Open Hardware

Lecture 2

TU Wien, 193.067 Free and Open Technologies (WS 2019/2020)
Christoph Derndorfer and Lukas F. Lang

Organization

- Lectures:
  - Weekly lecture to cover course materials (until December)
  - Lectures take place on Tuesdays, 17:00–19:00, Argentinierstraße 8, Seminarraum/Bibliothek 194-05
  - Attendance is mandatory, **de-registration possible until Friday, October 18, 23:59 CET**
- Group project:
  - In groups of 4 students
  - 3 meetings with lecturers during semester: **Fridays, November 8/November 29/January 10, 13:00–18:00**
  - Final presentations at the end of January: **Friday, January 24, 2020, 12:30-16:30**
- Final paper:
  - In groups of 2 students
  - Final presentations at the end of January: **Friday, January 31, 2020, 12:30–16:30**
  - Deadline: **Sunday, February 9, 2020, 23:59 CET** (no exceptions!)

Organization

- Grading:
  - 50% group project
  - 35% seminar paper
  - 15% participation during lectures
  - All course components need to be passed in order to pass the overall course!
- Course materials:
  - Will be provided at https://free-and-open-technologies.github.io
- For further questions:
  - Email christoph.derndorfer@tuwien.ac.at and lukas.f.lang@tuwien.ac.at

Lecture outline

1. ~~October 8, 2019: FLOSS (Free/Libre and Open Source Software)~~
2. **October 15, 2019: Open Hardware**
3. October 22, 2019: Open Data
4. October 29, 2019: Open Content/Open Educational Resources
5. November 5, 2019: Open Science/Research
6. November 12, 2019: Open Access
7. November 19, 2019: Open Spaces/Open Practices at Metalab Vienna
   - Location: Metalab, **Rathausstraße 6, 1010 Vienna**
8. November 26, 2019: Guest Lecture: Stefanie Wuschitz (Mz* Baltazar's Lab)

Free/Libre and Open Source Software

Recap & Catch-Up

Recap

- Definitions: FSF/OSI, free software / open source software
- Key figures: Stallman, Torvalds, Raymond, Peterson
- History of FLOSS and historic context
- Copyright/Copyleft
- Licenses (MIT, Apache, GPL, Creative Commons)
- Practical issues
- Contributors & incentives
- Open source as a business model
- How to contribute

Open source project metrics

- Number of different committers
- Regularity and size of contributions by committers
- Duration until contributions/pull requests are merged
- Response time to issues
- Number of open issues
- Activity on mailing lists and other communication channels
- Release cycle
- Availability of roadmap
- Availability of documentation

[1] Open Source Metrics
[2] Measuring Your Open Source Program's Success

Criticism of open source projects and communities

Accessibility:

> “Despite the rhetoric surrounding Open Source, which basically argues that ‘anybody can contribute,’ it seems instead that only those few participants who have managed to define and present themselves as ‘‘software craftsmen’’ eventually reach the status of developer in a project.” [1]

Inclusion:

> “The open source community needs to show less discrimination and more inclusion to tone down the male-dominated atmosphere into something that promotes participation and not strictly individual work.” [2]

[1] Ducheneaut, N. Comput Supported Coop Work (2005) 14: 323. https://doi.org/10.1007/s10606-005-9000-1
[2] Powell, W.E., Hunsinger, D.S., and Medlin, B.D. Gender Differences within the Open Source Community: An Exploratory Study. Journal of Information Technology 21, 4 (2010), 29-37.

Criticism of open source projects and communities

Participation and bias:

> “Our results show that women’s pull requests tend to be accepted more often than men’s, yet women’s acceptance rates are higher only when they are not identifiable as women.” [1]

> “...plausible explanations include the presence of gender bias in open source, survivorship and self-selection bias, and women being held to higher performance standards.” [1]

> “According to our success measures [women] are also less successful on [Stack Overflow] than men.” [2]

> “While women ask more questions than men, men are more active in all other forms of activity.” [2]

[1] Terell et al., Gender Differences and Bias in Open Source: Pull Request Acceptance of Women Versus Men, PeerJ Computer Science 3:e111 (2017).
[2] May, A., Wachs, J., and Hannák, A., Gender differences in participation and reward on Stack Overflow, Empir Software Eng (2019) 24: 1997.

Open Hardware

Free and Open Applied to Atoms

What is open (source) hardware?

