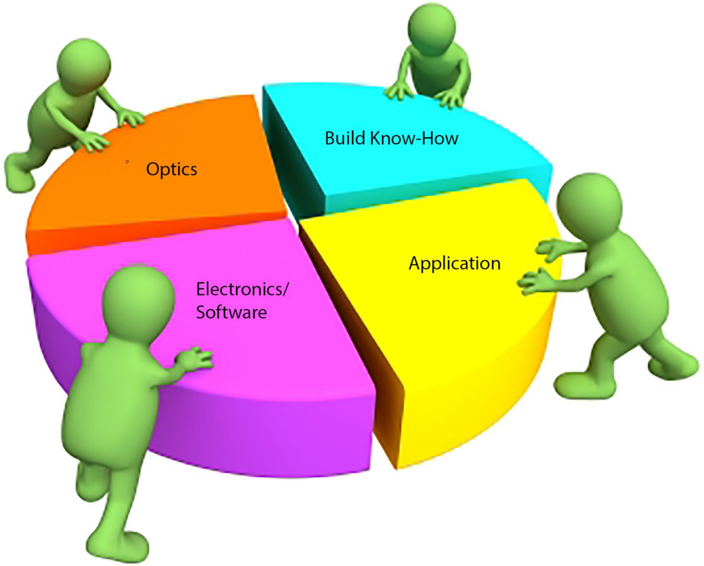
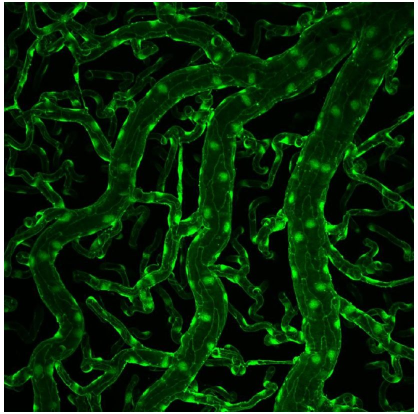
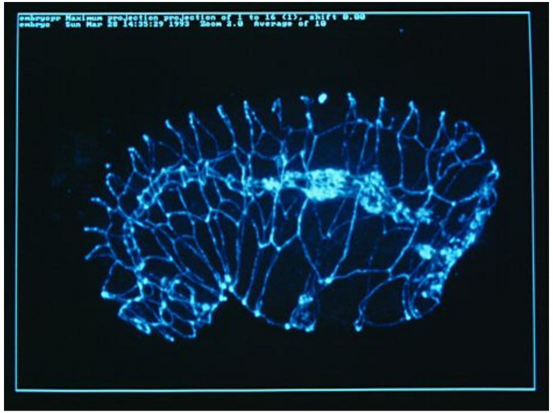
Open Hardware approaches for Laser Scanning Microscopy (LSM)







Kevin Eliceiri
Department of Biomedical Engineering and Medical Physics
Center for Quantitative Cell Imaging
Morgridge Institute for Research
University of Wisconsin-Madison
eliceiri@wisc.edu

