Filling the gap: Evaluating which legal mechanisms effectively protect freedom and openness in hardware

Author: Michele Marrali (u1261257)

Supervisor: Dizon M.A.

Second reader: Prof. Dr. Schellekens, M.H.M.

“When the winds of change blow some people build shelters and others build windmills.”

Ancient Chinese proverbs.

“Intellectual property is the oil of the twenty-first century”

Mark Getty

“Intellectual Property is an historical parenthesis”

Pirate Bay

Table of Contents

Chapter 0. An introduction to OSH

0.1 BACKGROUND

In the world of software, freedom and openness are well established business as well as collaborative development models. It has been three decades since the GNU project began. It than become the GNU/LINUX Operating System[1] (which many abbreviate as “Linux”) and now holds the biggest share in the market of mobile and server operating systems worldwide.

Many advantages of the Free, Libre and Open Source Software (FLOSS) development model are well documented[2], consequently during the last decade the ideals of freedom and openness in addition to the successful business model embedded in FLOSS (the business model very succinctly entail sharing the source code of the product, benefiting from imput, collaboration and labor from the community and profitting mainly from value added services, such as technical support, training, service selling, consulting and donations) has attempted to move into other areas of contemporary society, such as: art, literature and music, sustainable lifestyle, monetary system,[3] business and entrepreneurship activities,[4] and last but not less important hardware.[5]

In fact, hardware may be viewed as an alter ego for software, or at least the complementary part, since without the hardware software is useless. What the Open Source Hardware (OSH) communities around the world are developing, are not exclusively computer related, but various products ranging from microcontrollers to windmills, from electric vehicles to three-dimensional printable glasses. Many communities worldwide are developing projects that use the Internet to

---

1 W3tech : “Usage statistics and market share of Unix for websites”. Accessed, 19 Feb. 2014. Reachable at: http://w3techs.com/technologies/details/os-unix/all/all

2 Joseph Feller, Brian Fitzgerald, Scott A. Hissam and Karim R. Lakhani (editors), 2007. *Perspectives on free and open source software*. Cambridge, Mass.: MIT Press.

3 Respectively, Creative Common Licenses, Open source Ecology and Bitcoin and others cryptocurrency.

4 Ye, Huojie and Zhong, Shuhua, Business Accelerator Network: A Powerful Generator of Strategic Emerging Industries (August 16, 2012). OIDA International Journal of Sustainable Development, Vol. 04, No. 06, pp. 11-23, 2012. Available at SSRN: http://ssrn.com/abstract=2130921

5 For an extensive list of OSH http://p2pfoundation.net/Product_Hacking

share ideas and to make useful and interesting things. Some communities freely share informations on how to make the object while others choose to also commercialize the finished product.[6]

Many OSH projects use licenses such as instruments to manage intellectual property rights, this also because of the software heritage of a wide and rooted culture of the use of licenses, in fact many of the early OSH project started as a direct implementation of FLOSS principles and values translated into hardware context, since other than software the need for a four freedom (Infra 1.1.1) compliant hardware was felt[7] by FLOSS community. The effectiveness of copyright in the hardware context might not be straightforward, and because of this issue the main research question is "What are the legal instruments that can be used to protect the rights and freedoms of open source hardware communities, users and developers."

In this case the fundamental rights and freedoms that are at stake are the one to: use, study, improve, and share the work.

In order to address this question is important to to have a clear idea of what is OSH, where does it come from, and why is relevant, the first sub-question will than be, What is OSH? How is it different from FLOSS?

The next sub-question is: What are contractual and IP instruments that are currently used to protect OSH?

The final sub-question; What other legal instruments or mechanisms are available to protect the rights and freedoms in OSH? How can they be used?

---

6 Example of pure hardware information sharing might be, is open source ecology project, which aims to create a set of tools that allow self-sustainability this set of tools is composed of 50 is modular, DIY, low-cost, high-performance, machines is named global village construction set. Blueprints, schematics, list of parts and other needed material is listed in a wiky. A second type of project is OsVehicle which is a project that aims to build an OSH car, all designs are available for download, but parts or the hall vehicle can be purchased on-line from OSV manufacturers and designers as well.

7 Open core, is one of the first OSH projects and is strictly related with informatics, aiming to share designs of CPU's. Currently a similar very promising project is the Novena laptop, that propose a fully open hardware laptop.

0.2 SIGNIFICANCE

Why licenses matter?

As the free/libre and open source development model is shifting from software to hardware, some doubts and uncertentyes are being raised about how to protect inventor and hardware creators rights, as well as the right of contributors and users. The current OSH legal framework is based on copyright which is very good for protecting the plans, schematics, or source files of the projects and to grant authors the freedom to manage his exclusive rights.

Therefore the inventors or authors of any OSH project and work have the control over the schematics and the documentation. Creators can release these “source files” enabling them to prevent third parties from closing them through the use of a copyleft or share a like clause in the license. A copyleft clause implies that the work is four freedom compliant, but also require all modified and extended versions of the work to be licensed under the same or compatible terms, this mechanism ensure that a project always ramain free and all contribution are shared back with the community.[8] If using a Creative Common license is also possible for creators to include very restrictive terms such as a non-commercial clause in the license, this should enable them to prevent third parties to use such “source files” for their commercial benefit.

Yet OSH creators have no control over manufactured devices since when the documentation or “source file” is manufactured or “compiled” leading to the production of a physical object, the original creator has no right over the physical product developed from their “source files”.

In fact a manufactured physical device is likely to be easily reversed engendered in comparison whit software, once is is clear how a determined piece of hardware works, if a user wants to use it but not comply with any restrictive clause in the license, he can just re-implement the protected work creating an independent work whit the same functionality, doing so he do not violate copyright since this instrument protect just the expression of a work not the idea behind it.

Therefore this new physical object can be used for commercial purposes and can circulate without a

[8] What is Copyleft? available at: https://www.gnu.org/copyleft/copyleft.en.html

link to the documentation, even if the "source file" was licensed with a non-commercial or copyleft license. A very clear practical example is RepRap and MakerBot case, RepRap project is started in 2005 by releasing all documentation under a copyleft license, thanks to this project a huge amount of three-dimensional printing technology was released and many printers were built. Some members of this project founded the three-dimensional printers company MakerBot Industries in 2009, for the firs two generation of printer they complied with the copyleft restriction, releasing the documentation of their printers and sharing improvements back with the community, during this period MakerBot grew from a start-up to a multi million dollar company, and in 2012 when releasing their new printer, the Replicator Two none of the physical machine designed nor the graphical user interface (GUI) was shared[9], arousing harsh critics from RepRap comunity[10].

There are current debates whether a Hardware specific license is needed. There is controversy if such a license should include copyleft, weak-copyleft or whether it should be a permissive license. Learning the lessons that open source software licensing taught, what should be avoided at all costs is license proliferation, since it creates problems of compatibility and it weakens legal certainty. However, there is definitely space for more than one license since the array possible subjects matters of such licenses is very wide.[11] Therefore project strongly community based will probably be best fitted by a strong copyleft license, since non-profit distributed and participatory community are more comfortable with this kind of license. On the other hand, corporate founded projects might opt for a permissive license.[12]

Having a clear and well established OSH legal framework is important due to the great potential that such developing models have, in fact OSH potential still is not being used enough due to the

---

9 Bre Pettis, September 24, 2012 Let's try that again, Available at: www.makerbot.com/blog/2012/09/24/lets-try-that-again/

10 Josef Prusa, Open Hardware meaning, *Thursday, September 20th, 2012, Available at:* http://josefprusa.cz/open-hardware-meaning/ See also: Zach Hoeken, MakerBot vs. Open Source – A Founder Perspective, September 21, 2012 Available:http://www.hoektronics.com/2012/09/21/makerbot-and-open-source-a-founder-perspective/

11 Ibid. 5

12 Bonaccorsi, Andrea and Rossi Lamastra, Cristina, Licensing Schemes in the Production and Distribution of Open Source Software: An Empirical Investigation. Available at SSRN:http://ssrn.com/abstract=432641orhttp://dx.doi.org/10.2139/ssrn.432641

legal gray area which it is operating in. Legal certainty is an essential element that might contribute in unleashing OSH potential, from an innovative as well as an economical perspective.

Why does OSH matter?

Another factor which is worth considering when evaluating the relevance of OSH, and why it deserves attention are the ethical and social issues surrounding it. Electronic devices are increasingly more important in our daily routine and they influence users lives and decisions, yet users are mostly not in control of their devices since their design as well as their software are not shared and publicly accessible. To be in control of your device entails being able to have access to the four freedoms (infra 1.2.1), as well as access to hardware components to verify that no malicious hardware has been assembled, to be able to repair and improve you device but more in general to freely exercise the four freedoms on your hardware device.[13]

Hence OSH, would enhance awareness of how products are made and empower users in a more free and autonomous decision making process. An example of how hardware devices can embed policy decisions is clearly stated by A. Mellis when he describe how the output of airport security scanners decide on whether you will be searched or not. Should the design of this machine not stay in public hands in order for the travelers to evaluate whether the parameters are appropriate?[14] In this example some security issues are involved, but on the opposite it could be argued that is widely documented how security by obscurity in cyber security context is not a valid solution[15], this con be true in the off line world as well. Electronic devices are constantly performing many tasks highly relevant in our lives. With the IoT becoming a reality, machines will increasingly become more and more involved in our daily life and they will take or at list influence our decisions, in order to preserve our autonomy and freedom it is quite relevant to stimulate openness in hardware as well as

[13] NSA Chips, Radiowaves In Nearly 100,000 Foreign Computers: Report, 03/17/2014 Available at: http://www.huffingtonpost.ca/2014/0/15/nsa-chips-foreign-computers_n_4601797.html Accessed 17/5/2015

[14] David A. Mellis, "The moral imperative for open-source hardware", 07 OCTOBER 2010 Reachable at:http://dam.mellis.org/2010/10/the_moral_imperative_for_open-source_hardware/#comments1

[15] Alan Wlasuk, Security Through Obscurity? Don't Count On It., Available at: www.securityweek.com/security-through-obscurity-dont-count-it

in software since this can enable transparency, foster interoperability[16], avoid vendor lock-in and allow users to control their machines and not the other way around.

As FLOSS is a credible and transparent alternative to proprietary software, OSH could become a new model for distributed hardware development and distribution. Such a model could potentially apply to all types of hardware, but as I will highlight in a subsequent session (Infra 1.3) OSH model is particularly fit for hardware that gets "compiled" from a "source file" by automatic means, this include three-dimensional printable hardware such as printed circuit boards which are necessary components of the future market of IoT devices.[17]

A final but rather important reason to promote open hardware licensing by means of securing legal certainty, is that FLOSS as well as OSH fosters innovation. It is now well- demonstrated that inventors, programmers and coders are willing to contribute to open source projects for no reward other than that of making a tool more useful. Open sourcing a project works, numerous studies demonstrate how FLOSS is often more reliable, secure, scalable, performant and cost effective than proprietary software[18] and this can apply to hardware as well[19]. The more people that can contribute and get involved into a given work or project, the more value that work is likely to have, because of consequential extended testing, bug reporting, community support, contributions and therefore faster evolution and growth of the software. [20]

---

16 ALIPRANDI, S.. Interoperability and open standards: the key to a real openness. International Free and Open Source Software Law Review, North America, 3, sep. 2011. Available at: <http://www.ifosslr.org/ifosslr/article/view/53>.

17 Open hardware OS "RIOT" aims to be an efficient OS for small embedded devices meant to be incorporate in IoT devices. Available at: http://www.riot-os.org/#usage

18 David A. WheelerWhy Open Source Software / Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers!, Available: www.dwheeler.com/oss_fs_why.html

19 Jon Brodkin, Arduino creator explains why open source matters in hardware, too, Available at: arstechnica.com/information-technology/2013/10/arduino-creator-explains-why-open-source-matters-in-hardware-too/

20 Andrew M. St. Laurent, Open Source and Free Software Licensing, O' really, 2005

0.3 METHODOLOGY

In order to conduct my study I will primary use direct sources by looking at the different OSH projects, and at the different OSH licenses. I will also use secondary sources from OSH scholars and associations.

0.4 OVERVIEW OF CHAPTERS

In order to answer the central question on what is the legal instrument that best protects OSH, the thesis is divided into three chapters.

The first chapter will give a definition of OSH. It will explain what FLOSS is, how FLOSS principles apply to hardware, and why OSH is different from FLOSS. The second chapter will examine the licenses that are currently being used for OSH. It will evaluate the permissive versus copyleft and weak-copyleft approach by assessing the limitation of this licenses in hardware context. The last chapter will evaluate other legal instruments or approaches that can be used to protect hardware. They include prior art, defensive publications; trademarks, logos, certification and standards, and patents.

Chapter 1 What is OSH?

1.1 WHAT IS FLOSS?

To fully understand what OSH means, it is important to know where it originates, to this end in the present chapter two FLOSS definitions will be presented.

FLOSS means Free Libre and Open Source Software, it is an acronym that refers to software whose source code is available for users to use, study, share and improve, but that is not all.

Philosophically and ideologically the open source community differs from the free and libre one. It is not the case to go in depth in the dissertation of this argument, to the end of the discussion here is just relevant to note that the two communities split up in the end of the nineties, that the two communities share many of their core values, namely the four freedoms, but decided to divided in order to preserve a different identity. The free and libre one is linked to a more idealistic and romantic view of freedom in software as well as in other areas of life, making freedom a key component to be protected at any cost. In the effort to spread freedom in software as much as possile Free Software Foundation (FSF) suggest to use copyleft licenses, since this mechanism triggering a virtuous cyrcle, that as in the case of the linux kernel can over the years evolve in very complex products developed by many diferent actors in many differnt branches. The virtuos cyrcle might be favoured by the use of copyleft licenses this mechanism ensure that the product can not be closed. The founder of FSF, R. Stallman argue, that non free software, also known as proprietary software, is immoral since it is beyond the control of the user, making the user controlled by the software or by who control the software.

The Open Source Initiative(OSI) was set up as a marketing campaign for Free Software in 1998,[21] it also was conceived as a general educational and advocacy organization. OSI creation was prompted

[21] Björn Schießle, “Free Software, Open Source, FOSS, FLOSS - same same but different”, 2012, Available at: http://fsfe.org/freesoftware/basics/comparison.en.html

by the ground braking decision of Netscape to go open source, during this time a more business oriented view was emerging for in FLOSS world, OSI aims to endorse this view, sad with the words of a well known FLOSS scholar and entrepreneur:

"open source dump the moralizing and confrontational attitude that had been associated with 'free software' in the past and sell the idea strictly on the same pragmatic, business-case grounds that had motivated Netscape."[22]

Over time OSI evolved in a very proactive community which is more linked to a practical, convenience based motivation to keep the code open, for the aforementioned reason OSI is not strongly copyleft oriented, the use of a permissive license can be appropriate or even preferred depending on the business model and the kind of project that developers choose to endorse. In fact project such as Apache web server software (half of the web pages that we daily visit are on-line thanks to this software[23]), decided to use a permissive license, this does not make Apache a non-free software, it just allow third party to incorporate Apache into a proprietary product. Such a strategy can foster a wider adoption of the software since businesses and developers are not afraid to incorporate the software because they do not have to comply with any copyleft restriction, but at the same time it can discourage some contributors because they might feel their contribution are being used by others that will not share back their contributions.[24]

One of the main point of disagreement between the two communities was about using the word "free" since in English "free" can mean "gratis" as well as "freedom", this might mislead some people, and is not really attractive to corporate users. In practice, the debate is only based on political and philosophical issues, since nearly all free software would qualify as open source and

[22] History of the OSI, available at: opensource.org/history
[23] Usage of web servers for websites, Available at: w3techs.com/technologies/overview/web_server/all
[24] An example of how a community can fork because of contributions not being shared back whit the community is clearly reppresented by the Bacula-Bareos case, available at: http://www.bareos.org/en/faq/items/why_fork.html

vice versa.[25]

1.1.1 The free software definition

The free software definition was written by Richard Stallman almost three decades ago it has been improved along the way and currently it is composed of the four freedoms and a few sentences that explain how free software is a matter of freedom and not price:

"Free software" means software that respects users' freedom and community. Roughly, it means that the users have the freedom to run, copy, distribute, study, change and improve the software. Thus, "free software" is a matter of liberty, not price. To understand the concept, you should think of "free" as in "free speech," not as in "free beer".[26]

In the beginning freedoms where just two, freedom to copy/redistribute and freedom to change the software. Freedom to study the software was added in 1996[27] and finally also the basic freedom of Running the software was added as a separate point so that at present time the definition consist in four points ranging from null to three.

In current free software definition (version 1.135) the four freedoms are individually defined:

(freedom 0) The freedom to run the program as you wish, for any purpose.

(freedom 1) The freedom to study how the program works, and change it so it does your computing as you wish . Access to the source code is a precondition for this.

(freedom 2)The freedom to redistribute copies so you can help your neighbor.

(freedom 3)The freedom to distribute copies of your modified versions to others . By doing this you

[25] Why Open Source misses the point of Free Software, Richard Stallman, 26/11/2013. Reachable at: http://www.gnu.org/philosophy/open-source-misses-the-point.html

[26] What is free software? (The Free Software Definition),Reachable at:http://www.gnu.org/philosophy/free-sw.html

[27] What is free software? (1996 version), reachable at: http://www.ru.jnpcs.org/usoft/WWW/www_gnu.org/philosophy/free-sw.html

can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this. [28]

Next is stated that if a program falls short on one of the four freedom, than that program is proprietary. Proprietary software is defined as a an instrument of unjust power because the program controls the users, and the developer controls the program, this give developers precisely an unjust power. FSF's radical approach emerge when distinguishing various degrees nonfree distribution schemes they label all software that fall short on any of the four freedoms as equally unethical.

The definition goes on clarifyng certain points that have emerged douring the years, interesting is when commenting the freedom to distribute copies:

"includes the freedom to release your modified versions as free software. A free license may also permit other ways of releasing them; in other words, it does not have to be a copyleft license. However, a license that requires modified versions to be nonfree does not qualify as a free license.