“Open source hardware is hardware whose design is made publicly available so that anyone can study, modify, distribute, make, and sell the design or hardware based on that design.” [1]

OSHWA defines hardware as: “tangible artifacts - machines, devices, or other physical things”

This includes: electronic hardware, machine tools, vehicles, medical equipment, textiles, etc.

[1] Open Source Hardware Association, “Definition (English)”, accessed 2019/10/10

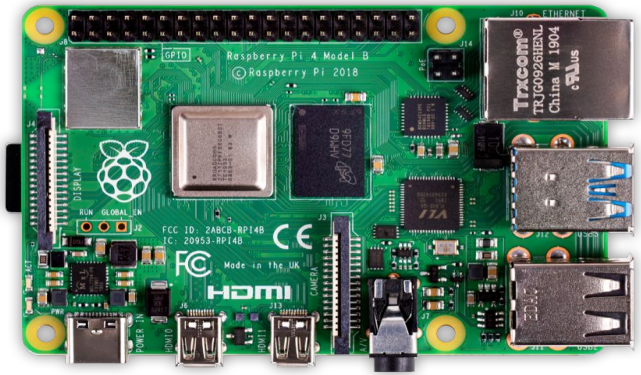
Open hardware?

Arduino

Yes: CC-BY-SA
But: Operated by a company (Arduino AG)

Raspberry Pi

No: Only schematics available under CC-BY-SA (and Raspbian OS is also FLOSS)
But: Operated by a non-profit foundation (Raspberry Pi Foundation)



[1] Arduino, ARDUINO UNO REV3 from https://store.arduino.cc/arduino-uno-rev3
[2] Raspberry Pi Foundation, Raspberry Pi 4: https://www.raspberrypi.org/

But what is open (source) hardware?

Software

- Source code (incl. comments)
- e.g. Linux kernel [1]

```
30  void cpufreq_add_update_util_hook(int cpu, struct update_util_data *data,
31                          void (*func)(struct update_util_data *data, u64 time,
32                                       unsigned int flags))
33  {
34          if (WARN_ON(!data || !func))
35                  return;
36
37          if (WARN_ON(per_cpu(cpufreq_update_util_data, cpu)))
38                  return;
39
40          data->func = func;
41          rcu_assign_pointer(per_cpu(cpufreq_update_util_data, cpu), data);
42  }
```

Hardware

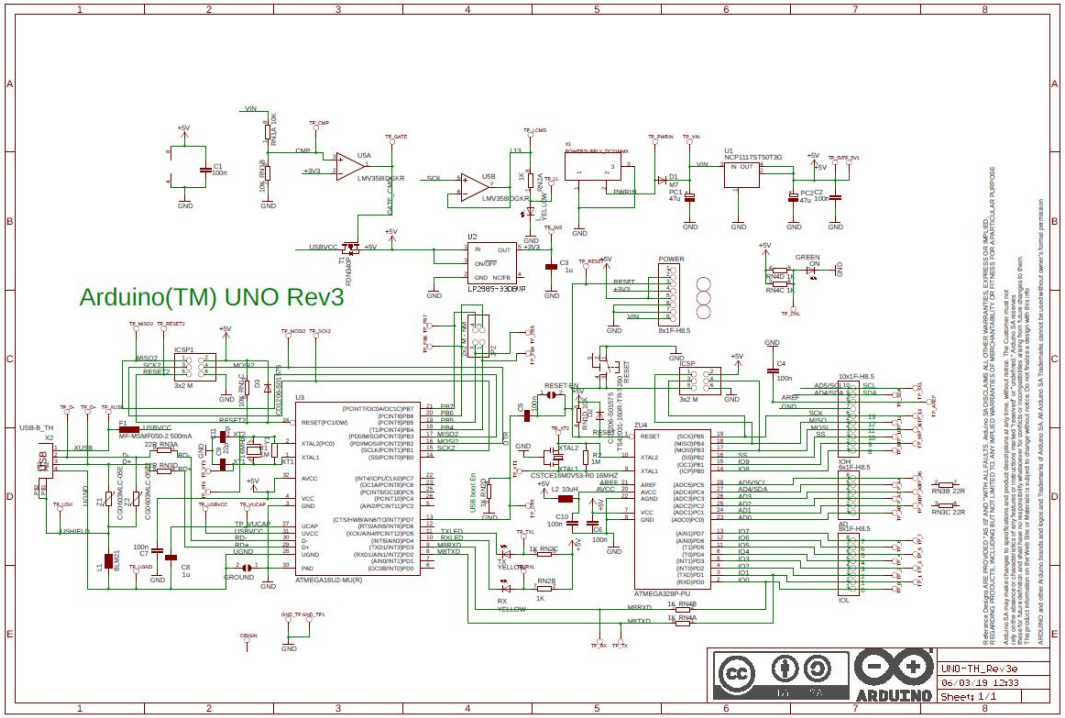
- Schematics
- PCB layouts
- Firmware (actually software but needed for hardware)
- Drivers (actually software but needed for hardware)
- e.g. Arduino UNO [2]