Kevin Eliceiri lab at UW-Madison

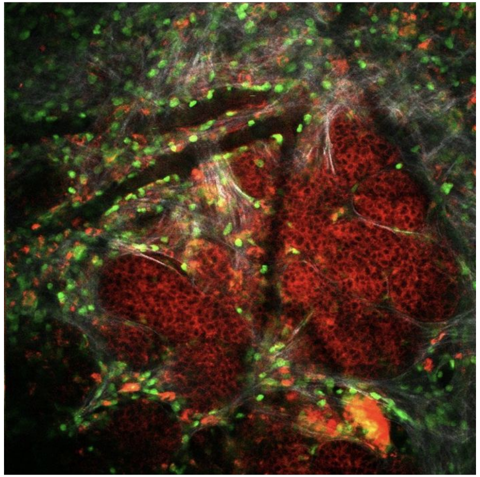
http://eliceirilab.org

a.k.a. Laboratory for Optical and Computational Instrumentation

µManager

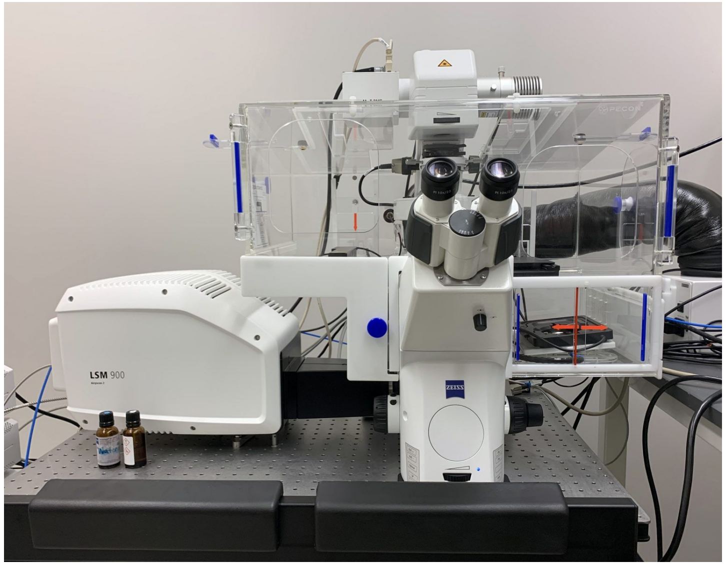
Open Source Microscopy Software

- Open Source Software Tools for Scientific Imaging
  - ImageJ, Fiji, Bio-Formats, Micro-Manager
- Imaging instrumentation for visualizing the cellular microenvironment in wound healing, cancer
  - Multiphoton, fluorescence lifetime microscopy, light sheet
- We believe in making our software and hardware tools accessible to everybody
- Experienced in open source software, but learning how to best do open microscopy hardware

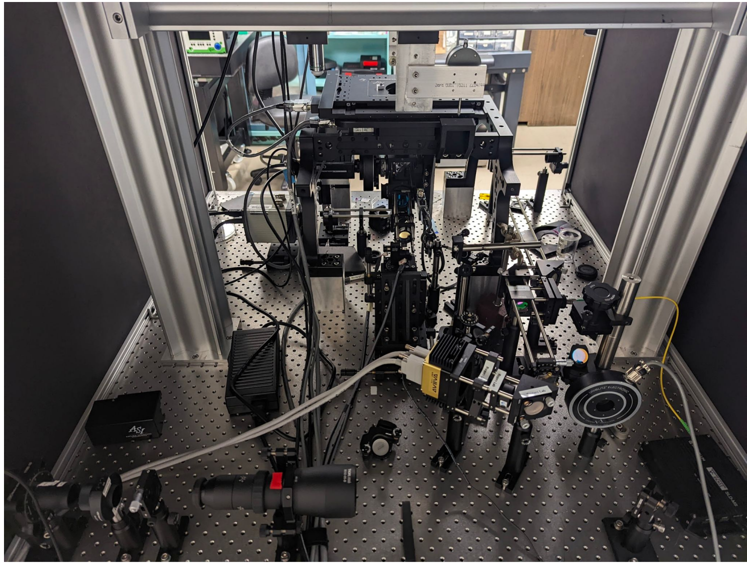


Typical Microscopes in biological research labs

(Instructional)



Commercial “turn-key”



