Enabling Practices: Unlearnings Checklist

Unlearning about institutions:

☐ The university expects that you’re going to patent something & make millions for them. It doesn't mean there aren't commercial opportunities when thinking through “ownership” in a different way, or through a different lens of the university's mission.

☐ Let’s unlearn staying under the radar because opening it may cause scrutiny, but that mindset keeps from connecting with other departments who can use it.

☐ Let’s unlearn that funders will scrutinize open hardware projects.

☐ Let’s unlearn that open hardware projects need to be big to be useful. Not all open hardware projects in academia aim to take over the world and that's okay.

☐ Let’s unlearn siloing and territorialism in the academic environment, cite projects with admiration.

☐ Stop believing it is ok to privatize publicly funded research through intellectual property.
We need to make clear there are ways for open hardware activity to be compatible with traditional academic metrics, for example, getting tenure.

Let's unlearn the norm that the usefulness of an open hardware project to current publications is critical for determining the level of support or attention it should receive. A project may be critical to publications later, or perhaps its usefulness might be greater in a different department within the university. Or maybe in another university altogether. That's why sharing and broad communication about open source hardware work is important.

Let's unlearn an open hardware project is worthless if you don't use it to produce a paper/thesis.

Everybody loves open hardware! There is something special about open hardware projects in academia compared to regular non-open hardware projects. It shows bravery to publish unfinished work, compassion to allow others to build upon, and courage to do something new. Let's be proud of that work.

Let's unlearn that all open hardware coming from academia comes from universities, versus other research institutes or labs like CERN, or nontraditional environments spaces like hacker spaces.

Unlearning ownership and control:

Let's unlearn that you have the right to prevent others from doing something because you had an idea.

Let's unlearn that you are the best one who can do the task at hand - people you work with are collaborators.

Adopting not just open designs, but the open tools that make them truly useful and accessible.

Stop assuming that open hardware is anti-commercial. Open hardware generates businesses and income.

Let's unlearn the engrained traditional role of patents in society.

Let's honor the difficult feelings of traditional clout in closed hardware: If your dad | PhD advisor | obnoxious aunt who used to work in tech in 1981 thinks you're giving away all your secrets, it can be hard to break free.
Let’s unlearn that open hardware applies to one domain i.e. Open source hardware is particularly well suited to electronics.

Let’s unlearn who can do open hardware i.e. You don’t have to be an engineer to develop open hardware project, and students/faculty can develop open hardware outside of STEM.

Unlearning about collaboration and sharing:

Let’s unlearn that copying is bad.

Let’s unlearn that we need perfection before sharing or waiting until the end of the project to share; let’s learn empathy for projects that aren’t complete.

Let’s unlearn siloing our data and experimental processes - not just the hardware itself, to enable reproducibility.

Think deeply about the value our projects give and how they can co-exist with “competitors”. Bring more value than the sum of the parts by working together through open hardware.

We need to remember that open hardware often has audiences beyond the original target. We need to remind the community that open hardware impact is bi-directional, it benefits the developer in addition to the adopter.

Unlearning about documentation:

Let’s unlearn that documentation is something you do *later* or *once*.

Let’s unlearn that documentation (outside of what goes into a publication) isn’t as valuable a contribution.

Good documentation is enough for projects to succeed, perfection is the enemy of done.