What else needs to be considered “source”?

Arduino Uno is open-source hardware! You can build your own board using the following files:

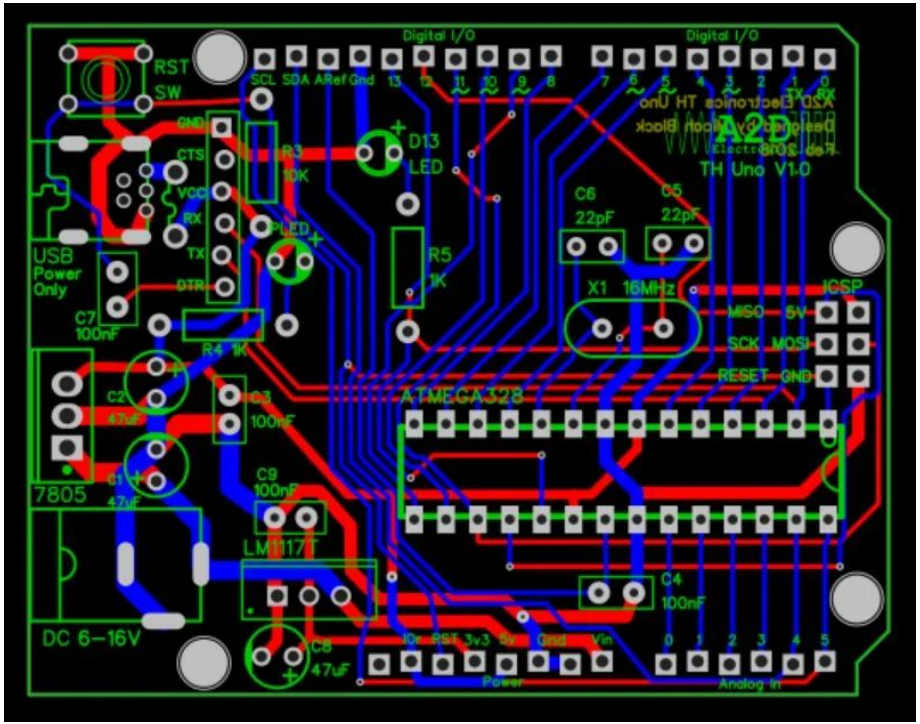
[1] GitHub, Linux kernel on GitHub, accessed 2019/10/11
[2] Arduino, ARDUINO UNO REV3 - Documentation, accessed 2019/10/11

Arduino UNO REV3 Schematics



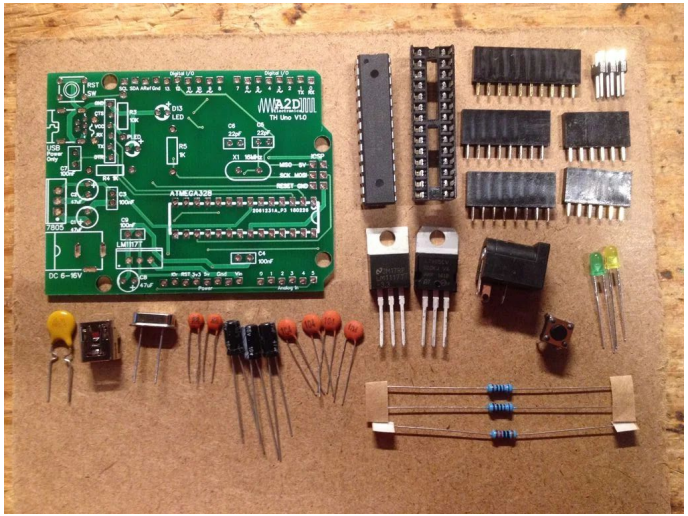
[1] Arduino UNI REV3 Schematics, accessed 2019/10/13