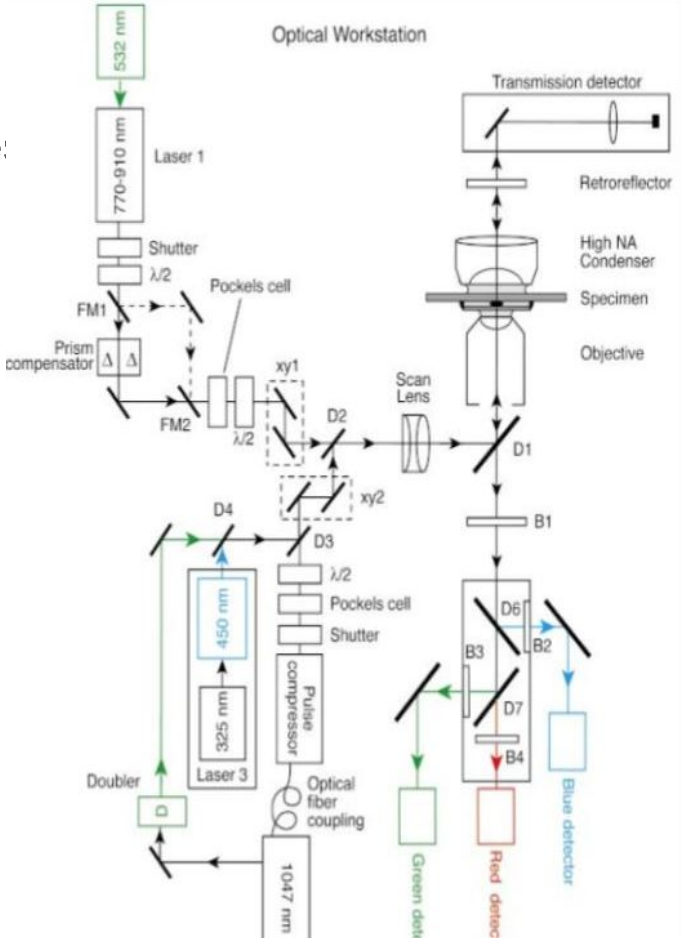
“Home-built”

“Home-built” (lab-built) microscopes drive the cutting edge

- Developing new imaging methods or techniques
- Testing modifications/improvements
- Setting up recently developed methods for biological applications

→ Need to share between research groups

- Typically: publish in a research journal (actual hardware details often scant-*see right)*
- How can open source hardware practices help here?
    - Emphasis on transparency, reproducibility (over cost, beginner-friendliness)



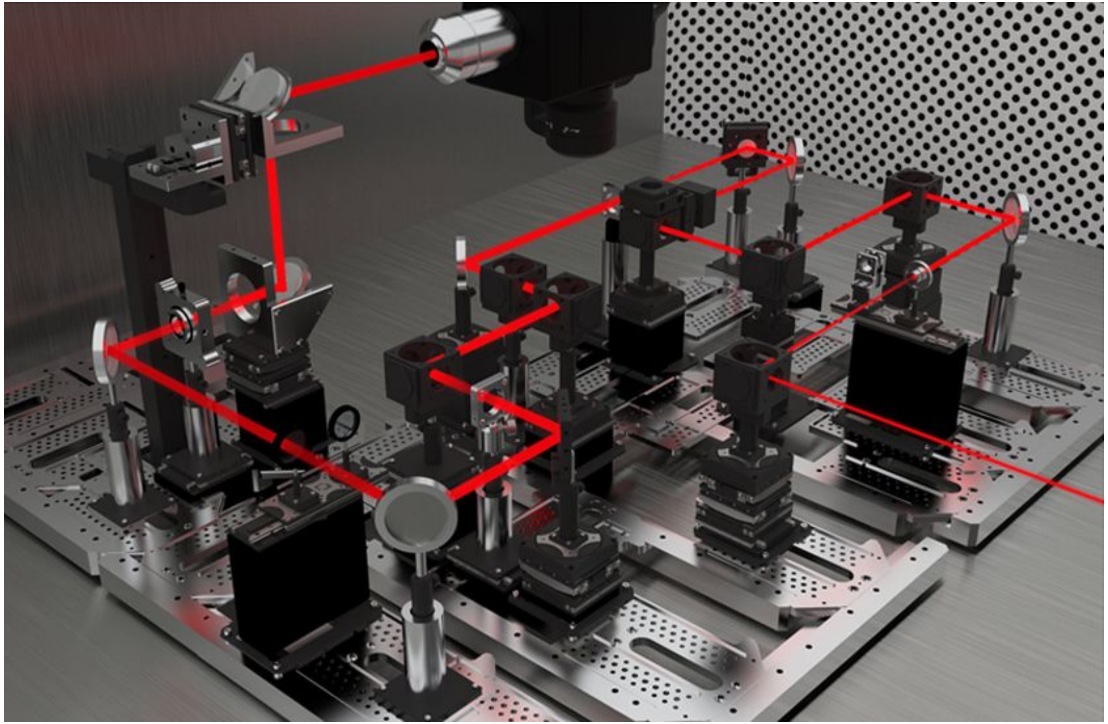
OpenScan:
software & hardware for laser scanning microscopy (LSM)

Laser — Power control — Polarization control — Scan mirrors & lens — Microscope stand & stage — Detectors — DAQ Electronics — PC & Software

- **Software**: modular, plugs into open source microscopy acquisition software (Micro-Manager)
- **Electronics**: currently off-the-shelf DAQ and TCSPC modules (pluggable)
- **Optics**: home-built optical breadboard systems

- Originally developed for use in lab and close collaborators on campus
- LSM is a *collection* of techniques
- How to cater to interest from others in the field?