In order for these freedoms to be real, they must be permanent and irrevocable as long as you do nothing wrong..."[29]

Here clearly emerge FSF's preference for copyleft licenses, in fact General Public License (GPL), which is the most authoritative and widely used FLOSS license issued by FSF is a strong copyleft license.

1.1.2 The open source definition

The open source definition was issued by the Open Source Initiative and it was written by Bruce Perence in 1998. This definition is strongly influenced by the Debian Free Software Guidelines (DFSG) in fact it mainly consists of DFSG but the term *free software* is replaced with *open source*.

[28] This are the four freedoms as from revision 1.135, Fri May 30 10:38:49 2014 available at http://web.cvs.savannah.gnu.org/viewvc/www/philosophy/free-sw.html?root=www&r1=1.134&r2=1.135

[29] Ibid. 25

Currently DFSG are part of *Debian Social Contract* which was also drafted by Perence together with other Debian developers DFSG were published just one year before the publication of Open Source Definition. [30]

The Debian project is the third officially recognized authority in the FLOSS movement, together with the Free Software Foundation and Open Source Initiative. Debian project aims to create a completely free and community based GNU/Linux distribution.

To understand where OSH concept comes from, it is important to focus on the open source definition because it was later reused in many different projects, including Open Source Hardware Initiative. The open source definition is opened by an introduction which clarify that despite the name open source means more that just access to source code:

"Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:"[31]

The criteria are ten and they deal with: free redistribution, source code, derived works, integrity of the author source code, no discrimination against person, groups or fields of endeavor and license issues.

A more detailed analysis of those requirements is useful, since open source definition was later reused in Open Source Hardware Initiative as a working definition for open source hardware.

Requirements that deserve attention are:

The free redistribution principle means that open source licenses must permit a non-exclusive commercial exploitation of the licensed work . Such a requirement embodies the fundamental character of open source licensing, which aims to deny anybody the right to exclusively exploit a work.[32]

---

30 Debian Social Contract version 1.1, 2004 Reachable at: http://www.debian.org/social_contract#guidelines

31 The Open Source Definition, Reachable at: http://opensource.org/docs/osd

32 1. Free Redistribution Available at: opensource.org/osd

The second paragraph of OSI definition is titled source code, it explains that in order to make freedom of modification effective, access to the source code is a must-have feature. Source code can be made available together with the compiled binaries, or it might be distributed separately. In both cases a link to source code have to be provided with compiled software.[33]

The paragraph titled “derived works” describes the open modification principle, which imply that licenses must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software. This is another of the core principles pertaining to open source licensing. The paragraph explicitly allows the redistribution under the same license, making copyleft restrictions OSI compliant, but does not make the adoption of such restrictions mandatory.[34]

Integrity of the author source code is an optional part of the definition, its author centered in the sense that aims to protect the authors original work integrity and his reputation. The examined paragraph states that licenses might limit the open modification principle, requiring users to redistribute modified source code and incorporate their modification as “patch” files. This is to protect the creator's reputation, but it must be noted that such a limitation only refers to the source file and not to binaries. In order to protect the work in compiled form the paragraph states that the license may require a distinct name or number for the original software[35].

The no discrimination provisions prohibit discrimination against person or groups as well as against fields of endeavor. The first provision was drafted to address concerns referring to possible misuse of the software created by groups with different ideologies, e.g. use of a analytical software coded by antifascist's NGO by a far right political party. The second prohibition is mainly directed rule out the licenses that impose non commercial restrictions[36]. Such a limitation is against the principles and values of open source, because would place additional restrictions that would violate, by

[33] 2. Source Code Available at: opensource.org/osd

[34] 4. Integrity of The Author's Source Code Available at: opensource.org/osd

[35] For a more detailed description, of the relevance of the project name see the paragraph on trademark (supra 3.1)

[36] Art 6 OSD, “Rationale**:** The major intention of this clause is to prohibit license traps that prevent open source from being used commercially. We want commercial users to join our community, not feel excluded from it.” available at: http://opensource.org/osd-annotated

neglecting the freedom to economically exploit the work, the core freedom to share and to use the work.[37]

What was previously defined as “Licensing issues” is a set of four provisions that require open source licenses to be generally and equally enforceable against all users.[38] The second of this four requirement is that licenses must not be specifically drafted to fit the need of a specific product, this is the case of a product made of a set of programs and this provision aims to avoid that individual parts of an aggregation of software would be distributed under a different license if not bundled in the aggregate package. Another requirement for a license to be open source compliant is that the license must not restrict other software , e.g. the license must not insist that all other programs distributed on the same medium must be open-source software . The last requirement is for licenses to be technology neutral, in this case it means that if a license requires a mouse click or other exclusively digital activities to be accepted it might not be accepted on paper as that might undermine technological neutrality.[39]

[37] 5. No Discrimination Against Persons or Groups, 6. No Discrimination Against Fields of Endeavor. Available at: opensource.org/osd

[38] 7. Distribution of License, 8. License Must Not Be Specific to a Product, 9. License Must Not Restrict Other Software and 10. License Must Be Technology-Neutral Available at: opensource.org/osd

[39] Ibid. 5, see http://opensource.org/osd-annotated

1.2 WHAT IS OSH?

1.2.1 Introduction

Hardware intended as physical artifacts, technical machines and tools exists since the beginning of civilization, knowledge on how to build it was variably free and public, secret or patented.

OSH as we intend it today first emerged from the sixties counterculture in the USA, when sharing was part of the "revolution", computers and networked communities were just starting and the first computer hobbyists groups were sharing design of their machines[40]. As microchips and PCB got more and more complex the control over computer-hardware design become more and more in the hand of corporations that kept it secret or monopolized by patenting it.

In the last decade a new wave of openness in hardware emerged, thanks to pioneering projects such as the Arduino board or RepRap printers, these projects backed-up by some PCB and electronics manufacturers such as Adafruit and Sparkfun industries, created a very big market of OSH devices that hobbyist around the world are using to create and share tools and objects that range from printers, to weather stations, drones, robots and many more things.

In order to understand what is currently meant by OSH the definition given by two of the most representative organizations are described below.

1.2.2 Opens Source Hardware and Design Alliance (OHANDA) definition.

OHANDA is an international initiative which started in 2009 with the aim to foster sustainable sharing of open hardware and design. OHANDA is also a label, designed in the sense of a non-

[40] This is the time of Homebrew Computer Club and Tom Swift Computer Terminal

registered trademark and intended to mark OSH devices and designs. This label is connected with the four freedoms from the free software definition, redrafted for hardware. Such a label if applied on the physical OSH device could serve as a physical link to the documentation, since on the OHANDA web site are hosted projects explanations furnished with documentation. Therefore if you see an OHANDA labeled machine and you want to build it, or even just study how it works, accessing the OHANDA web site and browsing it will allow you to access the desired documentation.
The definition that OHANDA adopts aims to determine what can be considered OSH and what is not.

OSH is any device that respects the four freedoms, namely: freedom to use, freedom to study and change, freedom of manufacturing, freedom to improve and release the improved device to the public.[41]

The strategy used by this project in order to define OSH is to modify FSF free software definition replacing the term "program" with "device / design".
Such an ambitious project surely deserved more attention unfortunately the mailing list is currently silent and last submitted OSH devices date back to 2012 fortunately another international project is emerging as a reference association for OSH developers, the Open Source Hardware Association.

1.2.3 Open Source Hardware Association (OSHWA) definition

The Open Source Hardware Association (OSHWA) is one of the most significant associations for OSH. The OSHWA project which started in 2010 aims to organize and structure OSH movements around common values and principles, educate the public about OSH, it aims to be the voice of the

[41] The 4 freedoms redesigned for hardware are available at: http://www.ohanda.org/

OSH community, and propose itself as a hub for the OSH community to meet, enabling different stakeholder, such as makers, entrepreneur, and corporate stuff to more effectively communicate.[42] OSHWA released a OSH definition, officially named “Open Source Hardware (OSHW) Definition 1.0”.

The core of the definition is preceded by a statement of principles. The preamble to the definition and the definition itself were first drafted by more than three hundred contributors in the Freedom Defined wiki, the definitions aim is to set up a few general requirements for the source of an hardware project to be “open”.

Firstly, it is required for the designs to be made publicly available, as this is a necessary condition for enforcing the five freedoms that are to be granted to the public when releasing any OSH device, these freedoms are:

“freedom to study, modify, distribute, make, and sell the design or hardware based on that design.”[43]

The reference to the free software definition's four freedoms is clear, but these five freedoms are copied and pasted from the freedomdefined.org project and are part of the free cultural work definition. It is to be noted that freedom to sell implies non-commercial restrictions incompatible with the open source hardware definition.[44]

The non-binding preamble continues by setting an interoperability condition requiring designs to be available in the preferred format to make modification. Subsequently it is promoted the idea of effective openness by emphasizing the relevance of interoperability and open standards, for the

[42] OSHWA web page “About” and “Board and http://www.oshwa.org/about/

[43] The definition of Free Cultural Works, is hosted on and available at: http://freedomdefined.org/Definition The definition was initiated by Erik Möller as a means to resolve ambiguity about the phrase "free content" in the context of the Wikimedia project family. It was inspired by the Free Software Definition. Helpful feedback was provided during the initial authoring process (in this chronological order) by Richard Stallman of the Free Software Foundation, Lawrence Lessig of Creative Commons, and Angela Beesley, board member of the Wikimedia Foundation.

[44] Ibid. 36

production and the making of OSH devices and related documentation:

"Ideally, open source hardware will use readily-available components and materials, standard processes, open infrastructure, unrestricted content, and open-source design tools to maximize the ability of individuals to make and use hardware."[45]

Finally, the statement of principles is closed by stating the relevance and aims of OSHW:
"aim of OSHW is to give people the freedom to control their technology while sharing knowledge and encouraging commerce through the exchange of the designs."[46]

After the "statement of principles" comes the definition itself, this definition is based on the open source definition that in turn is based on the DFSG. The definition V 1.0 starts with an introduction, which set boundaries of what can be defined OSHW. According to such a definition OSHW is a term that can be used for tangible artifacts (machines, devices or other physical things), whose design is made public so that anyone can make, modify, distribute and use those things. Here a reference to the freedoms embedded in open source hardware concept is stressed again.
The second paragraph of the introduction is of particular relevance since it states that hardware is different from software, because physical resources must always be committed for the creation of physical goods. This is a key difference that will be further examined in subsequent chapters of this thesis, the relevance of differences, and specially the fact that hardware opposed to software needs physical and "scarce resources" while in the software dimension scarcity is way less a problem, this imply legal and economical considerations(Supra 1.3.2, 1.3.3). The closing provision of the introduction consist of an exclusion of warranty and liability of the original designer, as well as the prohibition to use original designer's trademarks.

---

45 Open Source Hardware (OSHW) Statement of Principles 1.0 Available at: www.oshwa.org/definition/
46 Ibid. 21

The OSHW definition is based on twelve criteria which are: documentation, scope, necessary software, derived works, free redistribution, attribution, no discrimination against persons or groups and against fields or endeavor, distribution of license and other licensing issues.[47]

Of particular interest as it poses legal and practical challenges is requirement number four, titled "derived work" as the OSI definition which allows redistribution under the same license so it does OSHAWA definition which explicitly allows OSH to be licensed under copyleft licenses. As we will see later such a possibility is not as straight forward as it seems when it comes to enforcing it, since once an hardware device is manufactured it is hard to protect my means of copyright. Subsequent requirements rule out from OSH licenses non commercial clauses, such a firm stand is motivated by the fact that non commercial restriction are in the opinion of many too restrictive to fit the four essential freedoms. In addition to that it is worth considering that manufacturing hardware involve costs for material and tools, and the boundaries of what can be considered commercial and what not are sometimes fuzzy[48]. Attribution requirement also poses some enforceability challenges when it states that:

"license may require derived documents, and copyright notices associated with devices, to provide attribution to the licensor when distributing design files, manufactured products, and/or derivatives thereof."

Enforceability might be a problem when it comes to attribute a derived work or a copy of an original work when this are manufactured reversing engineering physical version of the original work. In fact the author of such new work is not bound to the license that was referring to documentation, ignorance can also be a cause of not compliance with a OSH license attribution requirement, in fact a physical good can be sold and when transferring the ownership not always

47 The OSHW Statement of Principles 1.0, and the OSHW definition 1.0, are available at http://www.oshwa.org/definition/

48 Leonidobusch, "Standardizing via Polling? Creative Commons' Study on Its Noncommercial-Clause", September 16, 2009, and Defining "Noncommercial"A Study of How the Online PopulationUnderstands "Noncommercial Use"September 2009. Available at: http://governancexborders.com/2009/09/16/standardizing-via-polling-creative-commons%E2%80%99-study-on-its-noncommercial-clause/ and http://wiki.creativecommons.org/Defining_Noncommercial

“links” to license, documentation and attribution are transferred.

OSHW definition is closed by an afterword, or an “openness clause”, here is recognized that open source hardware definition is just one way of sharing information, and knowledge and signatories of OSHW definition commit themselves in promoting and supporting all forms of openness and collaboration, whether or not they fit the OSHW definition. In fact OSHAWA is just one of the many stars in the galaxy of free and open knowledge sharing institution that populate the Internet, eg. A creative common licensed hardware project that adopts no derivatives or non-commercial clauses does not fit the definition but still is a form of knowledge sharing, because it grants you the freedom to use and study the work.

1.2.3 Working definition of OSH

OHANDA and OSHWA are two open hardware initiatives both have as main objective the promotion of openness in hardware but their approach greatly differs. Both projects in order to deal with OSH have to define it. The two initiatives define OSH differently:

OHANDA adopts a more general definition, which is clearly inspired by the Free Software Foundation's Free Software definition. OSHWA adopts a more detailed definition which includes the four freedoms but also many of the compliance criteria set in the open source definition. These definitions have both come from the software world, and have been adapted to the hardware world. Such a direct adaptation is convenient because of the solid legal roots of the two original software texts, but they also present some challenges because the object of the definition is different and some concepts, such as copyleft and attribution clauses work effectively with copyright licensing but are not designed to manage intellectual property on functional artifacts.

We can say that if an object is released to the public, the design files and the product itself are intended to respect either the four freedoms (from OSHWA) or the twelve criteria (from OSHWA), then this piece of hardware will be defined as OSH. In the words of OSHWA, extended definition:

"Free Libre and Open Source Hardware is a term for tangible artifacts — machines, devices, or other physical things — whose design has been released to the public in such a way that anyone can make, modify, distribute, and use those things."

1.3 HOW IS HARDWARE DIFFERENT FROM SOFTWARE

1.3.1 Introduction.

Hardware is defined as “major items of equipment or their components used for a particular purpose” or more specifically it can mean computer hardware, this is the collection of physical elements that constitutes a computer system. [49]

The totality of hardware intended as physical artifact, is quite more wide than the totality of software currently available, this also because hardware has existed since the beginning of civilization while software intended as non-tangible computer component, was introduced during the last half century. The vast array of objects that fall within the definition of hardware entails that some of them are more fit than other to be subjected to copyright protection.

When we talk about hardware we mean a very wide set of things, that vary in consistency, size, material, manufacturing process, etc. Laws surrounding such artifacts specify further requirements that have to be met, in fact legislation is custom made to make certain types of hardware fit for use in society, e.g. If a subject wants to manufacture a car, this has to be in line with all the requirements that are imposed by national motor-vehicle authority. [50]

It may be often more accurate to speak of open mechanical engineering, open chip design, open connection standards, etc. Lumping all these different things together under the "open hardware" moniker, when they have different technologies, different economics, different motives for participation, etc., may distort our legal thinking.[51]

[49] Oxford Dictionary definition.

[50] Codice della strada, Decreto Legislativo 30 aprile 1992 n. 285, Titolo III, in Italy: http://www.altalex.com/index.php?idnot=34124#titolo
Vehicle approval legislation in UK https://www.gov.uk/vehicle-approval/overview

[51] From a blo post by Luis Villa, Wikimedia Deputy General Counsel, available at:

When it comes to comparing FLOSS and OSH it might be convenient to divide hardware into two categories. The first category consist of hardware whose assembly process is automated and is executed on the basis of a digital file, such tools are somehow closely dependent and related with software. In fact there is an analogy between source code and schematics or cad files, as well as correspondence can be found between between compiling a binary file and printing a cup, an integrated circuit or running a laser cutter on the base of a digital source file. In other the words, products created through automated manufacturing process are more likely to be covered by copyright (Supra 2.5.4).

Example of this type of hardware is the one which gets printed, it includes 3D printable objects as well as printed circuit boards but also some projects built on material such as wood[52] or PVC. Laser cutting machines are used to shape the physical artifact that gets automatically manufactured from a source computer aided design (CAD) file. The kind of hardware which can be printed is the one that more closely resembles some of the characteristics that are typical of software, such as, direct creation of the product from a digital "source file". Obviously a few differences remain since in the case of a physical product physical raw material and a physical manufacturing machine is needed. Examples of this kind of hardware are all OSH designs released with STereo Lithography (STL) interface format extension, and hosted on collaborative repository such as thingiverse.com, github.com, cubehero.com or similar platforms.

The other category of hardware consist of those physical artifacts whose assemblage or building process which leads to the realization of the product is done by human labor, possibly using mainly standard tools and materials (supra 1.1.2).[53]

---

http://opensource.com/law/14/5/legal-issues-open-data-open-hardware

[52] Open source beehive, use a file to cut the beehive from a single piece of wood of 1.5 x 2m Available at: http://www.opensourcebeehives.net/

[53] Examples of this category are project such as open source hardware windmill Ventolone http://www.openventolone.com/ or bike battery charging "Rowan's portable pedal power generator" http://www.appropedia.org/Rowan%27s_portable_pedal_power_generator

With the aim of understanding which are the differences in development and realization between FLOSS and OSH differnces will be listed in the next session, this order to draw a comparative picture highlighting differences that will be pointed out using some fixed parameters also denominated dimensions. These differences are important since the IP approach to OSH, is heavily influenced and inspired by FLOSS. To assess which are the differences between FLOSS and different kinds of OSH will help in defining which legal framework is more suitable for OSH. A comparative view between software and different kinds of hardware can also help to evaluate if keeping the "source" open and sharing it, can bring in the hardware context a similar degree of innovation as FLOSS did in software context.

