create ambiguities when applied to hardware (e. g. “Source Code”, “Program”, “System Library”, etc.).

[21] Available at https://www.gnu.org/licenses/gpl-3.0.html.en. Last accessed on 31 December 2024.

[22] See https://www.gnu.org/licenses/license-list.html.en at the bottom. Last accessed 31 December 2024.

VI. LGPL

The Lesser General Public License (LGPL)[23] is a permissive license.

For the same reasons as mentioned above with regard to the GPL 3.0, the LGPL is probably not best suited for the use with OSH.

VII. Mozilla 2.0

The Mozilla Public License 2.0[24] is another copyleft license intended for licensing software.

The wording of this license is probably slightly more open to being used with hardware compared to the GPL licenses. In particular, Mozilla 2.0 covers all intellectual property rights, except trademarks.

However, this license is also clearly intended for use with software and therefore probably not best suited for OSH-licensing. Mozilla 2.0 does not use the terms “Documentation” or “Product” and does not distinguish between these terms. Instead, it uses the terms “source form” and “executable form” that are commonly used in software development. This leads to legal uncertainty. The license does not clearly tell users what parts of a piece of OSH are licensed and under what conditions when applied to hardware.

VIII. CC-licenses

The Creative-Commons-Licenses (CC-licenses)[25] are probably the best-known licenses for making available copyright protected works.

Similar to the CERN 2.0-licenses, CC-licenses come in different flavors. Licensors can inter alia decide whether they want to be credited as authors, whether modifications should be shared under the same licensing conditions, whether modifications are allowed, or whether only non-commercial use should be allowed under the license.

[23] Available at https://www.gnu.org/licenses/lgpl-3.0.html.en. Last accessed on 31 December 2024.
[24] Available at https://www.mozilla.org/en-US/MPL/2.0/. Last accessed on 31 December 2024.
[25] For an overview over CC-licenses, see https://creativecommons.org/licenses/. Last accessed 31 December 2024.

CC-licenses are designed to be applied to artistic or literary works or databases. Although they may be applied to other material, CC-licenses are probably not best suited for OSH-licensing.

One reason for this is that CC-licenses only apply to copyright and similar rights such as performance, broadcast, sound recording and database rights. Patents as well as trademark rights are explicitly not covered by CC-licenses. Design rights and rights to utility patents are likely also outside the scope of these licenses.

> **Example:** CC-BY-SA 4.0, Section 1. d., h. and i.[26]

Another aspect weighing against using CC-licenses for OSH is the lack of distinction between documentation and product.

> **Example:** CC-BY-SA 4.0, Section 2 a. 1.[27] allows the reproduction and sharing of licensed material. Applied to OSH, it remains unclear if this license grant allows the licensee to make a product. Assuming that the documentation is the "licensed material", it is possible that the CC-license only covers sharing the documentation but not making the product. Making the product may not be seen as "reproducing the licensed material".

[26] Available at https://creativecommons.org/licenses/by-sa/4.0/legalcode. Last accessed 31 December 2024.

[27] Available at https://creativecommons.org/licenses/by-sa/4.0/legalcode. Last accessed 31 December 2024.

D. License compatibility

License compatibility is important, when you want to combine components of OSH which are licensed under different licenses. Licenses are compatible when components licensed under different licenses can legally be combined and distributed.

> **Example:** A circuit board design might be licensed under CC-BY-SA 4.0, whereas a motor might be licensed under CERN 2.0 – s. You want to combine the two into one product. This may not legally possible, as the CC-license requires to name the author, but the CERN license requires to also provide information on tools needed to build a piece of OSH.

License compatibility can also come into play when you plan on publishing modifications of OSH-content under a different license than the original one, because you believe that it is better suited to your purposes.

> **Example:** You obtained a circuit board design licensed under CERN 2.0 – s as licensee and wish to publish your corresponding modifications under CC-BY-SA 4.0. This is not legally possible, as the CC-license is less strict than the CERN-license and does not meet the CERN requirements.