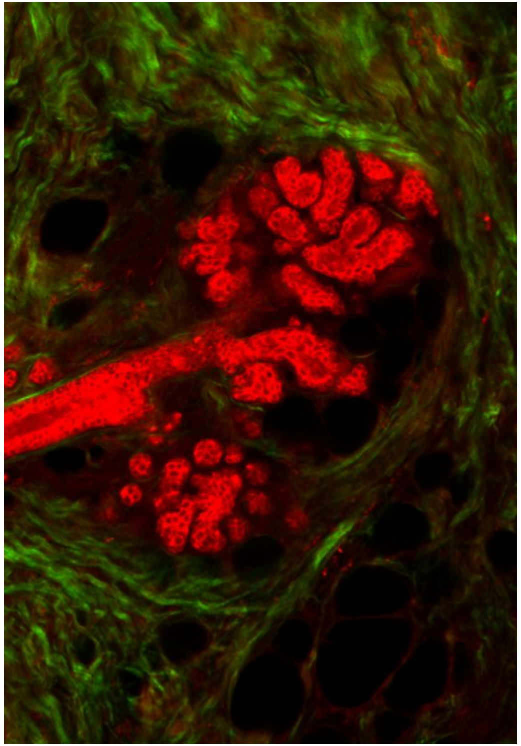
Optical Systems are complex:



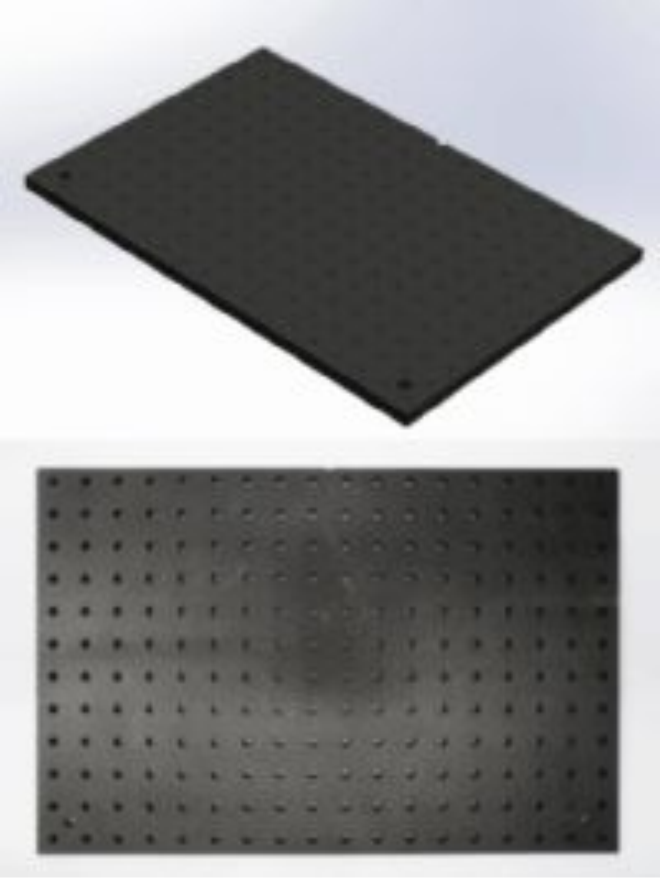
- More than list of parts
- Expected background knowledge to build
- Specific build knowledge
- Physics, electronics, computational skills
- Lots of adjustability (knobs!!)