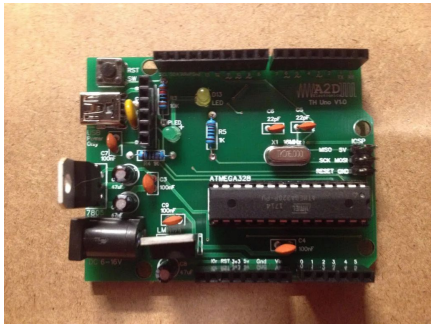
Arduino UNO REV3 PCB design

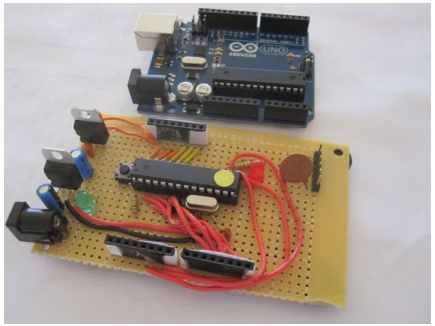


[1] Instructables, Building a DIY Arduino on a PCB and Some Tips for Beginners, accessed 2019/10/13

Build your own Arduino...







[1] Instructables, Building a DIY Arduino on a PCB and Some Tips for Beginners, by A2D Electronics, CC BY-NC-SA 4.0, accessed 2019/10/13
[2] Instructables, How to Make Your Own Arduino Board, by Higgs Boson, CC BY-NC-SA 4.0, accessed 2019/10/13

What do you do when your <insert product> breaks in Zambia?

- No 0800 number to call
- Maybe even no local reseller

Solution: Fix it yourself!

- Needed:
  - A device that can be fixed ✓
  - Documentation ✓
  - Tools ✓

[1] Christoph Derndorfer, CC BY-SA 4.0
[2] Christoph Derndorfer, CC BY-SA 4.0

Key figures

Bruce Perens

- Co-founder of the Open Source Initiative (OSI) with Eric S. Raymond in 1998
- Launched Open Hardware Certification Program in 1997

Massimo Banzi

- One of the co-founders of Arduino while working at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy (2003)

[1] By Friprog - derivative (loss-less jpeg crop) of https://commons.wikimedia.org/wiki/File:Bruce_perens_2009.jpg, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=47042743
[2] Di David Cuartielles - https://www.flickr.com/photos/mbanzi/5537363651/in/album-72157626169958305/, CC BY-SA 2.0, https://commons.wikimedia.org/w/index.php?curid=52312556

Key figures

Limor Fried

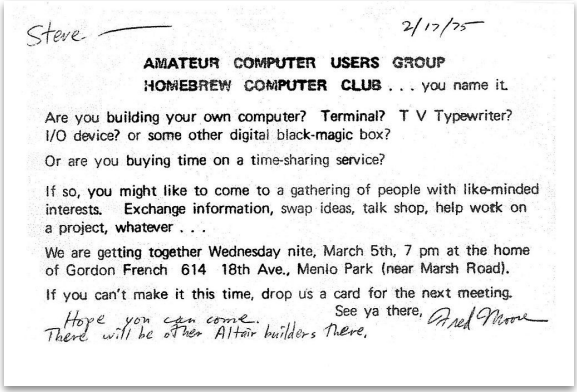
- BSc and MEng from MIT
- Founder and owner of open hardware manufacturer and vendor Adafruit Industries in 2005

Eben Upton

- Technical director at Broadcom
- One of co-founders of the Raspberry Pi foundation in 2009

[1] By TechCrunch - 167725033GH00048_TechCrunch, CC BY 2.0, https://commons.wikimedia.org/w/index.php?curid=39644510
[2] By Jim Killock - https://www.flickr.com/photos/jimkillock/7862804896/, CC BY-SA 2.0, https://commons.wikimedia.org/w/index.php?curid=34021784

Brief history


Steve — 2/17/75

AMATEUR COMPUTER USERS GROUP
HOMEBREW COMPUTER CLUB . . . you name it.

Are you building your own computer? Terminal? T V Typewriter? I/O device? or some other digital black-magic box?

Or are you buying time on a time-sharing service?

If so, you might like to come to a gathering of people with like-minded interests. Exchange information, swap ideas, talk shop, help wotk on a project, whatever . . .