In these cases, combination or re-licensing may only be possible, if the relevant licenses are compatible with each other. Otherwise, you risk infringement of the licensing conditions and may open yourself up to liability for IP infringement.

> **Example:** In some open-source-licenses, you may find the requirement that modifications must be licensed "under the same conditions". A compatible license is one that fulfills "the same conditions" as the original license.

> **Example:** In some open-source-licenses, you may find the requirement that modifications must be licensed "under this license or any later version". This means that all later versions of that license are compatible with the original.

For information on license compatibility for software licenses see the list published by the Free Software Foundation.[28]

[28] See https://www.gnu.org/licenses/license-list.html.en. Last accessed 31 December 2024.

Compatibility of OSH-licenses has to our knowledge not yet been the subject of further study. As a rule of thumb, however, you can probably not change between different flavors of OSH-licenses.

> **Example:** You can probably not change from CERN 2.0 – s to CERN 2.0 – p. It might, however be possible to change from CERN 2.0 – p to CERN 2.0 – s.

In general, to avoid the issue of license compatibility, you should consider only using licenses which are explicitly described as compatible by the license developer.

From the initial licensor's point of view, it follows that choosing the right license is critical to downstream users. If you chose a license that is complicated to use, you or any downstream user may not be able to change to another license later. To do this, you would probably need the consent of every user who has contributed to the specific piece of OSH. This may limit the Project's usefulness to downstream users.

E. Multi-Licensing and Version-Lagging

Some strategic considerations you might want to take into account with regard to value creation or license compatibility relate to Multi-Licensing or Version-Lagging.

“Multi-Licensing” means that you publish the same material under different licenses.

> **Example:** You have developed a design for a rotor. You publish it under CC-BY-SA 4.0 as well as CERN 2.0 – s.

The upside of this is that you may avoid license compatibility issues described above. You may try to license your piece of OSH under multiple licenses and see, which license proves most popular and practical.

The downside of this strategy is that you basically create licensing forks that cannot be changed back. Lacking compatibility, contributions in one fork are therefore probably not available to other forks.

> **Example:** You licensed the design for a rotor under CERN 2.0 – s as well as under CERN 2.0 – p. Assuming the licenses are not compatible, any contributions made available under CERN 2.0 – s will probably not be available to users only interested in licensing under CERN 2.0 – p.

“Version-Lagging” means licensing different versions of a piece of OSH under different licenses. This strategy may be suitable if you have a complex piece of OSH and are planning to develop it further or to market it. It can create opportunities for value creation, such as involving professional designers in the creation of OSH.

> **Example:** You created an open-source-MRI device. You make the basic model, which can only scan extremities, available under CERN 2.0 – s. You have another model, which can do whole-body scans and has more diagnostic options. This model you do not make available under an OSH license but instead sell it and the corresponding services on the open market.

F. Strategic Implications for FabLabs

The strategic implications of the OSH-licensing scheme as described above depend on the role FabLabs wish to take in the open-source innovation process.

If FabLabs merely intend to bring together people and tools to enable them to create whatever they need, FabLabs themselves probably do not need to implement a certain license. In that case, it is likely sufficient for the FabLab maintainers to have sufficient knowledge of licenses to be able to assist users in choosing and implementing the license which best suits their respective purposes as licensors or to assist users in fulfilling their obligations as licensees.

If however, FabLabs intend to act as a “knowledge hub” in which they take greater control of OSH-licensing, a proper licensing strategy and the type of OSH-license that best suits their purpose should be considered. The OSH-licensing could then be managed by the FabLab itself and should be set forth in appropriate statutes or rules on FabLab-use.