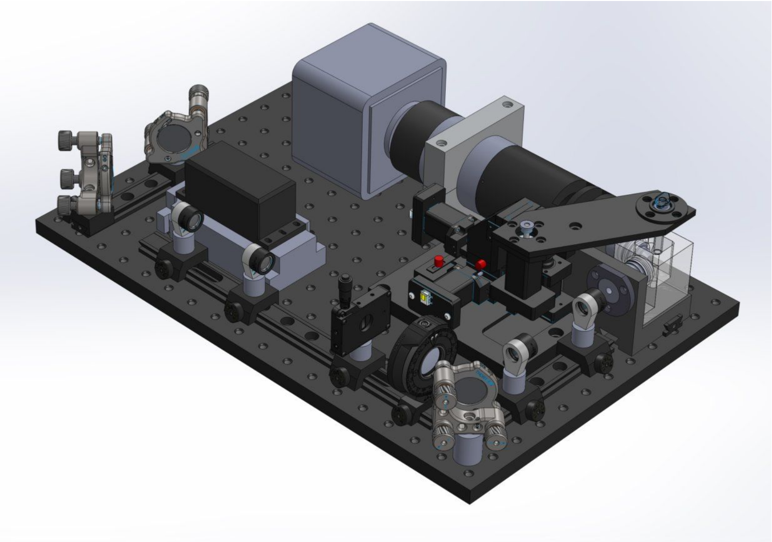
Challenge: what is the “source code” for optical builds?

- No obvious “CAD” format for optical breadboard builds
  - Placement of components is not enough for an optical system
- For research instruments, adaptability is crucial
  - Address pre-existing equipment and other constraints
  - Allow customization and improvement
- “Source code” should be integral to (and facilitate) design and development,
  not an afterthought
- Modification of a part should not require checking everything
  (learn from software engineering principles)
  - This also facilitates making and accepting external contributions to development
- Sharing and *re*-sharing (of modifications) should be easy



Case Study: OpenSPIM





- Open source light sheet microscopy
  Exact location of every part specified
- Detailed, step-by-step assembly instructions (and videos)
- Over 50 labs adopted
- Very manual, rarely updated and few outside contributions

Spectrum

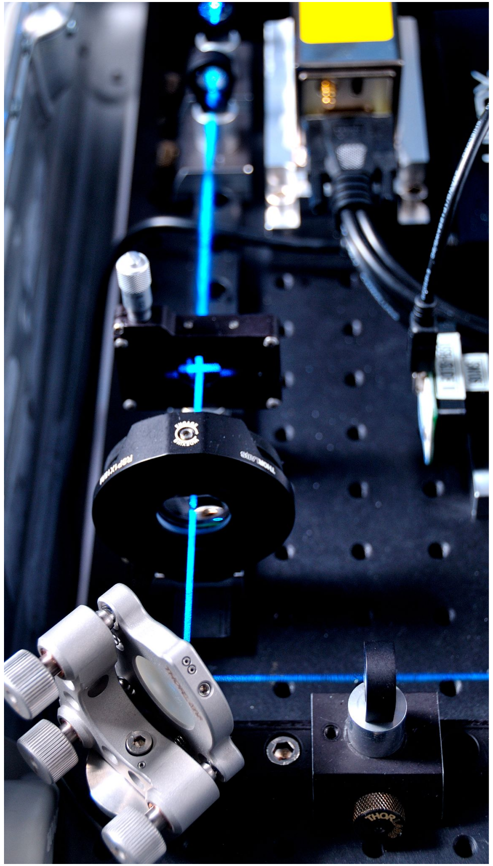
- Stereotyped build
- Step-by-step instructions
- Rigidly specified bill of materials
- Low requirement for expertise
- **Suitable for education, training**
- Sharing modifications is labor-intensive
- Less modular (except in designed-in ways)

- Flexible/adaptable build
- Separation of assembly/alignment techniques from system layout
- Allows substitution of generic parts
- **Suitable for cutting-edge R&D**
- Some expertise required (but identified)