In order to point out the differences between FLOSS and OSH, this chapter is divided into two sections, the first will outline differences regarding development, sharing and improvement processes of automated manufactured OSH, traditional OSH and FLOSS. This three categories will be compared considering the following parameters: Spatial, Temporal, Social, Economical and Evolutionary. In the second section of the the intellectual property and legal instruments designed to protect software and hardware will be evaluated in the specific context of OSH and FLOSS.

1.3.2 Section one: Empirical differences

The first parameter is worth to evaluate is "space", it is possible to affirm that when developing FLOSS physical and virtual space are not a constraint, since there is no scarcity of virtual space that is the one needed. When developing objects created by means of automated manufacturing process, than some physical space is required this will be the space that the machine takes that can variably be few centimeters if we consider a desktop 3D printer to several meters for more complex CNC machines.[54] Traditional hardware development require physical space, it depend of the device we are working on but most of the time different tools and materials, will be required this inevitably

[54] A small desktop 3D printer can be a model Prusa or a makerbot Replicator, while bigger CNC machines can occupy up tu 4 meters in length such as the haascnc HS-3R or even more.

require space.

Time is another parameter that is involved in the development of any project. Developing software require time, but independent and distributed development allows simultaneous participation and self organization enhancing the productivity per peer ratio, and allowing an indefinite number of developers to work on the same code at the same time. When developing hardware produced by automatic means source files can be shared and improved at software time rate, while actual manufacture and testing of prototypes might require longer time since it needs to be assembled. Longer time is needed when developing handcrafted hardware since development is limited by access to physical resources and space but also to the time required for actual construction, and prototyping.

Another important difference is related to economical resources committed to the realization of a project, one of the main areas where software and hardware differs is exactly the economical one, while in the physical world resources are scarce raw materials are costly, making copies involves raw materials which are of great expense as well as tools. In a virtual environment there is no economic cost of raw materials and making copies, a part from electricity needed to run personal computers and servers.

A last parameter that can be considered is the evolution of a project, that involve prototyping, improvements and newer version of the product. In software development "given enough eyeballs, all bugs are shallow"[55], this means that if there are enough people revising code, bugs in the code are quickly noticed and solved, and software release and upgrade life cycle can proceed fast. With hardware manufactured by automated means, version improvement and fixes to the designs can be done remotely from an indefinite number of contributors if the designs and source files are in digital format on a shared repository, but to make effective those fixes the object need to be manufactured, this involve time and costs slowing down the process. When building traditional hardware, in order to make fixes is necessary to work on the finished machine or prototype, and distributed flaw

[55] "given enough eyeballs, all bugs are shallow." is known as Linus' Law, from Linus Torvalds

reporting is limited by actual testing. Production of new version and involve high costs, time and specific expertize, making the all process much slower, less cost effective, and less participatory. [56]

1.3.3 Section two: Different goods different legal mechanism.

Intellectual property rights or Imposed monopoly privileges restrictions, as some proposed to call this set of different legal mechanisms[57] are used to protect exclusives rights connected to software and hardware development, the most commonly used legal mechanisms are: copyright, patents, trademarks and sui generis design rights.

1.3.3.1 Intellectual Property

The term intellectual property, first appear in an international agreement in the 1883 Paris Convention on Industrial Property,[58] but is only after the 1967 foundation of the World Intellectual Property Organization (WIPO) that the term acquire a general and widespread meaning. Some critics argue that such a name is misleading since property is a right in rem which derives from ancient roman law and imply complete ownership with possession, the difference with intellectual property is easy to spot if thinking of a car and the content of a book:

if someone use my(private property) car, I can not use it until I regain possession of it" on the contrary "if someone read my (copyrighted) book memorize it and make a copy of it, than it is violating my intellectual property but the book is still in my possession[59]

---

56 MALINEN, Tiina et al. Community created open source hardware: A case study of "eCars - Now!". **First Monday**, [S.l.], apr. 2011. ISSN 13960466. Available at: <http://firstmonday.org/ojs/index.php/fm/article/view/3357/2951>. Date accessed: 16 May. 2014. doi:10.5210/fm.v16i5.3357.

57 Richard M. Stallman, Did You Say "Intellectual Property"? It's a Seductive Mirage, Available at: www.gnu.org/philosophy/not-ipr.xhtml

58 Paris Conventionfor the Protection of Industrial Property, art. 13 Available at: www.wipo.int/treaties/en/text.jsp?file_id=288514

59 Cory Doctorow, "Intellectual property" is a silly euphemism, Available at: www.theguardian.com/technology/2008/feb/21/intellectual.property

Because of this reason the term property can be more appropriately replaced with monopoly, because intellectual property is not linked to possession.[60]

Intellectual property is divided into two categories: industrial property and copyright. Copyright, patents and trademark are different legal mechanisms that, serve different purposes, while the purpose of all this mechanisms might be in origin noble, they are sometimes abused[61].

1.3.3.2 Copyright

Copyright law is a branch of that part of the law which deals with the rights of intellectual creations. It concerns all forms and methods of public communication, printed publications, sound and video broadcasting and recording, even computerized systems and therefore software. Most works, exist only once they are embodied in a physical object, e.g. novels or movies , but physical existence is not an essential requirement since some works such as a musical composition or poems are a protected works even if they are not fixed in a tangible object, it suffices if the work is sad out loud.[62] One of the most important issues that have to be considered is that copyright covers only the form of expression of ideas, not the ideas themselves, what is here protected is the creativity in the choice and arrangement of words, numbers, colors, shapes or other. As it is suggested by the name[63] of this legal instrument assign to the author of a copyrighted work the exclusive right to "copy" it, or better to use, display, communicate to the public and distribute the work in the form in which the original work was expressed by the author. [64] Usually in Europe copyright Comes into being without formalities once the work is finished and the maker is the first rights holder if the work is the result of the author's own intellectual creation.[65] Some argue that copyright protection is a sine

---

60 by Michele Boldrin and David K. Levine, Against Intellectual Monopoly, Availble at: levine.sscnet.ucla.edu/general/intellectual/against.htm

61 Patents are meant to foster innovation, granting a monopoly to exploit a technical invention if the inventor disclose the invention, but when patents are granted for invention that are not novel or they do not involve an inventive step they can seriously hamper innovation: eg. Apple's Page touring patent Available at: http://assets.sbnation.com/assets/1751581/USD670713S1.pdf

62 This can vary across jurisdictions.

63 Online Etilmology Dictionary, http://www.etymonline.com/index.php?term=copyright

64 "WIPO Intellectual Property Handbook:Policy, Law and Use", WIPO 2004 Second Edition

65 Art. 6 Directive 98/93/EEC, of 29 October 1993 harmonizing the term of protection of copyright and certain related

qua non condition for progress, since it encourages individual creativity according exclusives rights to the author. Others on the contrary argue that creativity is an inherent need of humans; so if creativity, knowledge and in general all intellectual and artistic cration are shared, this would trigger much more innovation and grouth than if this content is loked-down through mechanisms such as copyright.[66]

Copyright is not designed for physical objects, notwithstanding it can apply also to physical artifacts if they satisfy the originality requirement[67], in fact to some extent it have to been recognized that copyright protection apply to the so called corpus *mysticum* as well as tho the *corpora mechanica*, since the work is always a *corpus mysticum*, while the exclusive rights may apply to the *corpora mechanica,* when this represent an author's own intellectual creation.

Copyright fully apply to OSH design and manufactured objects by protecting their artistic futures but not the functional aspects of the work[68], also if a physical object is manufactured from a copyrighted design the copyright is not automatically transferred to the object. In other words copyright might effectively restrict the distribution of the documentation and instructions, but not any manufactured devices built from such source files, unless the physical artifact satisfy the originality requirement[69]. E.g. building your own weather station might be convenient for many reason, let's say you want to build a MSP430-Wireless-Weather-Station originally developed by Jeremy E. Blum, this is an open source hardware and software project licensed via a Creative

---

rights.

66 Frank Thadeusz, No Copyright Law: The Real Reason for Germany's Industrial Expansion? Available at: www.spiegel.de/international/zeitgeist/no-copyright-law-the-real-reason-for-germany-s-industrial-expansion-a-710976.html

Also "During the republic's first hundred years, the U.S. was a "piratenation", with respect to foreign works of authorship" as noted by prof. R. Gorman and prof. J. Ginsburg., see also United Dictionary Co. v. Merriam Co., 208 U.S. 260 (1908). https://bulk.resource.org/courts.gov/c/US/208/208.US.260.129.html

67 Furniture made of discarded wood(Piet Hein Eek), Vrzr DC Alkmaar 20 February 2007

68 In Europe: Copyright protection is accorded to technical drownings to the extent of their artistic features, with art.17 of directive 98/71 on the legal protection of designs, it was established the cumulability of design rights and copyright protection, see also CJEU in case C-168/09.
In the US: Pictorial, graphic, and sculptural works" include two-dimensional and three- dimensional works of fine, graphic, and applied art, photographs, prints and art reproductions, maps, globes, charts, diagrams, models, and technical drawings, including architectural plans. Available at: copyright.gov/title17/92chap1.pdf
On the same stand is WIPO: http://www.wipo.int/copyright/en/#topics

69 Ibid. 30

Commons Attribution Share-Alike License and therefore it fits the OSHWA definition, all software, schematics and documentation is available on GitHub.[70] If I download the schematics and documentation and I add to the original design an anemometer, because I want to measure wind direction and speed, when I publish the modified design I have to comply with the copyleft restrictions, this means that copyright fully apply to the documentation. On the other hand if I manufacture the weather station and add to the manufactured object the anemometer when I distribute it I do not have to comply with any copyright restrictions since the manufactured object is not copyrighted. One of the issue that currently dominate the debate is, how to transfer some of the right that creators wish to retain from the source documentation to the manufactured device, in other words how to link documentation and the license agreement to the manufactured devices. While for hand assembled devices there are still many uncertainty on how to do so, for automated manufactured devices interesting proposal are putted forward. Using QR code or other linking tools embedded in the source file is one of the ideas that is currently more credited,[71] this technique together with some DRM-like[72] technology built in the printing machines in order to avoid malicious party to erase the links, would cause the printed device to have a physical link to license as well as attribution and the documentation.

Usually physical objects are crafted to serve a purpose, therefore they have a function, this function is not protected by copyright, a particular way to deploy the function that an object has can be copyrightable if it satisfy the originality requirement, e.g a cabinet have the functional purpose serving as storage for other things, if an artist or craftsman build a cabinet out of different colored discarded wood planks, since the construction methodology as well as the result is very different from the standard practice he might get copyright protection for building it in a certain way with

[70] MSP430-Wireless-Weather-Station https://github.com/sciguy14/MSP430-Wireless-Weather-Station.

[71] Eli Greenbaum, Three-Dimensional Printing and Open Source Hardware, New York University journal of intellectual property and entertainment law, 2013 Available at: http://jipel.law.nyu.edu/2013/04/three-dimensional-printing-and-open-source-hardware/

[72] digital rights/restriction management and other content access restrictions could be applied to three-dimensional printers in order to ensure that such devices only construct pre-authorized designs. Eli Greenbaum.

certain colored discarded wood boards[73], similarly some argues that because the Printed Circuit Board (PCB) is in itself an artistic work, therefore subject to copyright law.[74] In other words the subjective artistic component of the work have to be very relevant in order to accord copyright protection to a physical object and if it is accorded the protection is for the expression of the idea and not for the idea itself. This particular characteristics of copyright makes it not very fit for protecting functional object.

Another question that deserve attention is how do copyleft, attribution, non derivatives and commercial restrictive clauses apply to hardware. Some argue that such restrictions are not suitable for hardware device and their documentation, because once a device is manufactured it looses any link to copyright licensing terms and even if it comes with a copyright license it can easily be reversed engendered and therefore re-manufactured not using the restricted documentation. This is the so called "analog gap", because of the heterogeneity of hardware such issues can be valid for some hardware but not to other,

If we take the case of relatively complex PCB designs, it is quite clear that many hours are typically spent in the layout phase, and that many licensees who want to modify the layout might well take the route of abiding a copyleft clause and publishing their modified work under the same terms, just because the amount of time and effort involved in getting the ideas from the schematic and layout and doing it all from scratch might be prohibitive.[75]

1.3.3.3 Patents

Patents are granted by centralized authorities to creators of a technical invention, that comply whit the following requirements: such technical invention have to be novel, susceptible of industrial application and it must involve an inventive step.[76] Patents are expensive, this entry barrier rule out many open hardware developers which are amateurs or are not willing to go through an expensive,

[73] Ibid. 40
[74] From CERN OHL mailing list A. Kanz
[75] From CERN OHL mailing list, B. Perens.
[76] Att. 54, 56, 47 European Patent Convention

highly bureaucratic, potentially unsuccessful patent application. The good news is that one of the essential requirement for a patent to be issued is the novelty requirement. An invention is considered to be novel if it does not form part of the state of the art, the state of the art shall be held to comprise everything made available to the public by means of a written or oral description, by use, or in any other way, before the date of filing of the application.[77] This is a good news for OSH developers because if they come up with an idea and they make it public means a defensive publication, this will become prior art and at list in theory it will not be patentable. Unfortunately practice is different than theory and some technologies released to the public as OSH had being subsequently successfully patented by competitors.[78]

It is a fact that it is increasingly becoming difficult to draw a clear line between hardware and software[79], and borders between copyright and patent's subject matters are also increasingly getting fuzzy,[80] patents are granted for software and copyright is accorded to some physical creation and to source files that when "run" will generate physical artifact by automated means. Patents and copyright are two different legal mechanisms but both of them are in parallel evolving to fit free and open source business and development models. In fact as we will see, notwithstanding the more rigid framework in which patents are usually granted and managed, a new wave of patent pooling, sharing and interoperability mechanisms is emerging,[81] this new approach to industrial property through patent sharing, pooling and patent non aggression statements can be a promising way for corporate actors to be involved in OSH development and promotion.[82]

---

77 Art.45 c.1,2 European Patent Convenction

78 Julie Samuels, EFF's Fight for Open 3D Printing Continues at Ask Patents, Available at: https://www.eff.org/deeplinks/2013/03/effs-fight-open-3d-printing-continues-askpatentscom

79 Anthony Wing Kosner, Prediction 2014: Hardware Is The New Software, Available at: www.forbes.com/sites/anthonykosner/2014/01/01/prediction-2014-hardware-is-the-new-software/

80 BESSEN, James; MEURER, Michael J.. What's wrong with the patent system? Fuzzy boundaries and the patent tax. First Monday, [S.l.], jun. 2007. ISSN 13960466. Available at: <http://www.firstmonday.dk/ojs/index.php/fm/article/view/1867/1750>. Date accessed: 18 May. 2014. doi:10.5210/fm.v12i6.1867.

81 Wilbanks, J.T.; Wilbanks, T.J. Science, Open Communication and Sustainable Development. Sustainability 2010, 2, 993-1015. Available at: www.mdpi.com/2071-1050/2/4/993

82 On June 12, 2014 Tesla motors decided to not enforce his patent, generating *de facto* a huge ammoung of open knowledge, in electric car manufacturing. Available at: http://www.teslamotors.com/blog/all-our-patent-are-belong- you see also: OpenInventionNetwork is defensive patent pool and community of patent non-aggression which enables freedom of action in Linux;

1.3.3.4 Trademarks

Trademarks are an indication of the origin of a good, they already existed in the ancient world since they are closely connected with trade, even 2000 years ago craftsman were marking their creations and marking them with their signatures which serve as guarantee of the origin therefore the quality of a certain good.[83]

Trademarks are defined as:

any sign that individualizes the goods of a given enterprise and distinguishes them from the goods of its competitors.[84]

The definition point out the two fundamental aspects of this legal instrument, trademarks serve the purpose of indicating the source of a given good, but only if they allow the consumer to distinguish a product from goods manufactured from other enterprises, than the trademark fulfill his function. In the world of FLOSS and OSH trademark are of particular importance because they often serve as a certification that a given software or hardware are developed by a given community, through established practices and according to certain values. Knowing who developed a given software or hardware can help users to choose a relatively reliable and trustworthy product, this because when the source is shared and is available to anyone to make modifications, share and redistribute it, a third party may redistribute an independently modified version but if the the project name is trademaketed, the independently modified version will have to carry a different name, enabling users to ditinguish it from the original version. This mechanism is often explicitely stated in the license agreement, it serves the purpose of ensuring a stricter quality control over released versions.[85] Trademark is often in many OSH and FLOSS projects the one of the most valuable

---

83 Institute of Intellectual Property, Ch2, The History And Development Of Trademark Law Dr. Shoen Ono, Available at: www.iip.or.jp/translation/ono The History And Development Of Trademark Law

84 Ibid. 37

85 Chestek, P.. Who owns the project name?. International Free and Open Source Software Law Review, North America, 5, nov. 2013. Available at: <http://www.ifosslr.org/ifosslr/article/view/87>.

asset,[86] since the product itself or the product's specification are shared but the product name indicates the community and developers behind the product and if the community is trusted and reliable than the product will most probably have the same characteristics.

1.3.3.5 Design rights and industrial design

The first laws protecting design rights were promulgated in England, for the protection of industrial designs this was the Designing and Printing of Linens, Cotton, Calicoes and Muslins Act of 1787, which gave protection for a period of two months to:

"every person who shall invent, design and print, or cause to be invented, designed and printed, and become the Proprietor of any new and original pattern or patterns for printing Linens, Cottons, Calicoes or Muslins[87]."