G. Overview – classification of licenses

Permissive OSH-Licenses	OSH licenses with weak IPleft-effects	OSH licenses with strong IPleft-effects
Apache 2.0	CERN OHL 2.0 – w	CC-BY-SA
CC0 1.0		CERN OHL 2.0 – s
CERN OHL 2.0 – p		CERN OHL 1.2
LGPL		GPL 3.0
Solderpad 2.1		MOZILLA 2.0
		TAPR 1.0

H. FAQ – OSH licensing

I. Introduction to OSH licensing

1. Why is licensing necessary for Open Source Hardware (OSH)?

Licensing is essential to ensure OSH remains accessible and usable for downstream users. Without it, OSH could be appropriated, losing its openness, and users might be restricted from freely using, modifying or sharing it. (see Section A.)

2. What is the difference between OSS and OSH licensing?

OSH licenses must take hardware into account, which has different IP rules compared to software. While OSS licenses are often sufficient for digital code, OSH involves tangible products that require documentation and hardware-specific licensing. OSS-licenses generally do not sufficiently cover the rights in (1) the documentation as well as (2) the physical hardware that may be created based on the documentation. (Section B.VI.)

3. What are the two most typical legal questions someone involved in an OSH project might have?

First, do you have the necessary rights to do what you need to do without infringing the rights of others? Second, how can you effectively give users the rights they need under the conditions of your choice? (Section B)

4. How does the fact that OSH combines both tangible and intangible assets impact licensing?

Usually, licensing a piece of OSH consists of two parts: (1) licensing the documentation that instructs the user how to build the OSH-product (intangibles) and (2) licensing the making of the product itself (tangible). The effect of the license is to keep both accessible and usable to downstream users. (Section 0B.VI.)

5. What factors should I consider when selecting an OSH license?

When choosing an OSH license, consider your goals for openness, control, and potential commercial use. Decide how much you want to regulate derivative works and if you require attribution or modifications to be open. (Section B.III.; for the question of reciprocity and the IP-left-effect see Section B.IV.)

II. Types of OSH licenses and their applications

1. What are the main types of OSH licenses?

There are three primary types of OSH licenses: strongly reciprocal (strong IP-left), weakly reciprocal (weak IP-left), and permissive (non-reciprocal), each offering different levels of openness and control. (Section B.III. ff.)

2. What are strongly reciprocal (strong IP-left) licenses, and when should they be used?

Strongly reciprocal licenses require all modifications and derivatives to remain open source, promoting continued accessibility and openness (e.g., CERN OHL 2.0 – s). They are best for projects prioritizing downstream accessibility. (Section B.V.1.)

3. What are weakly reciprocal (weak IP-left) licenses, and in what situations are they appropriate?

Weakly reciprocal licenses require limited openness for modifications. Contrary to strongly reciprocal licenses, some parts, such as software used to make the product, may be proprietary. They are useful when a balance between openness and proprietary elements is desired (e.g., CERN OHL 2.0 – w). (Section B.VI.)

4. What are non-reciprocal or permissive licenses, and how do they work?

Permissive licenses allow broad use, including commercial applications, without requiring derivatives to be open source (e.g., CERN OHL 2.0 – p), offering flexibility for both commercial and non-commercial applications. (Section B.V.3.)

5. What are the main differences between strong and weak IP-left licenses?

Strong IP-left licenses require all modifications to be shared openly, while weak IP-left licenses may allow some elements (e.g., software) to be proprietary, providing a middle ground. (Sections B.V. and B.VI.)

III. Specific OSH licenses and their features

1. What are CERN Open Hardware Licenses (OHL) and their different versions?

The “CERN OHL family”, released in February 2020, offers three options—strongly reciprocal, weakly reciprocal, and permissive—to match varying levels of desired openness and control in OSH projects. All CERN OHL 2.0 licenses are applicable to the use and distribution of documentation as well as the manufacture and distribution of physical products. (Section C.II.)

2. What do all CERN Open Hardware Licenses (OHL) 2.0 have in common?

They allow you to do the following (not limited to non-commercial activities):

- Copy, publish and distribute the licensed documentation;
- Modify the documentation or any part thereof;
- Publish and distribute the modified documentation;
- Make and distribute the product.