Challenges in documenting complex assemblies

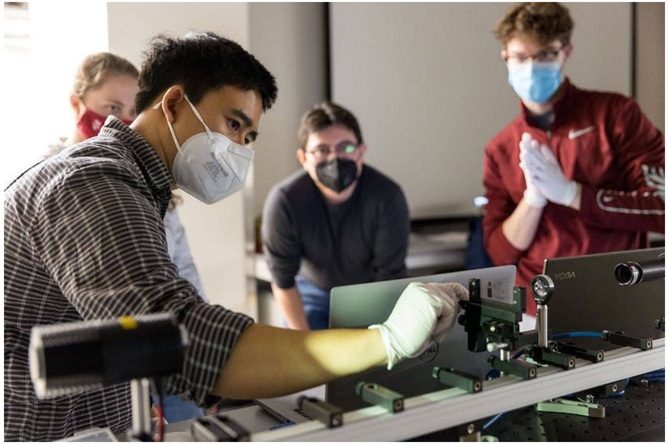
1. How to effectively document the assembly process without resorting to brittle and labor-intensive step-by-step instructions (for optical breadboard builds)

2. How to visually present builds consisting of many sub-assemblies while allowing collaborative and open development (for mechanical assemblies in general)



How to describe/document builds in an adaptable manner

- Identify techniques (procedures, subroutines, recipes) for assembly
    - E.g., laser alignment procedures
- Clarify dependencies between system components and their assembly
- Document system components and layout separately from techniques
    - Gives uncluttered description while allowing full details
    - Reduces work when making changes to a single component
    - Helps clarify required expertise
- Share in an easy-to-edit format (Jupyter Book, GitHub Pages)



Although this is for “optics” builds, the principles may apply to other types of hardware

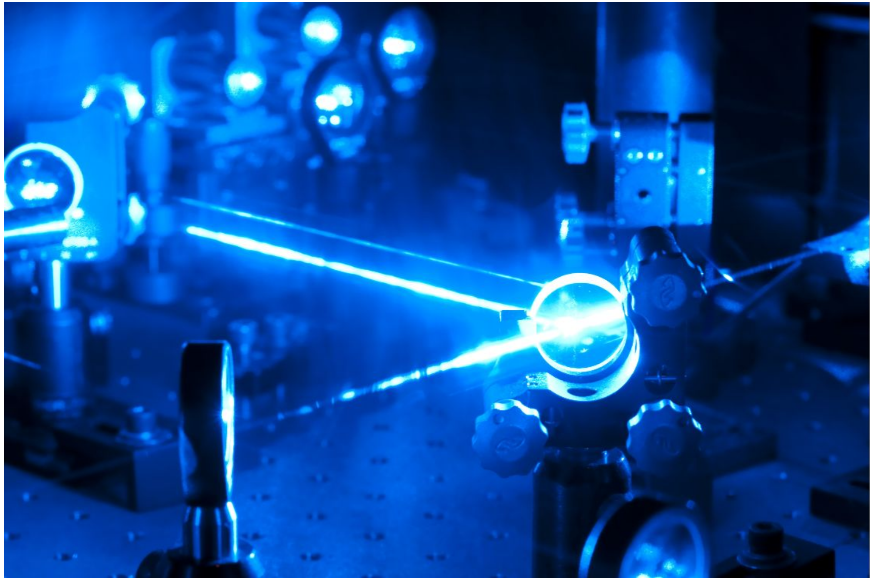
Perspective study (paper): A collaboration between our lab and OSHWA mentor Chris Chronopoulos (Allen Institute)

- Many scientific open hardware projects require complex assemblies of modular components
  - Components may themselves be open hardware, or commercial off-the-shelf

- Methods for collaborative development exist (e.g., GitHub)
  - Works for relatively simple hardware designs, but does not play well with binary CAD files
- Methods to display and manipulate modular assemblies of components exist (e.g., some commercial CAD programs)
- The community needs something with both properties (a web platform, interchange format, and/or other mechanisms)

Goals and next steps:

- Finalize OpenScan software release
- Publish OpenScan software paper with companion website
- Implement OpenScan in outside test labs (already identified, one in process)
- Publish modular assembly perspective paper
- Wrap up student training (OpenScan hardware documentation)
- Finish prototype documentation of OpenScan hardware and build process



Thank You!!

- Open-Source Hardware Association
- Sloan Foundation
- Alicia Gibb
- Lecia Ductan
- Mentors
  - Jinger Zeng
  - Chris Chronopoulos
- Mark Tsuchida, Jenu Chacko, Sruthy Dinesh and whole lab