The subject matter of the legal protection of industrial designs is the external or internal design which is applied to or embodied in a determined article or product. Industrial design protection serve the purpose of preventing competitors or any unauthorized third party to reproduce the protected design, but it does not prevent any party to manufacture similar articles differently designed but fulfilling the same utilitarian function. The conception or idea that constitutes the design may be expressed both in two-dimension or three-dimensionally.[88] It is usually required for according protection that a designs have to be applied to utilitarian articles, this differentiate it from copyright that is a mechanism that apply to purely aesthetic creations. On the other hand designs which are dictated exclusively by the function which the article is meant to perform shall be excluded from protection. This right is usually subject to registration, and to be granted the design have to be novel or at least original and have an individual character[89]. Such a right offer an intermediate protection between copyright and patents, since it is meant to apply to physical objects

---

86 GNOME, Legal and Trademarks Available at: www.gnome.org/foundation/legal-and-trademarks/

87 Sir Thomas Edlyne Tomlins, John Raithby, The statutes at large, of England and of Great Britain: from Magna Carta to the union of the kingdoms of Great Britain and Ireland, Volume 18, P. 328, Available on Google books

88 Ibid.37

89 Directive 98/71/EC, Art.3 p. 2

but it does not protect its functional aspects. Design rights are not particularly relevant to OSH, since they require registration and they do not protect any technical aspect of the work, the scope of this right is to have a monopoly on the specific design, while the aim of OSH developers is to share designs, also there is no danger of others "stealing" your creation since once a design is shared is not novel anymore therefore it makes others unable to register it. To the extent of enforcing restriction or copyleft rights on designs, to register a design and license it out furnished with restrictions might be a viable strategy for certain hardware categories such as composite products or products with innovative designs, some argue even PCB can be protected by such a right[90] which might fall within the definition, this states:

"'design' means the appearance of the whole or a part of a product resulting from the features of, in particular, the lines, contours, colors, shape, texture and/or materials of the product itself and/or its ornamentation."

While product id defined as:

"'product' means any industrial or handicraft item, including inter alia parts intended to be assembled into a complex product, packaging, get-up, graphic symbols and typographic typefaces, but excluding computer programs "[91]

Because of the wide subject matter and the low entry barrier[92], design right in continental European context might be a possible alternative to copyright for some OSH projects.

1.3.3.6 Integrated circuit (IC) topography right

A diplomatic conference was held at Washington, in 1989, the conference adopted a Treaty on Intellectual Property in Respect of Integrated Circuits. This is a copyright-like right that is

[90] Di Vivien Irish, Intellectual Property Rights for Engineers, 2nd Edition, p.57 Available on: Google Books
[91] Art.3, Council Regulation No. 6/2002 of 12 December 2001 on Community designs Council Regulation.
[92] Design registration costs are relatively low, on on line application can be performed for 350€.

concerned to two or three-dimensional layout or topography of an integrated circuit, which is an electronic circuit in which the elements of the circuit are integrated into a medium, and which functions as a unit. Because of their functional nature IC are not subject to copyright law , but their topography is also not a patentable subject matter therefore a sui generis right was accorded to IC topographies. PCB do not fall within the scope of this right because even if they have an electrical pattern they lack of a semiconducting layer.[93] Such a right is very interesting from a OSH perspective because it accord exclusive rights to the creator of a functional hardware that do not extend to his function but is not either solely linked to the appearance of it, unfortunately application of such right is limited by his nature of sui generis IC layout right that circumscribe the protection to a very limited set of devices.

[93] Ibid. 55 see also Directive 87/54/EEC Art. 2

Chapter 2 Copyright licensing

2.1 BACKGROUND

Software licensing could be traced back to the end of the seventies. The underlying reason for such development was the realization that software has become a viable market. Consequently, software companies regarded their source code as a secret intellectual valuable asset. As a result access to software in its human-readable source code form became more and more difficult as source code started to be considered as a valuable intellectual property asset having a monetary value.

During the the eighties[94] the first FLOSS license was drafted, this was the GPL. Since, then, the scene of FLOSS software licensing has proliferated for more than 30 years. Hundreds of different licenses have being released, leading to the well known problem of license proliferation and license compatibility.[95] Software licenses have multifaceted benefits of which two are most important. Firstly they serve the purpose of balancing the external relationship between various stakeholders, namely: developers, the community and the market. Secondly, they organize internal relationship and set hierarchy of values within a community, which might typically involve users, contributors and developers which might be corporate as well as independent individuals.

Currently, there is also a debate related to hardware licensing both at conceptual and practical levels. At a conceptual level one of the questions is related to whether there is a need for a hardware specific licensing or not. Similarly, if is assumed that there is a need for licensing which licensing model is more appropriate for OSH is also the point of contention. A few theoretical and practical

[94] Arnoud Engelfriet, A History Of Foss Law And Licensing, Available at: http://ifosslawbook.org/a-history-of-foss-law-and-licensing/

[95] Lawrence Rosen , License Proliferation , Available at: http://rosenlaw.com/pdf-files/ On a different stand: Robert W. Gomulkiewicz, Open Source License Proliferation:Helpful Diversity or Hopeless Confusion?, Available at: law.wustl.edu/Journal/30/Gomulkiewicz.pdf

proposals are also forwarded. On a theoretical side some licenses are being drafted but not used in practice to license a project; Solderpad License and Three Dimensional Printing License are some authoritative examples. OSH specific licenses used in practice to manage copyright in ongoing OSH projects are: TAPR OHL, CERN OHL, OCP[96], and some hardware version of well known non hardware specific agreements such GPL and Creative Commons.[97]

There are also empirical researches that are conducted to determine the practices of hardware licensing. For example in 2013 the OSHAWA community survey was conducted to determine what kinds of licenses have been used to release hardware source files. According to this survey, most of the respondents indicated that they released the files without any explicit license. The same survey has also indicated that the seven most used licenses are software licenses and the only hardware specific license showing up in this survey is CERN OHL. These results shows that amateur and hobbyist OSH developers are mainly inspired from the software world.

In the following sections an attempt is done to survey the most widely used licensing models. Once they are surveyed, they will also be evaluated. The licenses are classified as permissive licenses, weak-copy left licenses, strong copy left licenses and flexible creative common licenses. The permissive licenses include MIT, BSD Apache and Solderpad licenses. The weak-copy left licenses examples are MPL and CernOHL licenses. The strong copy left include Rep Rap license agreement and the GPL like OCP license agreement.[98] The "sui generis" flexible Creative Common Licenses, that allow layman licensor to draft a some right reserved customized license[99], are used by one of

[96] For TAPR OHL, TADD-1 RF Distribution Amplifier available at: http://www.tapr.org/~n8ur/TADD-1_Manual.pdf; a list of CERN licensed projects is available at: http://www.ohwr.org/projects; an overview of OCP licensed design is available at: http://www.opencompute.org/;

[97] A hardware adaptation of gpl is RepRap license available at: http://reprap.org/wiki/RepRapGPLLicence; while Creative Common licensed reference design and Eagle files are Arduino Uno design available at: http://arduino.cc/en/Main/arduinoBoardUno also Sparckfun release documentation under Creative Common BY-SA http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Dev/Arduino/Shields/Weather%20Shield.pdf and http://opensource.com/business/12/9/how-sparkfun-built-open-business

[98] A compatibility and FLOSS licensing scheme is available in David A. Wheeler, The Free-Libre / Open Source Software (FLOSS) License Slide, available at: www.dwheeler.com/essays/floss-license-slide.html

[99] More information and licenses are available at:http://creativecommons.org/about

the most successful OSH product, Arduino[100], as well as by one of the world leading semiconductor chip maker, Intel Corporation.[101] For each of the licensing model, one or more examples of an actual OSH product licensed under its terms will be provided. The aim is to give a bird's eye view of the current status in OSH licensing and evaluate both theoretical proposals as well as how they are used in practice. As well a OSH project for each of the licensing frameworks will be cited in order to evaluate from a practical stand point the degree of effectiveness of the licenses on specific products or projects, as well as how different projects choose different licenses and why.

2.2 WHICH LICENSING SCHEME GRANTS MORE FREEDOM, COPY-LEFT OR PERMISSIVE LICENSES?

As it was indicated in the background section of this chapter, there are debates related to licensing of hardware. There has been a longstanding debate among copy-left and permissive license advocates. Supporters of permissive licenses argue that when licensing under a permissive license since derived products are not bound to be licensed with the same license, they become more suitable to be incorporated in proprietary closed software. Thus, licensing imposes less restriction and therefore it grants more freedom. The copyleft restriction imposed by licenses such as the GPL imposes all subsequent developments to be released under the same or compatible license, such a mechanism is supposed to make source code openness “sticky”, or “viral”, inducing some to argue that FLOSS is a disease that attaches to everything it touches[102].

On the other side are the Free Software Foundation and other copy-left supporters who argue that the idea of freedom as a lack of restriction is naive. Freedom is granted if the four essential

[100] Ibid. 4
[101] Intel® Galileo Reference Design, can be downloaded from: https://communities.intel.com/docs/DOC-21824
[102] Silviu Stahie, Steve Ballmer's Legacy: Linux Is a Cancer, 2013.Available at: news.softpedia.com/news/Steve-Ballmer-s-Legacy-Linux-Is-a-Cancer-378948.shtml

freedoms are secured, and this can be accomplished only by accessing the source code. Hence GPLv3 and copyleft licenses, by requiring to share the source code on subsequent recipients, contributors and forks, grants more freedom than permissive licenses because it force more software to be free.[103]

It is also debated if members of a community are best encouraged to contribute by the use of a copy left or permissive licenses. It is widely acknowledged that distributed participatory projects are more successful if a copyleft clause is included in the license agreement.[104] This is because if a project is based on voluntary contributions, contributors are willing to keep the code open so that in the case of a fork (some other project taking up the source code and developing it autonomously or incorporating it in a larger work), subsequent versions, changes, and upgrades will be returned and shared back with the community. On the other side there are some examples of very successful projects that also have an active community but they are not copyleft oriented, eventually they are a minority in FLOSS landscape[105]. Observation of the practical trend indicates that projects which start as FLOSS and then go proprietary, even partly, are often creating internal frictions and misunderstandings within the community, which sometimes might drive to forks and even to legal proceedings.[106]

---

[103] Bradley M. Kuhn and Richard M. Stallman, Freedom or Power?, Available at: https://www.gnu.org/philosophy/freedom-or-power.html#f1

[104] OSCON 2013: Eileen Evans, "Licensing Models and Building an Open Source Community"

[105] See: Nicolas Suzor, What motivates free software developers to choose between copyleft and permissive licences? Available at: opensource.com/law/13/8/motivation-free-software-licensing, and Communities and Licenses: Permissive licenses vs copyleft (BSD/MIT/Apache 2.0 vs GPL), blog.technokrat.nl/?p=515 Available at: http://blog.technokrat.nl/?p=515

[106] 2013-10-11, Maik Außendorf, Why have you started a fork from bacula.org? Available at: http://www.bareos.org/en/faq/items/why_fork.html

2.3 WHY IS A HARDWARE SPECIFIC LICENSE NECESSARY?

This is a legitimate question that the skilled reader might ask, since OSH licenses mostly refer to documentation files, including but not limited to schematic diagrams, designs, circuit or circuit board layouts, mechanical drawings, blueprints and so on including all that is needed to know for an average person skilled in the art to build the object. These documents are typically shared in digital format, with the aim of licensing digital texts and documentations a few licenses are already available this licenses might be fit for the purpose of copyright or copylefting those files. Licnses fit for the purpose are the GNU Free Documentation License or as used by many relevant OSH projects the Creative Common licenses. The main concerns regarding the use of non-hardware specific licenses for OSH projects relate to language clarity, in fact many of the published licenses are clearly mirroring software licenses, but they modify definitions so that they consider hardware specificities. A software or a documentation license, although open source, don't sufficiently acknowledge the fact that tangible products would be manufactured on the basis of the licensed documentation.[107] A shortcoming of Creative Common licenses is the absence of a patent grant clause (infra 2.4.1), which might be useful when releasing or publishing software as well as hardware sources or documentation. This is why at Intel some are proposing to add in a separate document a patent grant clause for initial developers and contributors versions.[108]

[107] AYASS, M., SERRANO, J.. The CERN Open Hardware Licence. International Free and Open Source Software Law Review, North America, 4, may. 2012. Available at: <http://www.ifosslr.org/ifosslr/article/view/65>. Date accessed: 08 Jun. 2014.

[108] Proposal issued in a presentation held during the FSFE legal and licensing workshop 2014, the name and contacts of who issued the proposal will be given upon request since the workshop was covered by Chatham House Rule.

2.4 PERMISSIVE LICENSES

The family of licenses known as permissive licenses are those licenses that do not include any form of copyleft clause, allowing the recipient of the licensed item to redistribute it under any kind of license. The major restriction that this kind of license imposes is just to cite the original license. Given a proper citation and credit to the original license downstream recipients can add to it whatever other license and terms and conditions including proprietary ones.

2.4.1 Apache license:

The Apache License was first released in 2000 by the Apache Software Foundation, non-profit corporation that provides organizational, legal, and financial support for a broad range of FLOSS projects but mainly apache HTTP web server currently being one of the most used web server modular platforms. The first Apache licenses were quite similar to the very simple drafted BSD and MIT. Four years later the foundation issued a new license the so-called Apache License 2.0, this is a fuller and more complex license which expressly addresses patent rights and use of other licenses for derivative works[109], that are developed on top of an already Apache licensed work. The license starts with a long list of definitions that define source, object, work as well as derived works and contributors. It is important to notice that "You" (or "Your") shall mean any individual or Legal entity exercising permissions granted by the license, the same is true for most of OSH and FLOSS licenses, where the licensee is identified as You and the licensor as Contributor or Original Developer.

In clause number two titled grant of copyright license*:*

[109] In Art. 3 are grated patent licenses, while clauses 4 and 5 state conditions for the redistribution.

Each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.[110]

The subsequent paragraph is dedicated to the grant of a patent license, that mirroring the copyright clause grant perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable license to "You" for the making, selling, etc.

Within the patent license there is a safeguard disposition, this states that in the event that a licensee initiates patent litigation against any contributor on the basis that any part of the Work infringes on a patent, the license terminates with regard to that licensee as of the date the litigation is filed. Here the aim is to avoid a situation where a Licensee is both benefiting from the work and at the same time could file litigation against the licensor. This clause makes the Apache license incompatible with the GPLv2 license[111]. The Apache license is granted if some conditions are met. One of the conditions is the duty to inform the licensees of the terms of the license. The notable features of the license is that all disposition are accurately explained. Eg. there is an explicit paragraph that states how "You" can add your copyright notice to derivative works covering exclusively modifications or the derived work as a whole. It also explained that by default contributions are relicensed under the same terms, under which the original work was licensed, but contributors shall supersede or modify the terms by using any separate license agreement.[112]

[110] Art. 2 Apache License, Version 2.0 available at: www.apache.org/licenses/LICENSE-2.0.html

[111] The FSF has never considered the Apache License to be compatible with GPL version 2, citing the patent termination and indemnification provisions as restrictions not present in the older GPL license. Available at: https://www.apache.org/licenses/GPL-compatibility.html

[112] Art. 5 apache license 2

2.4.2 Solderpad License

Solderpad license is also theoretical and academic exercise. It was contributed by Andrew Katz, in the appendix to his "Towards a Functional License for Open Hardware" article[113]. The text of the license is now hosted on solderpad.com, from where it takes it's name. It basically consist of an adaptation of the Apache 2.0 license to hardware context.

Solderpad grants a perpetual, worldwide, non-exclusive, no-charge, royalty-free license for publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.[114]

The Work and derivative Works are the object of the license and they can be distributed in Source or Object form:

Source form shall mean the preferred form for making modifications, including but not limited to source code, net lists, board layouts, CAD files, documentation source, and configuration files.

and

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, the instantiation of a hardware design and conversions to other media types, including intermediate forms such as bytecodes, FPGA bitstreams, artwork and semiconductor topographies (mask works).[115]

The distinction between these two categories should not mislead the reader. In fact Solderpad mirrors the Apache License that similarly makes this very distinction. Owing the fact that this is a hardware oriented license the presence of the object as a covered matter can be more broadly

[113] Katz, A.. Towards a Functional Licence for Open Hardware. International Free and Open Source Software Law Review, North America, 4, may. 2012. Available at: http://www.ifosslr.org/ifosslr/article/view/69.
[114] Clause 2 Solderpad license
[115] Art. 1 Solderpad license.

interpreted. It is also worth noting that in this license, like in any other permissive licenses, there is no distinction between initial work and subsequent version or "larger work". Contributors are defined as:

the Licensor as well as any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.[116]

In fact a patent clause is included in the license likewise the Apache license this clause grants to You a patent license for the each contribution. The same clause also holds a patent retaliation provision, that force the end of the license in case of a licensee initiating a patent litigation.[117]

[116] Art. 1 Solderpad license available at: http://solderpad.org/licenses/
[117] Art. 3 Solderpad license available at: http://solderpad.org/licenses/

2.5 WEAK-COPYLEFT LICENSES

Weak-copyleft licenses are FLOSS licenses that mandate derivative works that arise from the weak-copylefted work to be licensed under the same license. There is still, however, a possibility to link weak-copylefted code with code subject to different licenses, including proprietary licenses. This approach forces the source code of the original work to be available, as well as modifications to it. Such licenses are not, however, considered as "ultra viral" since it is possible to bundle them or link them to other non-free software creating a larger work, containing weak-copyleft material together with proprietary licensed material[118]. Such larger works are accepted if the latter is a separate work and not a modification of the former. Examples of such licenses are the LGPL, the MPL, and the CERN OHL the last and one of the most prominent hardware specific licenses, developed by the well known European Organization for Nuclear Research.

The Free Software Foundation (FSF) decided to create a weak-copyleft license in order to settle the debate on GPL static and dynamic linking, as well as the debate on bundling and communication between free and non-free software.[119] The latter issue centers on the often not clear distinction between a mere aggregation of separate software and a single software built by aggregating more programs. The distinction, according to FSF, has to be established on a case-by-case base, distinguishing based on the mechanisms of communication and the semantics of the communication.[120] Such a distinction is also relevant in the hardware context, since well established

[118] Free Software Foundation, Why you shouldn't use the Lesser GPL for your next library Available at: http://www.gnu.org/licenses/why-not-lgpl.html

[119] Ibid. 25

[120] In the opinion of FSF the mechanisms can be exec, pipes, rpc; while semantics means what kinds of information are interchanged. If the modules are included in the same executable file, or modules are designed to run linked together in a shared address space they are most probably combined in one program, by contrast pipes, sockets and command-line arguments are communication mechanisms normally used between two separate programs. This means that if they are used in order to exchange informations between two or more modules this will probably be separate programs. But if the semantics of the communication are intimate enough, foe example the exchanging of complex internal data structures, that might be a basis to consider the two modules as combined into a larger program. From What is the difference between an "aggregate" and other kinds of "modified versions"?, Available at: http://www.gnu.org/licenses/gpl-faq.html#LinkingOverControlledInterface

See also: BAIN, M.. Software Interactions and the GPL. International Free and Open Source Software Law Review, North America, 2, feb. 2011. Available at: <http://www.ifosslr.org/ifosslr/article/view/44>.

boundaries on the use of strong and weak-copyleft are very useful to provide OSH with an effective and clear legal framework.