We are getting together Wednesday nite, March 5th, 7 pm at the home of Gordon French 614 18th Ave., Menlo Park (near Marsh Road).

If you can't make it this time, drop us a card for the next meeting.

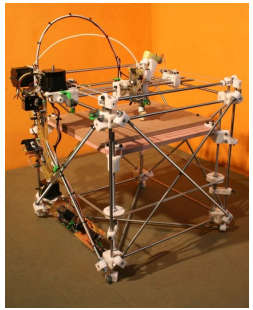
Hope you can come. See ya there, Fred Moore
There will be other Altair builders there.

- 1970s: “Give to help others.” [1]
  - Availability of first PCs: e.g. Altair 8800 (1974)
  - Homebrew Computer Club founded in Menlo Park, CA in 1975
  - “Without computer clubs there would probably be no Apple computers.” [1]
  - Steve Jobs and Steve Wozniak met and first introduced the Apple I at HCC in 1976
    - Apple I was sold as a DIY kit without components such as the power supply, keyboard, display, and enclosure
- 1980s: Commercialization
  - PCs become more widespread: Commodore Amiga, IBM PC, etc.
- 1990s: First open hardware initiatives [2]
  - Bruce Perens (OSI) launched Open Hardware Certification Program in 1997
  - David Freeman starts the Open Hardware Specification Project (OHSpec) – now defunct
  - Diehl Martin starts FreeIO initiative and website to discuss and share free hardware

[1] Stephen Wozniak, “Homebrew and How the Apple Came to Be”, accessed 2019/10/12
[2] Wikipedia, “Open-source hardware”, section “History”, accessed 2019/10/11
[3] By Gotanero - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=29592690

Brief history

- 2000s: Notable projects/products/companies emerge
  - Arduino: Single-board microcontroller, Interaction Design Institute Ivrea, Ivrea (Italy), 2003
    - Dozens of widely used microcontroller boards
  - SparkFun Electronics: electronics vendor and open hardware manufacturer, Colorado (USA), 2003
    - 140 employees and sells 2000 open-source components
  - RepRap: low-cost 3D printer, University of Bath (UK), 2005
    - One of the first low cost 3D plastic printers
  - Adafruit Industries: produces & sells open hardware products, NYC (USA), 2005
    - More than 100 employees, $45 million revenue (2016)
- 2010s
  - Raspberry Pi: single-board computer, Cambridge (UK), 2012
    - 25 million boards sold until 2019 (not open hardware, but open ecosystem)



[1] The original uploader was CharlesC at English Wikipedia. – Transferred from en.wikipedia to Commons. Transfer was stated to be made by User:Pac72., CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=3343501

Licenses

- FLOSS: Copyleft
- Hardware: Considerations for copyright (reproduction of works) and patents (reproduction of invention)
  - Intention of both: exclusive but time-limited right of creator to benefit from original work

- “Note that copyright (on which most licenses are based) doesn’t apply to hardware itself, only to the design files for it…” [1]
- “Hardware design documentation includes schematic diagrams, designs, circuit or circuit-board layouts, mechanical drawings, flow charts and descriptive texts, as well as other explanatory material.” [2]

[1] OSHWA, "Best Practices for Open-Source Hardware 1.0", accessed 2019/10/13
[2] CERN, “CERN launches Open Hardware initiative”, 2011

Licenses

- Re-use of FLOSS licenses:
  - Creative Commons Attribution Share-Alike (e.g. Arduino)
  - GPL (e.g. RepRap)
    - Stallmann in 1999: “Because copying hardware is so hard, the question of whether we're allowed to do it is not vitally important. I see no social imperative for free hardware designs like the imperative for free software. Freedom to copy software is an important right because it is easy now - any computer user can do it. Freedom to copy hardware is not as important, because copying hardware is hard to do.” [1]

→ Key difference between FLOSS and open hardware: **economics of physical goods**: cost, minimum quantities, logistics, feasibility, etc.