(Section C.II.1.)

3. Why is documentation crucial in OSH licensing, and what should it include?

Documentation is key for OSH, as it provides essential instructions for building and using the hardware. Licenses often require documentation to be publicly accessible, ensuring OSH’s usability. This means, e.g. in case of modifications, you must describe how to make these modifications and where to get the necessary material. (Section B.VII.4.)

4. What does 'Available Components' mean in OSH licenses?

The concept of Available Components means that under the CERN 2.0 licenses, it is possible to license pieces of OSH which are built with the help of generally and readily available components, even if they have to be made or paid for by the licensee, or the licensee has to obtain a license to get a hold of them. The advantage from the licensor’s point of view is, that the licensed documentation does not need to include any information on how to build these components. (Section C.II.1.).

5. What are patent retaliation clauses, and why are they included in OSH licenses?

Patent retaliation clauses automatically revoke a licensee’s rights under the license, if the licensee files an infringement suit against the licensor. These clauses are meant to promote cooperative use of patented OSH components. (Section C.II.1.)

6. Why are Creative Commons (CC) licenses generally not recommended for OSH?

CC licenses do not cover patent rights, design rights, or trademarks, making them less suitable for OSH projects that generally require broader IP protection than copyright alone. Additionally, they do not differentiate between documentation and product. (Section C.VIII.)

IV. Compliance, liability, and legal considerations

1. How do OSH licenses address liability for hardware defects or issues?

OSH licenses typically disclaim liability for defects, especially since OSH is often provided free of charge and developed in a non-commercial context. Please note that, depending on the applicable law, liability might not be excludable and such a clause might therefore be void in certain jurisdictions. (Sections B.VII.5., C.II.1, C.II.5.)

2. What happens if license conditions are violated?

OSH licenses generally terminate automatically if the licensee does not follow license terms, such as failing to provide a copy of the license with modifications. (Sections C.VII.6., C.II.1.)

3. Why is license compatibility important when combining OSH components?

License compatibility is crucial when merging different OSH elements, as incompatible licenses could lead to legal conflicts or restrict the use and distribution of the combined work. (Section D.)

4. How do OSH licenses handle improvements made by third parties?

Most OSH licenses, especially reciprocal ones (those having an IP-left effect), require that any improvements or modifications made by third parties be shared under the same license terms, ensuring continued openness. (Section B.VII.4.)

5. What is the 'viral effect' in OSH licensing?

The viral effect illustrates the IP-left effect. It means that modifications or derivatives of a licensed product must be licensed under the same terms as the original license, keeping the project and all derivatives open source. This IP-left effect thus works like a virus that impacts all enhancements and further derivatives. (Section B.IV.)

V. Strategic and practical considerations for OSH licensing

1. What is multi-licensing, and how does it benefit OSH projects?

Multi-licensing allows a project to be licensed under multiple terms, providing flexibility to users and potentially avoiding compatibility issues. (Section E.)

2. What is version lagging, and why might it be used in OSH licensing?

Version lagging involves licensing different versions of OSH under different terms, enabling flexibility in future developments and potential commercialization. (Section E.)

3. Can OSH licenses be modified to fit unique project needs?

Most OSH licenses are standardized to maintain consistency. However, some may allow minor adjustments, though any modification should align with the original license’s principles. (Section B.VIII.)

4. What is “defensive publishing”, and how does it protect OSH projects?

Defensive publishing involves disclosing information about an OSH project early, making it unpatentable by others and thus preventing exclusive rights claims. (Section B.II.2.)

5. How does OSH licensing support innovation in FabLabs?

OSH licensing enables makers/FabLabs to freely share, modify, and collaborate on projects, making it easier to innovate and improve collectively without IP restrictions and/or fear of IP infringement. (Section F.)

6. What role should FabLabs play in OSH licensing?

FabLabs can serve either as facilitators, helping users choose suitable licenses, or as knowledge hubs actively managing licensing strategies. (Section F.)