The CERN OHL is a license published by the legal department of the European Center for Nuclear Research, in collaboration with many OSH scholars.[121] The idea to implement OSH in CERN laboratories was brought up by Javier Serrano, a hardware engineer who was jealous of how his software coworkers were sharing, collaborating and contributing back to a diverse array of communities using FLOSS.[122] He asked the CERN legal department, in particular Ms. Myriam Ayass, to implement a license for sharing his hardware designs. Since then, three versions of the CERN OHL have been issued. The herewith discussion is, however, limited to version 1.2. One recurring element in the license agreement relates to concerns of proper acknowledgement the of previous licensor through copyright and trademark notices as well as maintaining of the disclaimer of warranties. An important difference between CERN versions 1.1 and 1.2 is that in the latter version the drafter chose to drop the *upstream notification* clause. This clause imposed the burden on licensees who modified a design the obligation to notify the changes to upstream licensor. Initially, this was an attempt to track the dissemination and evolution of designs in the author opinion an interesting idea but quite burdensome for downstream recipients especially if the modification were submitted by a substantial number of contributors. In the current version of the license, also clause 6.5 of V1.1 was removed. The removed clause was imposing as exclusive jurisdiction the place where an Intergovernmental Organization was settled. The clause was also requiring the dispute to be solved by arbitration.

The license starts with a preamble after a short introduction. It explains the purpose of the license as: *a tool to foster collaboration and sharing among hardware designers.*[123]

[121] All discussions are publicly available in the mailing list archive avialable at: http://lists.ohwr.org/sympa/arc/cernohl
[122] Cern infrastructure, including the biggest particle accelerator in the world, runs on Linux (RedHat)
[123] Cern OHL Preamle, available at: http://www.ohwr.org/attachments/2388/cern_ohl_v_1_2.txt

The text assigns the copyright of the text of the license to CERN and grants the freedom to use the license under the condition of not modifying it. The first section is devoted to definitions documentation is defined as:

> *"Documentation" means schematic diagrams, designs, circuit or circuit board layouts, mechanical drawings, flow charts and descriptive text, and other explanatory material that is explicitly stated as being made available under the conditions of this License. The Documentation may be in any medium, including but not limited to computer files and representations on paper, film, or any other Media.*

While product is defined as:

> *"Product" means either an entire, or any part of a, device built using the Documentation or the modified Documentation.* [124]

A subsequent section is devoted to the applicability of the license:

> *This License governs the use, copying, modification, communication to the public and distribution of the Documentation, and the manufacture and distribution of Products.*

This license contains two different sections. The first deals with the distribution of Documentation. Then, distribution of manufactured products is included. In the first section a copyleft clause is also included.[125] This clause imposes an express duty to relicense the documentation with the same or, where applicable, a later version of the License. In this same section a non-exclusive patent license

[124] 2.1 CERN OHL
[125] 3.4 e

is also granted. In the "manufacture and distribution of the product" section, a clause linking the manufactured hardware to the documentation is set:

> *The Licensee may manufacture or distribute Products always provided that, where such manufacture or distribution requires a license under this Licensee provides to each recipient of such Products an easy means of accessing a copy of the Documentation or modified Documentation .*[126]

The linking role of such a disposition is quite central to the license and at the same time quite innovative. It requires the licensee to link the physical artifact with the original or modified CERN OHL 's licensed documentation. To enforce such a provision, many possibilities are possible and OSH makers can unleash their creativity trying to figure out how to incorporate an analog link on their manufactured creations. This option might be fit for the purpose of enforcing the liking provision because a reference to license and documentation can be embedded in the source file and if the product gets manufactured by automated means the link can be physically printed when the source file gets manufactured. In other cases the possibility to place a copper plate with a URL on it was proposed and even stickers are considered as a viable option. The underlying idea of having a physical link on the physical product is quite important in order to effectively build, study, improve and share OSH designs by a community of makers and users.

The CERN OHL is defined by his drafters as "weak-copyleft". In fact the license is designed to ensure that modifications to the original design are shared with the community.[127] However, it does not attempt to stipulate that the designs of larger products that are integrated or linked with the OHL products also need to have their designs licensed under CERN OHL.[128]

[126] Art. 4.1 CERN OHL
[127] Clause 3.4 e
[128] Open Hardware at CERN, Mark Johnson, September 23, 2013 Available at:

An example of how a OSH device licensed under the CERN OHL can be used in larger and partly proprietary project can be theorized using the White Rabbit Ethernet switch.[129] This switch provides sub-nanosecond accuracy and picoseconds precision for the synchronization for large distributed systems, a network switch is a device that is used to connect devices together on a computer network by performing a form of packet switching. White rabbit is used in systems where large amount of data have to be synchronized almost in real time. Many of the large computer network systems that handle large amount of data, where time is critical (Eg. a stock exchange, or a particle accelerator) and which make use of White Rabbit, have some non OSH component[130] maybe even standing in the exact same network rack cabinet into which the White Rabbit switch is hosted. On the contrary it would be highly unpractical to have a switch that can be used exclusively with OSH devices. If considering the stock exchange is clear how many of the component of the stock exchange computer network will not be OSH, but white rabbit is a practical solution that might enable investors to take faster decisions.

osswatch.jiscinvolve.org/wp/2013/09/23/open-hardware-at-cern/

[129] Withe Rabbit applications, documentation and more it is available at: http://www.ohwr.org/projects/white-rabbit see also: http://www.whiterabbitsolution.com/

[130] A list of Wite Rabbit applications and projects it is available at: http://www.ohwr.org/projects/white-rabbit/wiki/WRUsers

2.6 STRONG COPYLEFT LICENSES

In the hardware context some licenses tried incorporating a strong copyleft clause. In the previous discussions it was already indicated how CERN incorporated such a clause(Supra 2.5). In the following section a relatively detailed discussion will be made as to how the OCP and Rep Rap licenses enforce copyleft in hardware. Historically, TAPR is also another hardware specific copyleft contractual instrument. Such an instrument will be examined in the end of the chapter given the fact that it was drafted to be enforceable as a license as well as a contract.

Closely related the long standing debate in hardware and software licensing context will be discussed, this debate relate to the question if copyleft obligations can be imposed by contractual means other than by licenses. The debate is centered on the question weather a license is a contract or not, while common law doctrine and scholars, specially in the USA, are convinced that licenses are not contracts, mainly because they are unilateral permissions:[131]

"A license is a unilateral permission to use someone else's property. The traditional example given in the first year law school Property course is an invitation to come to dinner at my house. If, when you cross my threshold, I sue you for trespass, you plead my 'license,' that is, my unilateral permission to enter on and use my property." E.Moglen FSF attorney

Even if E.Moglev and part of the American doctrine are in favor of a net distinction, the boarder

[131] See: Pamela Jones, The GPL Is a License, not a Contract, available at: http://lwn.net/Articles/61292/
Jeff Neuburger, Ninth Circuit Rules on License Conditions versus Contract Covenants in Dispute over World of Warcraft Bots – MDY v. Blizzard, Part I , Available at: newmedialaw.proskauer.com/2011/01/03/ninth-circuit-rules-on-license-conditions-versus-contract-covenants-in-dispute-over-world-of-warcraft-bots-mdy-v-blizzard-part-i/

See also: Jacobsen v. Katzer on artistic license, http://newmedialaw.proskauer.com/2008/08/13/federal-circuit-says-open-source-license-conditions-are-enforceable-as-copyright-condition/

See also: on Jacobsen v. Katzer http://softwarelawyer.blogspot.de/2008/01/jacobsen-v-katzer.html

But also, from a civil law perspective: Guadamuz, Andrés, Viral Contracts or Unenforceable Documents? Contractual Validity of Copyleft Licenses. E.I.P.R. Vol. 26, Issue 8, pp.331-339, 2004. Available at SSRN: http://ssrn.com/abstract=569101

between licenses and contracts are very fuzzy, specially in civil law system where contracts can be unilateral and consent can be expressed by a mere action or by tacit acquiescence.[132] Since the difference depends on jurisdictions and would deserve an extended discussion, here the question wheder a license is a contract or not will not be discussed, and the main stream of FLOSS licensing articles and essay assume FLOSS licenses are not a contract therefore here the same line of thinking will be followed,[133] nothwistending what floss doctrine and scholars argue the most recent american caselaw point toward FLOSS copyleft licenses to be mere contracts since the Jacobsen v. Katzer case was reverted creating even more uncertenties with Versata Software, Inc. v. Ameriprise Financial Services, Inc.[134]

2.6.1 General Public License (GPL)

In this section one of the most popular and renowned FOSS licenses, the GNU GPL, will be discussed. As it was already stated, such a license has been in the market for almost three decades. In these three decades it has evolved, being redrafted three times. The version to be considered in this section is the third version of the GPL, also known as GPLv3, which was released in 2007. The relevance of this license not only hinges on the nature of general and public, on the fame of the authors[135], and the legal certainty that such a licensing model represents. Rather it is also related to the fact that many of the disposition thereby contained was challenged in courts[136] and because such a license is currently used by some relevant hardware projects.[137] This license is opened by a

[132] Bénédicte Fauvarque-Cosson,Denis Mazeaud, European Contract Law: Materials for a Common Frame of Reference: terminology, guiding principles, model rules, page 20

[133] Ibid. 48

[134] Aaron Williamson, Versata Court: Breach-of-Copyleft Claim not Preempted by Copyright Act, Available at: https://torekeland.com/blog/versata-copyleft-case

[135] Richard Stallman, Eben Moglen, Richard Fontana

[136] Joris Peeters, General Public License in Court - Analyses of the case law in EU countries Available at: https://www.law.kuleuven.be/jura/art/44n4/peeters.html

[137] RepRap, license all documentetion under GPLv2 or later all what is done in order to fit a software license to an hardware device is to include a paragraph outside the license text that states: *"For this purpose the words "software" and "library" in the GNU General Public Licence are taken to mean any and all computer programs computer files designs images videos sound recordings data results documents and all other copyright information*

preamble, which define the subject matter of the license:

> *The GNU General Public License is a free, copyleft license for software and other kinds of works.*

Here it can be appreciated how the drafters were already aware of the multiple possible objects of the license. This is evident when undertaking the choice to mention software and other kinds of works, allowing an indefinite array of works to be licensed under the GPL. The rest of the preamble sets the three main purposes that this licenses aims to achieve. First and most importantly, the GPL aims to keep software free in the sense of safeguarding the four freedoms, to use, study, share and improve. In the preamble the FSF interpretation of freedom is concisely expressed:

> *"To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others."*

The second purpose of this license is to make sure that licensees are aware of the absence of any warranty on the functioning of the software. Such a statement is quite common in FLOSS licenses since they are designed to apply to software that might be buggy or under development, in fact very often FLOSS is available since the very early stage of development, on a git repository or similar platforms, this because these are the mechanisms that enable distributed and decentralized software development. The last purpose, which can be read in parallel with the first, is that the licensed software must be free of restrictive patents. In fact if a patent applies to the licensed software, such an intellectual property restriction must be licensed in parallel with the code.

After meticulously defining the object of the license in the section number zero of the license, the

---

available from the RepRap project."

first section describes and define what is intended as source code.

The subsequent section explains which rights are granted under the license, and also makes the licensee aware that such rights are granted only if some conditions are accepted. These are mainly in section five and six. In section five the document describes requirements for compliance when distributing modified versions of the original source code. Here it is worth noting how the duty to license the whole work under the same license is put forward. This clause makes GPL a strong-copyleft license since it attaches to everything it is bundled with.[138] In section number six is explained under which condition is possible to convey compiled software, this is possible only if copies of the corresponding source code is made available, together with the binaries or upon request of users.

It is worth noting how the drafters of this license wanted to highlight the opportunity to profit from the GPL licensed work. This is done in section four where the case of the work being distributed in verbatim copies is considered. The freedom to distribute the work without warranty is separated from the freedom to charge for additional services of providing the software or even providing it with warranty. A defensive patent retaliation clause is also included in the eighth section, the considered section is titled termination. Section eleven is completely devoted to patents. Two paragraphs deal with different aspects of possible interaction between GPL licensed works and patents. The first paragraph is simply a grant of patents to every distributor, licensor, and modifier.

In the second is considered the scenario where someone is distributing a GPL software relying on a patent license.[139] This patent licensee knows that the program infringes some patents but he has a patent license, so he's not going to be sued. But third parties might be sued if the licensee redistributes such patented work. These clause tries to deal with such a scenario, requiring the patent licensee that is knowingly relying on a patent license to do something to ensure that he's shielding others when they carry out the freedoms that the GPL gives you.[140] The clause mention

[138] Clause 5 section C
[139] Section 11 GPL V3
[140] 4.2 Richard Stallman, speaking in Brussels, February 25th 2006, Available at:http://www.ifso.ie/documents/rms-

knowledge (knowingly relying) because it consider the case of blanket cross licensing agreements between corporations, these agreements may include hundreds of patents. In such a case a corporation holding a patent license but not being able to be a patent licensor might not even be aware of holding that patent license and therefore they are not bounded by the examined clause.

2.5.2 Rep Rap License

RepRap project was founded in 2005 by the mechanical engineer and professor Adrian Bowyer, it was the first low-cost 3D printer to be completely licensed and documented as OSH. As stated by the RepRap community RepRap printers currently are the most widely used 3D printer among the members of the maker community.[141] The founder of the project chose to release documentation under a custom license, named the Rep Rap License, it incorporates as it is the GPL text, extending the object of the license by broadening the definitions of the terms *software* and *library* in order to mean:

> *"any and all computer programs computer files designs images videos sound recordings data results documents and all other copyright information available from the RepRap project"*[142]

The undertaken approach, consist in using a software specific license and by changing the subject matter of the license applying it to hardware documentation project, this is not incompatible with the text of GPL, since in GPL's preamble it is expressly contemplated that the GPL might be fit for licensing software as well as other works (supra 2.6.1). The choice to use a carefully drafted, well tested, and authoritative license such as a modified version of the GPL in order to make it fit for

gplv3-2006-02-25.html#patents

[141] Jarkko Moilanen e Tere Vadén, Manufacturing in motion: first survey on 3D printing community, Statistical Studies of Peer Production. Availabe at: surveys.peerproduction.net/2012/05/manufacturing-in-motion/

[142] RepRapGPLLicence, available at: reprap.org/wiki/RepRapGPLLicence

hardware licensing is an interesting strategy that can provide a solid foundation which ensures effectivenes and legitimacy of the new born OSH license. The Rep Rap license was drafted in 2005, in the text it refers to the GPLv2, but it is a flexible clause that expressly allows licensor to use to any later GPL version. It is interesting to note that the drafters decided to open source the RepRap trademark, making it a non trademarked but copylefted name that has become a *common good* to use for many different related projects, in fact presently there are dozens of different RepRap designs.[143]

The copyright covers the teardrop device, the neologism "RepRap" in upper, lower, or any mix of cases in any typeface, and the phrase "Fused Filament Fabrication" and its abbreviation to "FFF". The teardrop, "RepRap", "Fused Filament Fabrication" and "FFF" are also covered by the GPL. Anyone can use them freely in any way they choose, but they may not be registered as trademarks or restricted in any other way. [144]

RepRap project faced many of the problems that a sloppy and unfit legal framework might pose, specially patent-wise.[145] This issues are hampering open and collaborative development, community based solutions and creating legal uncertainty. Rep Rap Vs Makerbot, is a good example of how lack of legal certainty can undermine distributed grassroots innovation and OSH development.[146]

---

143 A list of ready to make models is available at: http://reprap.org/wiki/RepRap_Options#Models

144 Ibid. 58

145 This problems comes form the so called Replicator controversy and patents granted without satisfaction of novelty requirement, a good summary of clashes occurring between RepRap and Makerbot community are outlined here. The first issue is available at: http://reprap.org/wiki/Replicator_controversy an updated overview of the issue is available at: http://richrap.blogspot.de/2014/05/makerbot-patents-twist-knife-on-open.html

146 See: Quick-release extruder patent application 2013, and MK5 extruder 2011 but also Three-dimensional printer with force detection patent US 20140117575 A1 and Steve Graber video posted on you tube three weeks before makerboot patent application, (https://www.youtube.com/watch?v=e119OXzkk7s)

2.6.3 Open Compute Project License

Open Compute Project was initiated by Facebook with the intention of disclosing datacenter designs; from motherboards, to rack servers and networking, up to the full datacenter design. This project aims to attract a wider community of datacenter developers which would contribute their designs, enabling more efficient datacenter designs to be created. Apparently such a model is demonstrating successful since the result is a datacenter full of vanity free servers[147] which is 38% more efficient and 24% less expensive to build and run than other state-of-the-art data centers, currently open computer hardware is running in datacenters of Facebook, Microsoft, Goldman Sachs, Fidelity Investments, Rackspace, Bloomberg, Riot Games and Orange.[148]

This project published two different licenses, one is defined as a more permissive license, modeled on Apache license. The other one is defined as more prescriptive and is modeled on the AGPL, which is the GPL modified to fit software running over a network (SaaS). An interesting feature of those licenses is found when defining the object of such licenses. In fact the subject matter of Open Compute License is *complete production files* this term is used instead of documentation, such a term is wider than CERN's definition of *documentation* since it also includes firmware, which is a software element really close and interdependent with electronic hardware, firmware is explicitly excluded in the CERN OHL.[149]

The so called more Open Compute prescriptive license is a rather strong copyleft license. It defines as "improvement" any derivative work, a specification or any modification extension or improvement of the functionality described in a specification, this is quite a wide definition of

---

147 Sean Gallagher, How Facebook threatens HP, Cisco, and more with its "vanity free" servers, arstechnica.com/information-technology/2013/07/how-facebook-is-killing-the-hardware-business-as-we-know-it/

148 Rich Miller, Open Compute Gets Down to Business, With New License and Certifications, February 3, 2014 Available at: www.datacenterknowledge.com/archives/2014/02/03/open-compute-gets-business-new-license-certifications/

149 Section 2.3 of CERN OHL, while in Open Compute Project clause 1, firmwares are included in clause 1, "complete production files"

See also: Servers can be 'vanity free' – Components that are taken for granted in conventional environments, such as chassis covers, can be eliminated to free up capital for more compute or storage capacity. Available at: http://www.penguincomputing.com/products/custom-rack-solutions

improvement. The copyleft clause is placed in section number seven entitled *Conditions for Distribution of Compliant and Licensed Products*:

> *You may exercise the licenses granted to you in this license subject to the condition that all licensed products and compliant products sold, offered for sale, or otherwise distributed by you or on your behalf are accompanied by full and complete copies of all complete production files, and (2) a copy of this License.*

This would be a strong copyleft clause except that it comes with an exception stating:

if you create an extension or addition of functionality of a Specification , if there is a commercially feasible means of manufacturing it as a separate physical component, the condition in this Section 7 does not require you to treat it as an Improvement[150]

This copyleft exception is a safeguard for companies participating in the project that want to develop some functionality or specification without giving them back to the community and retain the exclusive right of commercial exploitation.