[1] Linux Today, Richard Stallman -- On "Free Hardware", accessed 2019/10/13

Licenses

- Open hardware specific licenses:
  - CERN Open Hardware Licence [1]
  - TAPR Open Hardware License [2]

- Certification approach:
  - OSHWA Certification: “An easy and straightforward way for producers to indicate that their products meet a uniform and well-defined standard for open-source compliance” [3].
  - Currently ~330 projects have been certified:
    - E.g. Arduino and Raspberry Pi add-ons, see https://certification.oshwa.org/list.html

[1] OHWR, "CERN Open Hardware License", accessed 2019/10/12
[2] TARP, “The TAPR Open Hardware License”, accessed 2019/10/12
[3] OSHW, “OSHWA Certification”, accessed 2019/10/12

OSHWA certification requirements

To meet the requirements for OSHWA self-certification:

1. Create products that comply with the OSHWA definition.
2. Ensure that all of the creator's own contributions to an open source product using the certification mark are shared as open source in accordance with the agreement and these requirements.
3. Ensure all parts within the creator's control are open source. Use best efforts to distinguish any third-party proprietary components. **Third-party components such as chips must have fully accessible and shareable datasheets for hardware to be considered open source.**
4. Self-certify each project by completing the online license form - yearly renewal.
5. Register with OSHWA each unique product bearing the Certification Mark, and otherwise abide by the usage guidelines.

[1] OSHWA, "Requirements for Certification", accessed 2019/10/11

Open hardware without a (western) legal framework

[1] Shenzhen: The Silicon Valley of Hardware (Full Documentary) | Future Cities | WIRED, accessed 2019/10/12

Levels of open hardware

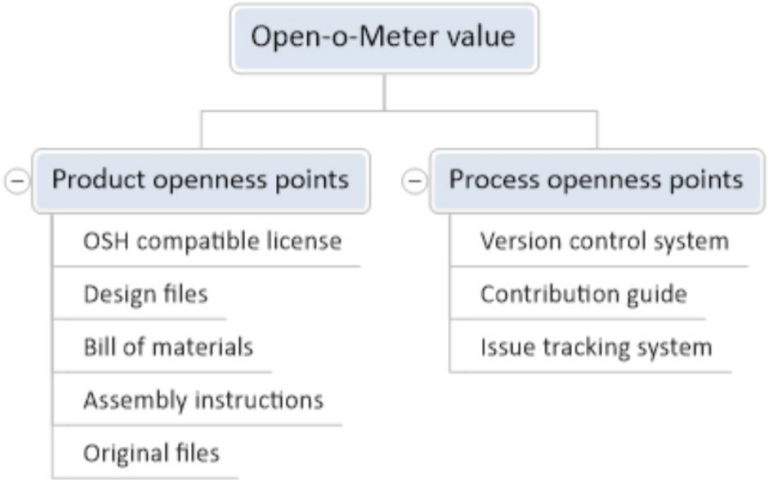
- Fully open:
  - Example: Arduino (hardware, firmware, drivers, software)
- Partially open:
  - Example: Raspberry Pi (hardware schematics, software, documentation open; but firmware, hardware details not open, especially Broadcom ARM chip)
- Closed hardware, but (partially) open drivers:
  - Examples: AMD, Nvidia graphics cards (more or less)
- Closed hardware, no open drivers:
  - Example: Baseband chips (exception: Openmoko phone in 2009), but projects working on it: Seeeduino GPRS [1], Osmocom [2], etc.

→ No mass-produced open hardware phone available (projects: ArduinoPhone 2.0 [3], DIY Cellphone [4])

[1] seeed, "Seeeduino GPRS", accessed 2019/10/12
[2] Osmocom, "Open Source Mobile Communications", accessed 2019/10/12
[3] Instructables, "Arduino Pgone 2.0", accessed 2019/10/13
[4] DIY Devices, "DIY Cellphone", accessed 2019/10/13

Levels of open hardware

- “What is a products’ source?”
- Open-o-Meter [1]



[1] Bonvoisin, J. and Mies, R. (2018), Measuring Openness in Open Source Hardware with the Open-o-Meter, Procedia CIRP, Volume 78.

Advice for building open hardware

- Product
  - Design files
  - Supporting documentation
- Process
  - Licenses
  - Toolchain
  - Practical aspects, e.g. common versioning of files and hardware revisions

- Resources:
  - Best Practices for Open-Source Hardware 1.0 [1]
  - A guide to Open Source Hardware [2]

[1] OSHWA, “Best Practices for Open-Source Hardware 1.0”, accessed 2019/10/13
[2] OPEN!, “A guide to Open Source Hardware”, accessed 2019/10/13

Practical issues

- Legal frameworks
  - More complex than with FLOSS
  - Theoretical vs. practical protections (e.g. hoverboard example from Shenzen clip or FaKey Makey issue [1])
- Economics of physical goods
  - Manufacturing, logistics, economies of scale, etc.
- Key components protected by patents and/or NDAs
  - Cellular / WLAN chips
  - Multimedia codecs
  - Drivers (binary blobs vs. source code)