It is quite clear that this license is corporately driven, and designed for large hardware projects. This results in the apparent focus on patent law which is considered in two ad hoc clauses, one for the grant of patent licenses in the initial specifications and one for licensing patents on improvement of specifications.

2.6.4 Three Dimensional Printing License (TDPL)

Chronologically speaking, the TDPL is the latest OSH license. It was drafted in 2013 by a well

[150] Art.7 OCP

known FLOSS scholar and it was attached in the appendix of a academic journal article that further explain the ratio of the license.[151] The license, however, is only a theoretical exercise since it has not yet being applied to any OSH project. The license is drafted expressly considering physical objects produced throughout automated manufacturing process. The terms of the license define two different objects to be protected by copyright: build form and source form of the work. The build is intended to be:

> *any form of the Work which is provided to any device for the manufacture of a Printed Article in an automated manufacturing process.* [152]

The source form is intended to be:

> *means the preferred form for making modifications to the Work. Source Form does not include file formats in which only minor changes (such as repair, resizing or changes of orientation may bemade), such as STL file formats.* [153]

The difference between the two forms is that the former is not illimitably modifiable, but it can be executed by a computer aided manufacturing machine transforming the file into a physical object. The latter, defined as source form, is a file in formatted in the in the preferred form for making modifications. For example if we consider academic text writing and subsequent automated manufacturing process procedure,(also known as ink or laser printing) we can imagine the source form to be a .tex or .txt file[154], and as build form a .pdf file ready to be printed but hard to modify.
A similar analogy can be theorized considering 3D printing, in this environment a quite common extension to distribute source files is .scad, this is a format that can be opened by a standard FLOSS

---

151 Eli Greenbaum, Three-Dimensional Printing and Open Source Hardware, New York University Journal Of Intellectual Property And Entertainment Law, Available: jipel.law.nyu.edu/2013/04/three-dimensional-printing-and-open-source-hardware/

152 Art.1.1 TDPL

153 Art. 1.9 TDPL

154 Latex (.tex), Open Document (.odt) or plain text are (.txt) are open standard, that can be freely modified by a diverse set of text editors.

CAD software, when compiled the user is able to see a three dimensional render of the source code, a rather big advantage of this format is that whatever text editor can open the source file and allow users to perform modifications of such files because they are mere ASCII code, making the .scad file modifiable by any text editor. [155]

The preferred format for distributing an object in build form for three dimensional printing is usually .STL (StereoLithography) since these files are an open file standard and are widely used in computer aided manufacturing and rapid prototyping. The .STL file format is good for printing out parts because it describes a part's surface geometry. On the contrary, .STL files are not very good for doing Computer Aided Design (CAD). They can be imported into CAD applications but are sometimes difficult to change, depending on the software used.[156] This entail that in order to comply with the four freedoms, and OSHAWA definition of OSH, both source form and build form have to be available enabling others to use, study, share and improve. This license defines a printed article as a physical article created by using digital files of the work in an automated manufacturing process. Such an open definition do not limit the scope of this license only to three dimensional printing files, indeed TDPL is also applicable to files executable by computer numerical control (CNC), milling, laser cutter machine[157], and files executable by many other automated manufacturing processes machines. Another important definition is the one of the *required notice* which is intended to provide information regarding authorship and copyright, interesting is the last part of this definition which include in the required notice:

> *A required notice also includes any element of a Work which causes the imprinting of the foregoing information on a Printed Article.*[158]

[155] If looking on thingiverse most if not all customizable things are distributed as .scad files, .stp is also a a standard CAD format that can be read by many CAD software since is an OSI standard.

[156] A more detailed explanation of file formats available for 3D printing is on-line in RepRap wiky, available at: http://reprap.org/wiki/File_Formats

[157] Lasersaur, The Lasersaur is an open source laser cutter. Designed it to fill the need of makers, artist and scientist who wanted a safe and highly-capable machine. Unlike others it comes fully loaded with knowledge to run, maintain, and modify. Available at: http://www.lasersaur.com/

[158] Art 1.8 TDPL

After the first section devoted to definitions the license explains what the licensee is allowed to do, he is allowed to:

reproduce, modify, prepare Derivative Works of, publicly display, publicly perform and distribute the Work and such Derivative Works in Source Form or Build Form, and generate Printed Articles using the Work.[159]

A licensor can benefit from the aforementioned rights if three conditions are respected. The first condition is a prohibition of relicensing the work, the following is the prohibition of modifying or removing any required notice. The next provision is a copyleft clause[160] that force derivative works to be relicensed under the same terms as well as to insert a modified notice. It is than considered the case where a required notice is included in the source file, and during the automated manufacturing process it causes the imprinting of information on the manufactured object. This is a very peculiar and interesting clause, that allow third parties to retrieve licensing terms and the documentation from a link printed in physical object. In fact this provision require a modified notice to be imprinted in the same location and format of the previous notice.

The next section[161] is titled "N*o removal of notices from printed articles "* in this section the so called *analog gap* is expressly addressed, in fact when the article gets printed copyright does not apply anymore to it and there is no prohibition for third parties to re-digitalize the article and eliminate the required notice. Allowing them to distribute it without reference to license or documentation making it *de facto* closed source. To prevent such a scenario this clause sets up a contractual obligation not to remove the required notice, the obligation is addressed to third party who's the property of the object gets transferred. In the author's opinion this two clauses currently

[159] Art. 2 TDPL
[160] Art. 3.3 TDPL
[161] Art. 4 TDPL, No Removal of Notices from Printed Articles

represent the most efficient copyleft-like hardware solution, on the contrary other scholars argue that imposing contractual obligations in a license is highly unpractical:

He suggests that we use contract instead, and that we apply contract to all parties downstream, and then sophomoricaly announces that this will have a transactional cost! Of course it will. That's why we don't do it. Having everyone who buys, resells, and distributes Open Hardware individually sign a contract is unworkable and everyone who has approached the problem has looked at this for a minute and discarded it.[162]

2.6.5 Creative Common Licenses:

Creative Commons is a no-profit organization founded by professor Lawrence Lessig in 2001, the organization has released several copyright-licenses known as Creative Commons licenses. These licenses allow creators to decide which rights they reserve, and which rights they waive for the benefit of recipients and downstream creators,[163] establishing a "modular" mechanism that allow authors to license their work under a customizable some right reserved license.[164]

This set of licenses has had a remarkable success and it had being applied to material released over widely accessed internet platforms such as Wikypedia which make use by default of dual licensing CC-BY-SA[165] and GNU Free Documentation License. [166]

Creative common licenses are relevant to OSH because many relevant projects[167] use them to license documentation, this license are widely adopted to copyright content hosted in web-pages

[162] Bruce Perens, comment to the three-dimensional printing license, available at: http://lists.ohwr.org/sympa/arc/cernohl/2013-04/msg00006.html

[163] Lawrence Lessig , "Free culture : how big media uses technology and the law to lock down culture and control creativity ",Penguin Group (USA) Inc., New York , 2004

[164] About Creative Common licenses, Available at: http://creativecommons.org/licenses/

[165] In this verison, CC-BY-SA, means Attribution clause and Share a like(copyleft) clause

[166] Wikypedia copyright policy is available at: http://en.wikipedia.org/wiki/Wikipedia:Copyrights

[167] Arduino,Open Source Ecology, SparkFun

because they are designed to apply to images, text, video and data, this are the kind of works that are mostly used to document a OSH design[168].

It is worth to note two peculiarity about the use of this licenses for OSH documentation:

the first is that this licenses are drafted for creative works and data, but not for technical documentation this makes them not an excellent solution for OSH documentation mainly because of lack of language clarity but also because creative common licenses lack of any reference to patent, in fact this licenses do not include any patent grant clause and neither patent retaliation clause.

The second reason for not using Creative Common licenses is because of their nature of "some right reserved licenses" in fact the author of a work can include a copyleft clause, but also a non commercial and a non derivatives clause this two restriction do not fit the open source neither the free software definition, therefore if a subject license their documentation with a non commercial or non derivative license the work will not be OSH.[169]

2.6.6 The contract issue and the Tucson Amateur Packet Radio (TAPR) OHL:

In general, licenses could be understood as authorizations to do an act which in principle is forbidden. In fact OSH and FLOSS licenses use the fact that users of those works might want to use, share or modify the work without the original author express consent, this action would in principle make them copyright infringer. FLOSS and OSH licenses allows the licensee to exercise the basic four freedoms on the licensed work[170] and in exchange some of the licenses might ask users to adopt behaviors which ensure reciprocity, requiring downstream recipients to share back improvement with the community, this is the case of copyleft clauses in the license. Some license explicitly claim not to be a contract, while others do not consider the issue expressly, a key

[168] See: http://www.instructables.com/

[169] Alicia Gibb, On Creative Commons and Open Source, Available at: www.oshwa.org/2014/05/21/cc-oshw/

[170] The freedom to use, study, share and improve the work.

difference between licenses and contracts is the unilateral nature of the first[171] on the contrary contracts are characterized by an offer and an acceptance but this is can vary across jurisdictions(supra 2.5).[172]

The TAPR OHL attempts to achieve the same ends for hardware as the GPL does for software, but follows a somewhat different track. The GPL is expressly a copyright license, and claims not to be a contract. Conversely, the TAPR OHL is built around both license and contract concepts and operates similarly to a click-wrap agreement .[173] In fact while this license is furnished with an express and very complete patent grant, it lacks of any copyright grant. The contract obligations are used to bound recipient and create a copyleft-like obligation to force downstream recipients to contribute back their improvement and require downstream recipients to redistribute under the TAPR agreement any copy of the physical work or of the documentation.[174]

When considering contracts to enforce copyleft-like obligations it was noted by a renown FLOSS scholar that a licensor can also use an old-fashioned contract saying "I am making this available to you on the condition that you do (or do not do) the following with it." Such a clause could be included, however, to the extent that the underlying material isn't subject to copyright. Thought some US and Israeli case law[175] point out how such a scheme might be feasible, and how parties can contractually create copyright-like obligations for non-copyrightable materials.[176]

Other scholars are critical to such an approach stating that contract law is not going to work very well. Since creating a web of contracts which seek to create a pseudo-intellectual-property-right may work as long as everyone with access to the hardware or the designs is bound by the contract. The predicament of such an arrangement is that it only takes one person to gain access and not be

---

171 Par. 9 (Acceptance Not Required for Having Copies) GPL V3

172 See:Eben Moglen, https://fsfe.org/campaigns/gplv3/barcelona-moglen-transcript.en.html#q7-a-contract

173 John R. Ackermann, Toward Open Source Hardware, University Of Dayton Law Review, Volume 34 Winter 2009 Number 2, Available at: www.tapr.org/Ackermann_Open_Source_Hardware_Article_2009.pdf

174 Art. 4-5 TAPR License.

175 ProCD vs. Zeidenberg and US Supreme Court case (Feist v. Rural Telephone), also District Court of Tel Aviv 1981 case of Kotlitzky v. Alkalai and 1991 case of 2307/90 Hadas Engineering Systems Energy v. Haboneh Engineering Systems, Ltd. Again 2005 case of Matan Y. Communication and Location Systems v. Miltel Communications Ltd.

176 From CERN OHL Mailing lists Available at: servicelists.ohwr.org/sympa/arc/cernohl

bound by the contract for the scheme to fail. If the scheme fail it means that some licensee is in breach of the contract, the licensor can act against the licensee but have no remedy against the good faith recipient under contract, because contractual obligation are *in personam* relationship between the parties.

Another critic directed towards the use of contract instead of licenses is that copyright has an international span and is more homogeneous than contract law which can substantially differ from country to country.[177] At last it is important to notice a concern that some scholars as well as activists highlighted. These stakeholders are worried that even if open source hardware legal community manage through the use of a non-patent patchwork of IP rights to impose open obligations on any specific piece of hardware or if such end is achieved using contract law, such a mechanism might not be a win for OSH community because it might empower the proprietary side to apply even more protection to their designs than they do nowadays, defeating the purpose of spreading knowledge and open technologies.

[177] Richard M. Stallman, Don't Let 'Intellectual Property' Twist Your Ethos, Available at: https://www.gnu.org/philosophy/no-ip-ethos.html See also: Giuditta Cordero Moss, Lectures On Comparative Law Of Contracts, Available at: folk.uio.no/giudittm/PCL_Vol15_3[1].pdf

Chapter 3 Other legal instrument useful to manage intellectual property in OSH projects

3.1 TRADEMARK

Trademarks are signs capable of being represented graphically in particular: words, including personal names, designs, letters and numbers but also the shape of goods or of their packaging, provided that such signs are capable of distinguishing the goods or services of one undertaking from those of other undertakings. [178] When it comes to FLOSS and OSH projects, trademarks are the name of the project and the logo[179]. The overall purpose is maintaining control over a project by having the monopoly over the use of the name and distinctive signs. For example, very successful projects such as Arduino PCB's and SparkFun Industries maintain leadership position in the market, by releasing all specifications, blueprints and schematics but maintaining the control over their trademark and innovating at a very hight rate. In doing so they do not lose control over the technology they share[180] rather successfully manage also to maintain an active community of developers, contributors, customers and product evangelists that produce documentation and projects based on the aforementioned PCB and electronic devices themselves pushing newbies to use the already used or most well documented technology.[181]

The market capitalization indices surveys are indicating that the pecuniary values of intangible

[178] Art. 4 Council Regulation No 40/94

[179] See, Gnome trademark guidelines, http://www.gnome.org/foundation/legal-and-trademarks/,see also ubuntu intellectual property policy, http://www.ubuntu.com/legal/terms-and-policies/intellectual-property-policy

[180] In fact many cheaper Arduino clones are availoable on the market but most user still buy the original board even if it is a bit more expensive.

[181] Raspberry is a well known single board computer, but on the market at the same price there are much powerfulalternatives such as Odroid-U3 from the Corean company Hardkernel. Onion-PI is availble at:http://arstechnica.com/information-technology/2013/06/onion-pi-turns-raspberry-pi-into-tor-proxy-and-wireless-access-point/ while Odroid-U3 is available at:http://www.hardkernel.com/main/products/prdt_info.phpg_code=G138745696275&tab_idx=1

assets of which trademarks are one of the major intangible assets, account for a large percentage of company tech companies assets. According to such survey results it is estimated that intangible assets account for about 80 percent of the average firm's value.[182] In fact the portfolio of intangible assets of a highly innovative and technical research and development firms is inextricably intertwined with its patents. As a result it could be deduced that the company's brands signify, for companies that offer services or goods and are not mainly focused on RD, the most valuable assets.

Once a brand has developed market reputability, in terms of product positioning, trademarks play an indispensable role. In FLOSS and OSH business model tracking the origin of a product even becomes more important. Part of the reason is that the open source business model is loose in terms of copyright and patent protection. Thus, there is a likelihood that companies and other market players might, based on the initial developers shared technical specifications and/or source files and code, manufacture a clone product, but if the original project is trademarked the newcomers will have to redistribute it under a different name. If the original developer managed to create an active community around the product is likely that the initial developer have a considerable advantage over competitors because his product is more documented and supported. The extended testing and bug reporting of various developers in the chain, and most importantly the support of the community, give the original developer a considerable advantage upon new comers. A good example are products developed for newbies students and hobbyists of information technology and electronics such as the Debian-based distribution Ubuntu, and the OSH microcontroller prototyping board Arduino. Both of these products are clones or even better versions based on the original good.[183] The products are, however, distributed under a different name. Employing different naming makes the products less appealing to potential customers. Less reputability of the products basically

[182] See Ocean Tomo (2010). The S&P 500 is a free-floating, capitalization-weighted index, publishedsince 1957, of the prices of 500 large-capcommon stocks actively traded in the US. Thestocks included in the S&P 500 are those of largepublicly-held companies that trade on either of thetwo largest American stock market exchanges:the New York Stock Exchange and the NASDAQ.

[183] Linux Mint, Ultimate Edition and Bodhi Linux are Ubuntu based distros, while Sparkfun's RedBoard or XinoRF are examples of Arduino based developers board.

emanates from the fact that the products lack community support and documentation, another factor of paramount importance the reliability that the extended presence on the market could provide, pushing new costumers towards products that are more documented and supported therefore easier to use.

A logical question a reader could ponder about trademark is an enforcement question. How a project can enforce a trademark? The answer is very simple. One of the most legally significant step that a project can take is to register the trade mark. Fortunately, registration is affordable to individual hobbyist and star-ups.[184] Besides registration it is also usually recommended to specify in writing some trademark guidelines. The purpose of the guideline is to define and clarify trademark ownership and how a third party may use it.