[1] Michael Weinberg, "MaKey MaKey and the Limits of Open Source Hardware Licensing", accessed 2019/10/13

Criticism

- Hardly anything is really fully open hardware
  - Run:
    - Yes
  - Modify:
    - Yes, but how are you going to build an open hardware device yourself without the factory, chips, drivers, etc.?
  - Redistribute copies:
    - Distributing hardware is never going to be free like distributing software
  - Distribute modified versions:
    - See Modify and Redistribute points
- A lot of underlying components and technologies are patented
  - E.g. H.264 compression uses 100s of patents which makes it essentially impossible to build open hardware on/supporting them

Incentives

For companies:

- Monetary:
  - Free to obtain, commercialization
- No need to reinvent the wheel:
  - Allows to modify/adapt
  - Focus on differentiation
- Open hardware as de-facto industry standards to build on:
  - Arduino
  - Raspberry Pi (Compute Modules)
- Community:
  - Issue reports, feedback, etc.

For individuals:

- Altruistic
- Gain experience and knowledge
- Low cost components
- Community support
- But: Committing can be much harder then with software (patch committed, but how will you get new hardware revision?)

For academics:

- Allows others to use research
- Can be built upon
- Meeting their own needs (e.g. Arduino)

Open hardware as a business model

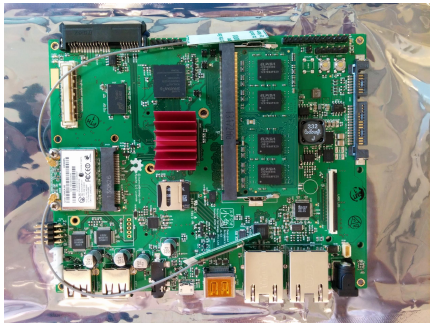
- Selling pre-assembled kits, boards, etc. instead of individual components:
  - Examples: Sparkfun, Adafruit, Raspberry Pi kits, etc. (and also Arduino)
- Branding:
  - Examples: “Arduino” is a trademark for official Arduino products, other products should use “Xxxxxx for Arduino” or “Xxxxxx (Arduino-Compatible)” in their name [1]
- Sales of related components:
  - Examples: Sparkfun, Adafruit, and many other stores sell cables, tools, etc.
- Sales of related software:
  - Examples: In 2011 Instructables.com was bought by Autodesk who sells software for 3D design (AutoCAD), PCB design (EAGLE), etc.

[1] Arduino, FAQs – “What should I call my boards?”, accessed 2019/10/12

Open hardware as a business model

- Ad-supported or subscription based content and documentation:
  - Examples: Instructables.com for DIY guides was funded with ads and sponsored contents
- Books and magazines:
  - Examples: Many books and magazines with Arduino, Raspberry Pi project ideas, guides, etc.
- Crowd-funding:
  - Examples: Crowd Supply crowdfunding platform for hardware projects with “user rights” (~open) for a percentage of campaign sales (in 2018 the company was acquired by Mouser Electronics, a distributors of electronics components → sales of related components)

Examples: Novena

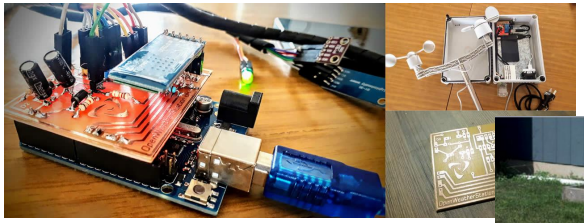


- Open-hardware and FLOSS-friendly computing platform:
  - Desktop, laptop, motherboard
  - Started in 2012 by Andrew “bunnie” Huang (“Hacking the Xbox”) and Sean “xobs” Cross
  - Launched on Crowd Supply in 2014 (Mainboard bundle for $250)
- Main goal: “We decided to produce a laptop that was as free as possible of closed-source embedded firmware.” [1]
- Main reason: “...be able to inspect and understand as much of the system and its components as we could, so if we came across bugs or other anomalous behavior, we could rely on [ourselves].”
- Open-source requirement influenced selection hardware:
  - CPU, battery controller, Wi-Fi module
  - E.g. ARM Freescale i.MX6 (system-on-a-chip) instead of Intel x86 microprocessors (accepts firmware updates that can’t be debugged or inspected)