The following three guidelines from the Ubuntu trademark suffice to explain the trademark guidelines ratio. The Ubuntu guidelines allow the use of the Ubuntu logo and font to be printed on shirts or in general to be used by the community members based on three conditions. First, there is no commercial intent behind the use. Second, the use of the trademark it has to be strictly related to the product, if the way the trademark is used might mislead potential costumers this means that the trademark is not used according to the guidelines. Third, there is no suggestion (through words or appearance) that the project is approved, sponsored, or affiliated with Ubuntu or its related projects unless it actually has been approved by and is accountable to the Ubuntu Community Council.[185]

The use of trademark and logos by community members is critical in a OSH/FLOSS project, because on the one side founding members of the project wants the project to get larger and larger

[184] In Italy registration of a mark is usually below two hundred euros, while a community wide trademark can be around nine hundred euros. Sources: http://www.uibm.gov.it/index.php/marchi/registrare-un-marchio-in-italia/quando-registrare-un-marchio-2 and https://oami.europa.eu/ohimportal/en/fees-and-payments

[185] Ubuntu trademark guidelines are available at: http://design.ubuntu.com/brand/ubuntu-logo

but on the other side it have to be controllable, so that the released material is compliant with a set of reliability, quality and stability parameters, here there is a tension between openness and effectiveness of the released products. Therefore it is highly recommendable to establish clear guidelines to enable contributors to make a proper use of the brand and avoid misunderstanding that may result in legal proceedings as it is as demonstrated in the Freecycle Vs FreecycleSunnyvale case.[186]

The Arduino project chose a different approach and issued two different trademarked logos. The first identifies projects, initiatives and activities coming directly from Arduino. The second one is a community logo (different from the official one but rather similar), that is intended to be used by different Arduino-based communities around the globe that self-organize local activities and brand on-line content focused on Arduino.

The community logo is customizable allowing users to change the colors and is furnished with a space where to insert the specific name of the Arduino-based project. It is intended to be used on web pages, social networking pages and merchandising material. It is expressly prohibited the use for commercial products and electronic boards even if not commercial this in order to not create confusion between officially released boards and community developed ones.[187] Likewise all major FLOSS and OSH projects make extensive use of trademark policies although they differ substantially on the way they forbid the use of trademark for commercial purposes and for a products that are of the same kind as trademarked one. For example, Python forbids the use of its trademark for any other programming language Arduino for any other electronic board.[188] Usually a FLOSS project requires the use of its trade mark for the redistribution of copies of its software only if the software is unmodified or if it passes a test suite.[189] This is in order to maintain a main branch

[186] CHESTEK, P.. Who owns the project name?. International Free and Open Source Software Law Review, North America, 5, nov. 2013. Available at: http://www.ifosslr.org/ifosslr/article/view/87.

[187] Arduino Community Logo, and guidelaines are available at: arduino.cc/en/Trademark/CommunityLogo

[188] PSF Trademark Usage Policy, available at: https://www.python.org/psf/trademarks/

[189] How To License The Powered By OpenStack Logo , Available at: http://www.openstack.org/brand/powered-by-openstack/

officially released by the core team.

Third parties that want to fork a FLOSS project are, however, free to do so but they are required to change the name and can not use the same logo as trademark. A different strategy that can bring even more openness into a project, mainly because it would allow anyone to use the project name ,is the choice not to register the trademark, or license it with a GPL-like license that allows anyone who comply with certain restrictions to use the name that distinguishes the project. Such a choice can be risky because it can generate uncertainty over the origin of a product and this can undermine the reliability of the tool or product. In fact if other manufacturers release less stable clones or upgrades of the original device using the "public domain trademark" the users might not be sure of which device and from which manufacturer they are acquiring. Such an option can only be effective if a strong community is *de-facto* ensuring and testing the quality of products released under the shared trademark or project name and if there is one commonly accepted and used platform where share the materials and release different versions of the product, such as the RepRap wesite.[190]

Trademarks are a very effective tool to control and manage the ownership and monopoly-like rights over products or projects name and distinctive marks, independently from their software or hardware nature. The brand and logos carry values and represent the community the distinctive marks that identify the project is a crucial asset that have to be accurately managed with solutions and precautionary measures such as a well drafted trademark policy and/or a community logo, in order to empower users to be part of the community but at the same time safeguard the origin and goodwill of official released project and products.

[190] Reprap indipendently developed models are listede at: http://reprap.org/wiki/RepRap_Options while the license and trademark agreement are outlined at: http://reprap.org/wiki/RepRapGPLLicence

3.2 PATENT SHARING MECHANISMS

Patents are a set of exclusive rights granted by a sovereign state as a result of a time and resource consuming bureaucratic procedure, also known as “Patent Application” this characteristics makes patents an instrument that can be used and afforded mostly by corporate OSH developers or by traditional hardware manufacturers if they decide to make their product free and open source. To effectively enforce any restriction on a physical technical invention a patent is required, because patents are designed to protect technical invention establishing a trade off between the inventor that disclose his invention contributing to the advance of the state of the art, and the sovereign state that give the monopoly to the inventor over the commercial exploitation of his invention. Once a patent or a monopoly over the making, using, selling, offering for sale, or importing the invention is granted the holder can use different licensing methods to “share” it, licenses are permissions issued by the licensor that allow others to make, use, sell, or import the patented inventions.

3.2.1 Patent pooling

Patent pools are not new, in fact they existed since the eighteenth century, when two competing sewing machine manufacturers decided to share patents in a pool instead of fighting over innovation and waisting their resources in litigation.[191] Following this virtuous example many pools have been created ranging from airplane manufacturing to technology standards such as DVD, RFID[192] or MPEG 2.[193] The most relevant question is how patent pools can be relevant to OSH?

Some patents pools charge for licensing out technology, other do it for free, as in the case of Linux Patent Common / Open Invention Network. Linux Patent Common and its subsequent evolution the Open Invention Network (OIN) are attempts to tackle the problem of patents in the FLOSS environment. In fact OIN is a defensive patent pool originally formed by seven large companies[194]

[191] New Uses for Patent Pools, Bart Showalter and Trampas Kurth Available at: www.iptoday.com/articles/2008-9-showalter.asp

[192] RFID Journal, available at: http://www.rfidjournal.com/articles/view?2636

[193] Mpeg LA, MPEG-2 Introduction, available at: www.mpegla.com/main/programs/m2/pages/Intro.aspx

[194] Google IBM NEC Novell Philips Red Hat Sony Available at: www.openinventionnetwork.com/community-

that operate in the information technology business sector within Linux systems. The founding members decided to pool some of their patents together and create a shared portfolio. Currently, however, anyone willing to sign the license agreement can access this portfolio of Linux-related patents and gain a:

> *royalty-free, worldwide, nonexclusive, non-transferable license under OIN Patents to make, have made, use, import, and distribute any products or services*[195]

On the other side, Licensees commit to grant a license to the other players in the pool:

> *you, on behalf of yourself and your Affiliates, grant to each Licensee and its Subsidiaries that are Subsidiaries as of the Eligibility Date a royalty-free, worldwide, nonexclusive, non-transferable license under Your Patents for making, having made...*[196]

OIN's cross license mechanism allows patented components of the Linux system to be a common good between members of the community. It acts as one of the key methods through which open source leaders and innovators can deter patent aggression, fostering freedom of action in FLOSS.
It should be noted that the formalities to access the pool are very low. Both corporations and as individual can access the pool by simply printing and signing the license agreement. This ease of use resulted in OIN becoming a large community of Linux developers with almost nine hundred licensees representing the larger players in FLOSS ecosystem. Such a project might be a good source of inspiration for OSH developers that operate in a highly patented environment such as three dimensional printing. This mechanism foster innovation overcoming the limitation imposed by patents, contributing to share technical knowledge and it is therefore an appropriate tool that demonstrated itself very effective in the Linux environment and on the wake of this successful project more similar pools are seeing the light (Infra. Defensive Patent License).

---

of-licensees/

[195] Art. 1.1 OIN License agreement

[196] Art. 1.2 OIN License agreement

3.2.2 Unilateral patent non-aggression statement

Unilateral non-aggression statements are a new form of patent sharing, a righteous example is represented by Ecopatent Commons and Tesla motor's unilateral patent grant.[197] The first example is an initiative started in 2008 by IBM, Nokia, Pitney Bowes and Sony in partnership with the World Business Council for Sustainable Development (WBCSD). It was later joined by other transnational corporation, the mission of the common is:

> *the free sharing of knowledge which can provide a fertile ground for new collaboration and innovation. Sharing environmental patents can help others become more eco-efficient and operate in a more environmentally sustainable manner—enabling technology innovation to meet social innovation.*[198]

By pledging patents to the common, applicants share the inventions protected by the submitted patents and covenant, or pledge, not to assert any of the pooled patent against implementers of the shared technology.[199] The Eco-Patent Commons is an interesting solution to push companies to share their patents, and collect them in a centralized database. The idea of Eco-Patent Commons is not to share patents that are central to a company business model since it would be counterproductive for the company itself. The common is designed to be formed by those patents that leading businesses or universities may hold, that provide environmental benefit and do not represent an essential source of business advantage for these companyes. This can eventually lead to share "secondary" patents, intended as patents that do not give the company a considerable technical advantage over competitors, that have become obsolete or more in general that are not highly innovative.

The second example cited above is a single-firm initiative. This company decided to:

---

[197] Elon Musk, All Our Patent Are Belong To You, published on June 12, 2014 Available at: http://www.teslamotors.com/blog/all-our-patent-are-belong-you

[198] About the Eco-Patent Commons, Available at: http://ecopatentcommons.org/about-eco-patent-commons

[199] Eco-patents, Ground Rule, Available at: secopatentcommons.org/about/rules

not initiate patent lawsuits against anyone who, in good faith, wants to use Tesla's patented technology; [200]

Such a worthwhile initiative is expressly announced in the name of open source. As some sources claim this can be a marketing initiative or an attempt to spread and standardize Tesla's battery technology[201]. Nonetheless it is a groundbreaking initiative that has no or little precedent, in fact Tesla is currently a market leader company in electric car manufacturing. A copyright analogy can eventually be traced with Netscape disclosure of its source code, nevertheless it have to be considered the important difference, in the examined case Tesla is economically growing and wealthy company while Netscape in 1998 was loosing the browsers war.[202] It is to be hoped that others hardware big players will follow the same line and share their knowledge by rejecting patent aggression. This would be a important change of course in international patent management polices, since rejecting patent aggression is the best way to foster innovation, on the one side a company is forced to innovate at a higher ratio since it can not rely on any accorded monopoly on the other side availability of technological invention can foster decentered innovation, because more actors around the globe can use, study, share, improve and sell technologies that otherwise won't be accessible.

3.2.3 The Defensive Patent License (DPL)

The recently published Defensive Patent License (DPL) was drafted by J. Schulz and J. Dourban, with the help and collaboration with many relevant stake holders, representing FLOSS and OSH industry, scholars and activists. The stakeholder included representatives of Open Source Hardware

[200] Elon Musk, All Our Patent Are Belong To You, Available at: www.teslamotors.com/blog/all-our-patent-are-belong-you

[201] Tesla's Patent Giveaway: Please Use Our Batteries, Available at: blogs.wsj.com/corporate-intelligence/2014/06/12/tesla-patent-giveaway-please-use-our-batteries/

[202] R.E. Lord,Is Tesla Motors a Good Stock to Buy?, Available at:www.prognog.com/investing/clean-tech/is-tesla-motors-a-good-stock-to-buy.html and http://finance.yahoo.com/echarts?s=TSLA+Interactive#symbol=tsla;range=1y;compare=;indicator=volume;charttype=area;crosshair=on;ohlcvalues=0;logscale=off;source=undefined;

Association, Electronic Frontier Foundation, OIN, Wikimedia Foundation, and many others. DPL v1.0 is more than just a patent license. It is a new legal mechanism that aims to create a network of patents shared between DPL users that would enable any licensor to become a licensee of other users by creating a mutually beneficial mechanism where users can benefit from other's patents and this encourage innovation and reduce patent litigation. [203]To be more precise the license is a distributed standardized license, where the licensor offers licenses for his entire patent portfolio, in fact as is stated in DPL the licensor:

> *has committed to offer a license to each of its Patents under the DPL*[204]

beneficiary of this licenses are all entities that already accessed the network, in the DPL they are defined as user:

> *DPL User a worldwide, royalty-free, no-charge, non-exclusive, irrevocable (except as stated in Sections 3(e) and 3(f)) license, perpetual for the term of the relevant Licensed Patents, to make, have made, use, sell, offer for sale, import, and distribute Licensed Products and Services that would otherwise infringe any claim of Licensed Patents.* [205]

A reader can appreciate how the license is perpetual barring a few of exceptions, this are: in the case where a licensee sues a DPL user offensively, or in the case licensee stop offering patents under DPL and issue exclusives licenses, or choose the discontinuation route by announcing in due time he will stop offering DPL licenses. [206] In this last case the licensor who opt for the discontinuity route will not have access to patents in the pool unless he doesn't renegotiate them with the every single licensor. The users who already obtained a DPL license, however, can still use it even if the licensor preceding him is leaving the network. To other parties, who are not DPL users, it is possible

---

[203] Schultz, Jason and Urban, Jennifer M., Protecting Open Innovation: The Defensive Patent License as a New Approach to Patent Threats, Transaction Costs, and Tactical Disarmament (April 16, 2012). Harvard Journal of Law and Technology, Vol. 26, 2012. Available at SSRN: http://ssrn.com/abstract=2040945

[204] DPL Art. 1.7 par. A

[205] DPL Art. 2 License Grant

[206] DPL Art 1.4 - 1.5

to issue normal licenses. The normal license is not the exclusive license since the patented technology will also be under DPL. The underlying rationale of DPL membership mechanisms makes it easy to be accessed and hard to leave and in the opinion of the drafters, this can be an incentive to join the network, but it can also scare many potential users. The push factor for companies to join this DPL network primarily emanate from the fact that costs and benefit of patent applications gets distributer over a network, this can reduce application rejection risks, cross licensing costs and litigation costs. It is also pointed out that such a licensing scheme can favor the entrance of start ups or small businesses by providing them with pro-bono consultancy from the DPL foundation and partners and helping them to file patent applications and access the DPL network. Not least but lastly a favorable point of DPL mechanism is that it would help fight so called "patent trolls." It help prevent patent trolls since a patented technology is licensed to the network and even if a troll acquire a license from the original licensor. Thus, he will not be able to sue others since they also are licensee of that technology through DPL.

Besides the abovementioned benefit there are also some concerns that are being raised about the project. These are mainly the lack of incentives for companies to join the network, and more specifically critics point out that requiring companies to share their entire portfolio is excessive and might scare some important actors. It is important, however, to note that the requirement to share their entire portfolio is established in order to avoid that companies share just the lousy or "secondary" patents and keep the strategic ones for exclusive licensing,[207] to avoid this unpleasant situation and to avoid companies making subsidiaries to which assign the strategic patents. DPL requires users to share all the portfolio and set ad-hoc mechanisms in order to avoid the use of multiple entities, this is achieved through a clause that defines DPL licensor and licensee as:[208]

any individual, corporation, partnership or other entity ….including all Affiliates of

[207] As is the case for Eco patent common.
[208] Ibid. 17

such entity.[209]

Affiliate is broadly defined as any juridical person which the licensor or licensee possess more than 50% ownership, or which is controlled directly or indirectly by the licensor or licensee.[210] Other concerns relate to the possibility of free riding, antitrust issues, as well as the difference with the OIN system. With regard to the last concern it is relatively easy to spot how OIN is centralized and it's applicability is limited to the Linux system. While DPL aims to be as much decentralized as possible. Even if DPL is information technology oriented and inspired, it is not limited to it like OIN, this will hopefully allow the networking of all kind of patents that are in the portfolio of licensor and licensees. Free riding is a possibility but it has to be acknowledge that free rider most probably will be small companies or companies that can not contribute to the pool with a patent portfolio, this because if a company can fully exploit high-tech patents it is highly probable that it will soon or later contribute back to the community by pledging patents. The idea of DPL is to create a mechanism to effectively share knowledge, moral and community norms are suppose to do the rest, once the community of users will reach a critical threshold.

The official launch of this revolutionary instrument is expected in November 2014. A lot of the success of the DPL is highly dependent on the relevance of initial licensor's patent portfolio. Once a critical threshold is exceeded the network will automatically attract new members and will be fully operational.

Some critics[211] argue that patents are an obsolete instrument specially unfit for software, but to imagine a world without them is utopian, this because software patents are here to stay.[212] If from one side is good to try and limit the use of patents in software by lobbying governments and and raising concern about their legitimacy, from the other is good to face reality and develop

[209] Art. 1.14- 1.15

[210] DPL Artt.1.1, 1.14, 1.15

[211] FSF is fermly against software patents see: Richard Stallman, Software patents — Obstacles to software development, Available at: https://www.gnu.org/philosophy/software-patents.html

[212] J. Schultz, presentation at Stanford Center for Internet and Society, available at: https://www.youtube.com/watch?v=ttB_mjcIKcY

instruments to dam the consequences of an excessive use of patents in high-tech industry. DPL is an innovative instrument with ambitious prospects, in the author's hope DPL will be an instrument that will bring more openness in the international patent framework, fostering grassroots technological development and technology transfer.[213] But this will depend on how big players will react to DPL, if following Tesla example or by supporting the status quo.

3.3.4 Prior art and defensive publications

In order to successfully file a patent application three basic requirement have to be meet, namely, the invention has to be novel, it has to involve an inventive step and it has to be susceptible of industrial applicability. The novelty requirement imply that the invention is not part of the state of the art or a prior art. What is considered the state of art or prior art is knowledge that existed prior to the relevant filing or priority date of a patent application, whether it existed by way of written or oral disclosure.[214] In practice, however, patent offices often lack the commensurate time and resources and when filing high-tech patents a lack of examiners competency is also reported[215]. The patent offices usually have excessive workload, occasionally granting bogus or poor quality patents.[216]

To help patent examiners have quick and reliable access to disclosed prior art it becomes a practical necessity to issue a proper defensive publication and disclose the invention on a widely accessed prior art database. When an enabling publication or defensive publication is made, no one else can get a patent for the invention and the inventor share the benefits of the invention with the public.

[213] As defined by UNFCC, one of the majour obstacle in implementing this instrument is precisely Intellectual Property, and more specifically patents, http://unfccc.int/cooperation_and_support/technology/items/1126.php

[214] WIPO Intellectual Property Handbook:Policy, Law and Use, WIPO 2004Second Edition, Wipo Publication no. 489

[215] Doug Lichtman & Mark. A. Lemley, Rethinking Patent Law's Presumption Ofvalidity, StanfordLaw Review, Volume 60, Issue 1, Available at: http://www.researchgate.net/publication/228188392_Rethinking_Patent_Law%27s_Presumption_of_Validity/links/02e7e51eaa79346c23000000

[216] Daniel Nazer, Why is the Patent Office So Bad At Reviewing Software Patents?, HomeAboutOur Available at: March 17, 2014 https://www.eff.org/deeplinks/2014/03/why-patent-office-so-bad-reviewing-software-patents

Publication is, thus, the least expensive means of ensuring freedom of use of and promoting innovation, making this solution attractive for individual or hobbyist OSH developers.

Defensive publications are documents that provide descriptions of a product, device method or in general anything that can be a patentable invention so that it enters the public domain and becomes prior art. Another advantage of the aforementioned document is that it is less complicated and more flexible compared to a patent application, this means that the inventor itself or any layperson can draft one. A defensive publication is usually composed of a title, the background of the innovative concept, what the invention does and the advantages it has over previous ways of doing similar things, and finally a description of how the invention works.[217]

Some noble attempts to put together databases of defensive publications already proved effective, the most renown projects are: *research disclosure*, *defensive publications* and *linux defenders*.[218] These initiatives point in the same direction, namely, creating a well organized database furnished with well drafted enabling publications that can help patent examiners find prior art. This very same databases are also used by third parties interested in the invalidation of a patent application, by submitting prior art that can force a patent office to reject a patent application, this was the case of Electronic Frontear Foundation when undertaking the "Patent busting project"[219] as well as fight for Open 3D Printing campaign[220].

Defensive publications are a very effective instrument for FLOSS and OSH developers that do not want or cannot afford to go through the patent application process. In such cases, submission of prior art can enable inventors to have the paternity of the invention but not a monopoly over the use,

[217] Defensivepublications, FAQ, How to write enabling publications, Available at: www.defensivepublications.org/defensive-pubs-faqs

[218] http://www.researchdisclosure.com/, http://www.defensivepublications.org, http://www.linuxdefenders.org/

[219] An EFF Initiative to protect innovation and fight bogus patents: available at: https://www.eff.org/patent-busting

[220] EFF's Fight for Open 3D Printing Continues at Ask Patents, Fight for Open 3D Printing, Julie Samuels, March 18, 2013 Available at: https://www.eff.org/deeplinks/2013/03/effs-fight-open-3d-printing-continues-askpatentscom

sale and distribution of the invention. The defensive publication mechanism is perfectly in line with FLOSS, OSH theoretical and philosophical framework, in fact the inventor that decide to disclose his creation means a defensive publication, shares the knowledge and at the same time promotes distributed and grassroots innovation. Unfortunately, it is not possible to use this tool to bind further inventions based on the published one to be shared using the same tools, this because once an invention is published it enters the public domain.

Conclusions

There are thousands of different designs that are released in order to enable people to build their own machines, vehicles, tools, laptops and much more. The means of production of these objects are often partly or fully automated, making the documentation and source files key assets that can be shared, but in order to motivate inventors to share their knowledge a clear legal framework have to be provided to them. Due to the rapid evolution of technology and society it is widely acknowledged that the law maker is laking behind, and IP is one of the subject that more is suffering the pre-digital approach, instead of asking for a global reform of the IP legal instruments what is here proposed is to use redyly available legal instruments to promote freedom and openness in hardware.

To this end copyright applies well to the documentation and source file in general allowing the author to enforce restrictions such as copyleft and binding downstream recipients to share back improvements or to credit the original author of the digital files.

There are many hardware specific licenses that can fit different needs by granting different degree of restrictions over the shared work, there are also sector specific licenses such as the TDPL that can fit some specificity of source files of automatically manufactured objects. As a result, it is indispensable to make sure that makers and OSH developers understand what can be covered by copyright and what licenses are appropriate to use.

In parallel, initiatives such as defensive publications or royalty-free patent pooling should become common and widely used mechanisms, available to OSH developers so that patentable inventions are appropriately shared by means of a defensive publication, backed-up by a solid legal protection that prevents others to patent it.

On the other hand, initiatives such as Linux Defenders or Electronic Frontier Foundation Patent Boosting Project, have to grow involving more contributors and activist and spread in the hardware context this is likely to prevent patents to be granted for prior art, by acting as watchdogs over filed

patent applications.

Patent pooling and defensive patent licensing are also tools that are becoming more popular in the corporate world. These approaches positively influence individual and hobbyist hardware development, since they can access the OIN and possibly the future defensive patent network, this allows individuals as well as corporate user to benefit from patented technologies, in fact by becoming members of the pools users can use a very large patent portfolio royalty free, while being bounded to contribute back to the network.

The answer to the main research question posed in the beginning of this thesis is: the legal instruments that have to be used to protect OSH developers and users are mainly copyright, trademark and patents following the guidance presented in this thesis. Depending on the project a case by case evaluation will have to be performed by choosing the instruments and tune them so that they can best protect the right of the authors, motivate the community to contribute and at the same time be available for others to use, study, share and improve.

A set of easy to read IP management guidelines is required for individual and hobbyist OSH developers in order to enable them to keep the control over their creation and at the same time share them. A well drafted IP best practice guide for OSH developers should cover: trademark management and guidelines drafting, copyright licensing, and patent management. Patent management is an optional detail that have to be considered just in case of a patentable technical invention, in this case it can be centered on defensive publication as well as in patent sharing mechanisms based on the preferred approach. The overall aim of having a set of guidelines is to increase legal certainty for OSH developers and give them an easy way to understand how to manage intellectual property in their project.

As an appendix to this thesis a web-site will be published, this contains references and an easy to understand explanation on how the legal instruments described in this thesis works, namely it will be highlighted how other than licenses is important to use trademark and defensive publications.

The purpose of this website is to hopefully help makers and inventors that want to release the documentation of their projects to have a clear understanding of which instruments are available and the extent to which this instrument are fit to protect their creation. Hopefully the site will contribute to foster and structure the discussion around OSH legal framework, because OSH is a reality, and is here to stay.

Books:

- Andrew M. St. Laurent, Open Source and Free Software Licensing, O' really, 2005
- Michele Boldrin and David K. Levine, Against Intellectual Monopoly, Cambridge University Press, 2008
- Sir Thomas Edlyne Tomlins, John Raithby, The statutes at large of England and of Great Britain: from Magna Carta to the union of the kingdoms of Great Britain and Ireland, Volume 18. Available on: Google Books
- Di Vivien Irish, Intellectual Property Rights for Engineers, 2nd Edition. Available on: Google Books
- Arnoud Engelfriet, A History Of Foss Law And Licensing. The book is freely accessible at: http://ifosslawbook.org
- Lawrence Lessig , "Free culture : how big media uses technology and the law to lock down culture and control creativity ",Penguin Group (USA) Inc., New York , 2004
- WIPO Intellectual Property Handbook:Policy, Law and Use, WIPO 2004 Second Edition, Wipo Publication no. 489
- Bénédicte Fauvarque-Cosson,Denis Mazeaud, European Contract Law: Materials for a Common Frame of Reference: terminology, guiding principles, model rules, Munich : Sellier. European Law Publishers, 2008.

Journals articles:

- Joseph Feller, Brian Fitzgerald, Scott A. Hissam and Karim R. Lakhani (editors), 2007. *Perspectives on free and open source software*. Cambridge, Mass.: MIT Press.
- Ye, Huojie and Zhong, Shuhua, Business Accelerator Network: A Powerful Generator of Strategic Emerging Industries (August 16, 2012). OIDA International Journal of Sustainable Development, Vol. 04, No. 06, pp. 11-23, 2012. Available at SSRN: http://ssrn.com/abstract=2130921
- Bonaccorsi, Andrea and Rossi Lamastra, Cristina, Licensing Schemes in the Production and Distribution of Open Source Software: An Empirical Investigation. Available at SSRN:http://ssrn.com/abstract=432641
- ALIPRANDI, S.. Interoperability and open standards: the key to a real openness. International Free and Open Source Software Law Review, North America, 3, sep. 2011.

Available at: http://www.ifosslr.org/ifosslr/article/view/53.

- MALINEN, Tiina et al. Community created open source hardware: A case study of "eCars - Now!". First Monday, [S.l.], apr. 2011. ISSN 13960466. Available at: <http://firstmonday.org/ojs/index.php/fm/article/view/3357/2951>.
- Eli Greenbaum, Three-Dimensional Printing and Open Source Hardware, New York University Journal Of Intellectual Property And Entertainment Law, Available: <jipel.law.nyu.edu/2013/04/three-dimensional-printing-and-open-source-hardware/>.
- BESSEN, James; MEURER, Michael J.. What's wrong with the patent system? Fuzzy boundaries and the patent tax. First Monday, [S.l.], jun. 2007. ISSN 13960466. Available at: <http://www.firstmonday.dk/ojs/index.php/fm/article/view/1867/1750>.
- Wilbanks, J.T.; Wilbanks, T.J. Science, Open Communication and Sustainable Development. Sustainability 2010, 2, 993-1015. Available at: www.mdpi.com/2071-1050/2/4/993
- Chestek, P.. Who owns the project name?. International Free and Open Source Software Law Review, North America, 5, nov. 2013. Available at: <http://www.ifosslr.org/ifosslr/article/view/87>.
- Robert W. Gomulkiewicz, Open Source License Proliferation:Helpful Diversity or Hopeless Confusion?, Available at: <law.wustl.edu/Journal/30/Gomulkiewicz.pdf>.
- AYASS, M., SERRANO, J.. The CERN Open Hardware Licence. International Free and Open Source Software Law Review, North America, 4, may. 2012. Available at: <http://www.ifosslr.org/ifosslr/article/view/65>.
- Katz, A.. Towards a Functional Licence for Open Hardware. International Free and Open Source Software Law Review, North America, 4, may. 2012. Available at: http://www.ifosslr.org/ifosslr/article/view/69.
- BAIN, M.. Software Interactions and the GPL. International Free and Open Source Software Law Review, North America, 2, feb. 2011. Available at: <http://www.ifosslr.org/ifosslr/article/view/44>.
- Guadamuz, Andrés, Viral Contracts or Unenforceable Documents? Contractual Validity of Copyleft Licenses. E.I.P.R. Vol. 26, Issue 8, pp.331-339, 2004. Available at <SSRN: http://ssrn.com/abstract=569101>.
- Schultz, Jason and Urban, Jennifer M., Protecting Open Innovation: The Defensive Patent License as a New Approach to Patent Threats, Transaction Costs, and Tactical Disarmament (April 16, 2012). Harvard Journal of Law and Technology, Vol. 26, 2012. Available at SSRN: <http://ssrn.com/abstract=2040945>.

• Doug Lichtman & Mark. A. Lemley, Rethinking Patent Law's Presumption Of validity, StanfordLaw Review, Volume 60, Issue 1, Available at: http://www.researchgate.net/publication/228188392_Rethinking_Patent_Law%27s_Presumption_of_Validity/links/02e7e51eaa79346c23000000>.

Informative articles:

- W3tech : "Usage statistics and market share of Unix for websites". Accessed, 19 Feb. 2014. Available at: http://w3techs.com/technologies/details/os-unix/all/all
- List of OSH available at: http://p2pfoundation.net/Product_Hacking
- What is Copyleft? available at: https://www.gnu.org/copyleft/copyleft.en.html
- Bre Pettis, September 24, 2012 Let's try that again, Available at: www.makerbot.com/blog/2012/09/24/lets-try-that-again/
- Zach Hoeken, MakerBot vs. Open Source – A Founder Perspective, September 21, 2012 Available:http://www.hoektronics.com/2012/09/21/makerbot-and-open-source-a-founder-perspective/
- Alan Wlasuk, Security Through Obscurity? Don't Count On It., Available at: www.securityweek.com/security-through-obscurity-dont-count-it
- David A. Mellis, "The moral imperative for open-source hardware", 07 OCTOBER 2010 Available at:http://dam.mellis.org/2010/10/the_moral_imperative_for_open-source_hardware/#comments1
- Jon Brodkin, Arduino creator explains why open source matters in hardware, too, Available at: arstechnica.com/information-technology/2013/10/arduino-creator-explains-why-open-source-matters-in-hardware-too/
- David A. WheelerWhy Open Source Software / Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers!, Available: www.dwheeler.com/oss_fs_why.html
- Björn Schießle, "Free Software, Open Source, FOSS, FLOSS - same same but different", 2012, Available at: http://fsfe.org/freesoftware/basics/comparison.en.html
- Usage of web servers for websites, Available at: w3techs.com/technologies/overview/web_server/all
- Why Open Source misses the point of Free Software, Richard Stallman, 26/11/2013. Available at: http://www.gnu.org/philosophy/open-source-misses-the-point.html
- What is free software? (The Free Software Definition),Available

at:http://www.gnu.org/philosophy/free-sw.html

- Defining “Noncommercial”A Study of How the Online PopulationUnderstands “Noncommercial Use”September 2009. Available at: http://governancexborders.com/2009/09/16/standardizing-via-polling-creative-commons%E2%80%99-study-on-its-noncommercial-clause/
- Richard M. Stallman, Did You Say “Intellectual Property”? It's a Seductive Mirage, Available at: www.gnu.org/philosophy/not-ipr.xhtml
- Cory Doctorow, "Intellectual property" is a silly euphemism, Available at: www.theguardian.com/technology/2008/feb/21/intellectual.property
- Frank Thadeusz, No Copyright Law: The Real Reason for Germany's Industrial Expansion? Available at: www.spiegel.de/international/zeitgeist/no-copyright-law-the-real-reason-for-germany-s-industrial-expansion-a-710976.html
- Julie Samuels, EFF’s Fight for Open 3D Printing Continues at Ask Patents, Available at: https://www.eff.org/deeplinks/2013/03/effs-fight-open-3d-printing-continues-askpatentscom
- Wilbanks, J.T.; Wilbanks, T.J. Science, Open Communication and Sustainable Development. Sustainability 2010, 2, 993-1015. Available at: www.mdpi.com/2071-1050/2/4/993
- Institute of Intellectual Property, Ch2, The History And Development Of Trademark Law Dr. Shoen Ono, Available at: www.iip.or.jp/translation/ono The History And Development Of Trademark Law
- Lawrence Rosen , License Proliferation , Available at: http://rosenlaw.com/pdf-files/
- Robert W. Gomulkiewicz, Open Source License Proliferation:Helpful Diversity or Hopeless Confusion?, Available at: law.wustl.edu/Journal/30/Gomulkiewicz.pdf
- David A. Wheeler, The Free-Libre / Open Source Software (FLOSS) License Slide, available at: www.dwheeler.com/essays/floss-license-slide.html
- Bradley M. Kuhn and Richard M. Stallman, Freedom or Power?, Available at: https://www.gnu.org/philosophy/freedom-or-power.html#f1
- 2013-10-11, Maik Außendorf, Why have you started a fork from bacula.org? Available at: http://www.bareos.org/en/faq/items/why_fork.html
- OSHWA community survey datas are available at: http://www.oshwa.org/oshw-community-survey-2013/
- Open Hardware at CERN, Mark Johnson, September 23, 2013 Available at: osswatch.jiscinvolve.org/wp/2013/09/23/open-hardware-at-cern/
- Pamela Jones, The GPL Is a License, not a Contract, available at:

http://lwn.net/Articles/61292/

- Jeff Neuburger, Ninth Circuit Rules on License Conditions versus Contract Covenants in Dispute over World of Warcraft Bots – MDY v. Blizzard, Part I , Available at: newmedialaw.proskauer.com/2011/01/03/ninth-circuit-rules-on-license-conditions-versus-contract-covenants-in-dispute-over-world-of-warcraft-bots-mdy-v-blizzard-part-i/
- Aaron Williamson, Versata Court: Breach-of-Copyleft Claim not Preempted by Copyright Act, Available at: https://torekeland.com/blog/versata-copyleft-case
- Joris Peeters, General Public License in Court - Analyses of the case law in EU countries Available at: https://www.law.kuleuven.be/jura/art/44n4/peeters.html
- Jarkko Moilanen e Tere Vadén, Manufacturing in motion: first survey on 3D printing community, Statistical Studies of Peer Production. Availabe at: surveys.peerproduction.net/2012/05/manufacturing-in-motion/
- Sean Gallagher, How Facebook threatens HP, Cisco, and more with its “vanity free” servers, arstechnica.com/information-technology/2013/07/how-facebook-is-killing-the-hardware-business-as-we-know-it/
- Rich Miller, Open Compute Gets Down to Business, With New License and Certifications, February 3, 2014 Available at: www.datacenterknowledge.com/archives/2014/02/03/open-compute-gets-business-new-license-certifications/
- Alicia Gibb, On Creative Commons and Open Source, Available at: www.oshwa.org/2014/05/21/cc-oshw/
- Richard M. Stallman, Don't Let ‘Intellectual Property’ Twist Your Ethos, Available at: https://www.gnu.org/philosophy/no-ip-ethos.htm
- Giuditta Cordero Moss, Lectures On Comparative Law Of Contracts, Available at: folk.uio.no/giudittm/PCL_Vol15_3[1].pdf
- New Uses for Patent Pools, Bart Showalter and Trampas Kurth Available at: www.iptoday.com/articles/2008-9-showalter.asp
- Elon Musk, All Our Patent Are Belong To You, published on June 12, 2014 Available at: http://www.teslamotors.com/blog/all-our-patent-are-belong-you
- Tesla’s Patent Giveaway: Please Use Our Batteries, Available at: blogs.wsj.com/corporate-intelligence/2014/06/12/tesla-patent-giveaway-please-use-our-batteries/
- R.E. Lord,Is Tesla Motors a Good Stock to Buy?, Available at:www.prognog.com/investing/clean-tech/is-tesla-motors-a-good-stock-to-buy.html

- FSF is fermly against software patents see: Richard Stallman, Software patents — Obstacles to software development, Available at: https://www.gnu.org/philosophy/software-patents.html
- EFF's Fight for Open 3D Printing Continues at Ask Patents, Fight for Open 3D Printing, Julie Samuels, March 18, 2013 Available at: https://www.eff.org/deeplinks/2013/03/effs-fight-open-3d-printing-continues-askpatentscom