[1] Huang, A. and Cross, S. (2015). “Novena: A Laptop With No Secrets: How we built a laptop with nothing but open-sourced hardware and software”, IEEE Spectrum, accessed 2019/10/14
[2] The open-source hardware laptop motherboard, Novena, by Autopilot, CC BY-SA 3.0

Examples: Arduino and Raspberry Pi

- Open Weather Station
- StereoPi
  - Stereoscopic camera based on RPi
- Many DIY/educational projects
  - Arduino project hub
  - Raspberry Pi projects, project books



Open Weather Station,
by Francisco Clariá, Apache license 2.0

- Commercial applications
  - Flightradar24.com uses RPi for data collection

Raspberry Pi projects book,
CC BY-SA-NC 3.0

Examples: litteBits

[1] YouTube, “Welcome To littleBits”, accessed 2019/10/14

Examples: MaKey Makey

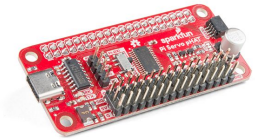
[1] YouTube, “MaKey MaKey: An Invention Kit for Everyone”, accessed 2019/10/14

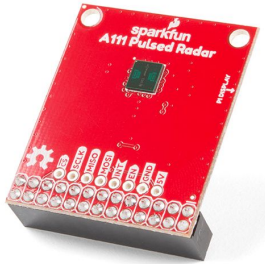
Available equipment

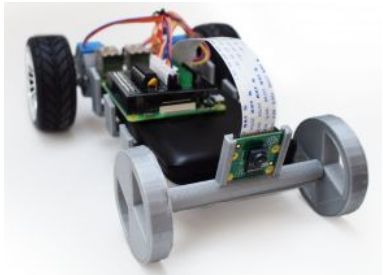
- Arduinos + Idiotware Shield + Ethernet Shield
- Raspberry Pi Zero WH
- MaKey MaKey
- Omega2
- littleBits Kit
- 3D printers
- Books, magazines, etc.

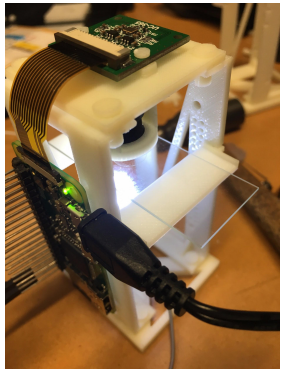
Open hardware project ideas

- Build a HAT (Hardware Attached on Top)
  - E.g. for Arduino or Raspberry Pi
  - Raspberry Pi stereoscopic camera
  - Open-source 1D computer interface, weather station
- Contribute to existing Austrian open hardware projects:
  - AutCar
  - OpenDrone
- Overlap with open research:
  - Open-source light tomography demo kit
  - Open-source microscopy
    - Super-resolution, 3D reconstruction, lensless microscope









Picroscope,
by RiksEddy, CC BY-NC-SA

See course website for example descriptions!

[1] SparkFun, SparkFun Servo pHAT for Raspberry Pi
[2] SparkFun, SparkFun Pulsed Radar Breakout - A111
[3] AutCar, AutCar - Model One (Dev Kit)

Upcoming tasks

- Next lecture: Open Data:
  - **Tuesday, October 22**: 17:00–19:00, Argentinierstraße 8, Seminarraum/Bibliothek 194-05
- Project group forming and topic selection:
  - **Friday, October 25**, via email to **both** lecturers
  - See project ideas at https://free-and-open-technologies.github.io
- First project meeting (45 min., discussion of your project idea):
  - **Friday, November 8, 13:00–18:00**, Argentinierstraße 8, **project room**
  - Will send out slots via termino.gv.at
- Paper group forming and topic selection:
  - **Friday, November 29**, via email to **both** lecturers

Literature/Resources

Alaejos, R. and Calvo, R. (2011), Arduino the Documentary

Dosemagen, S., Liboiron, M. and Molloy, J. (2017), Gathering for Open Science Hardware 2016. Journal of Open Hardware, 1(1), p. 4.

Powell, A. (2012), Democratizing production through open source knowledge: from open software to open hardware, Media, Culture & Society, 34(6), 691–708

Pearce, J. M. (2015) Quantifying the Value of Open Source Hardware Development. Modern Economy, 6, 1-11.