nature protocols

https://doi.org/10.1038/s41596-023-00855-2

Protocol

Check for updates

An inexpensive semi-automated sample processing pipeline for cell-free RNA extraction

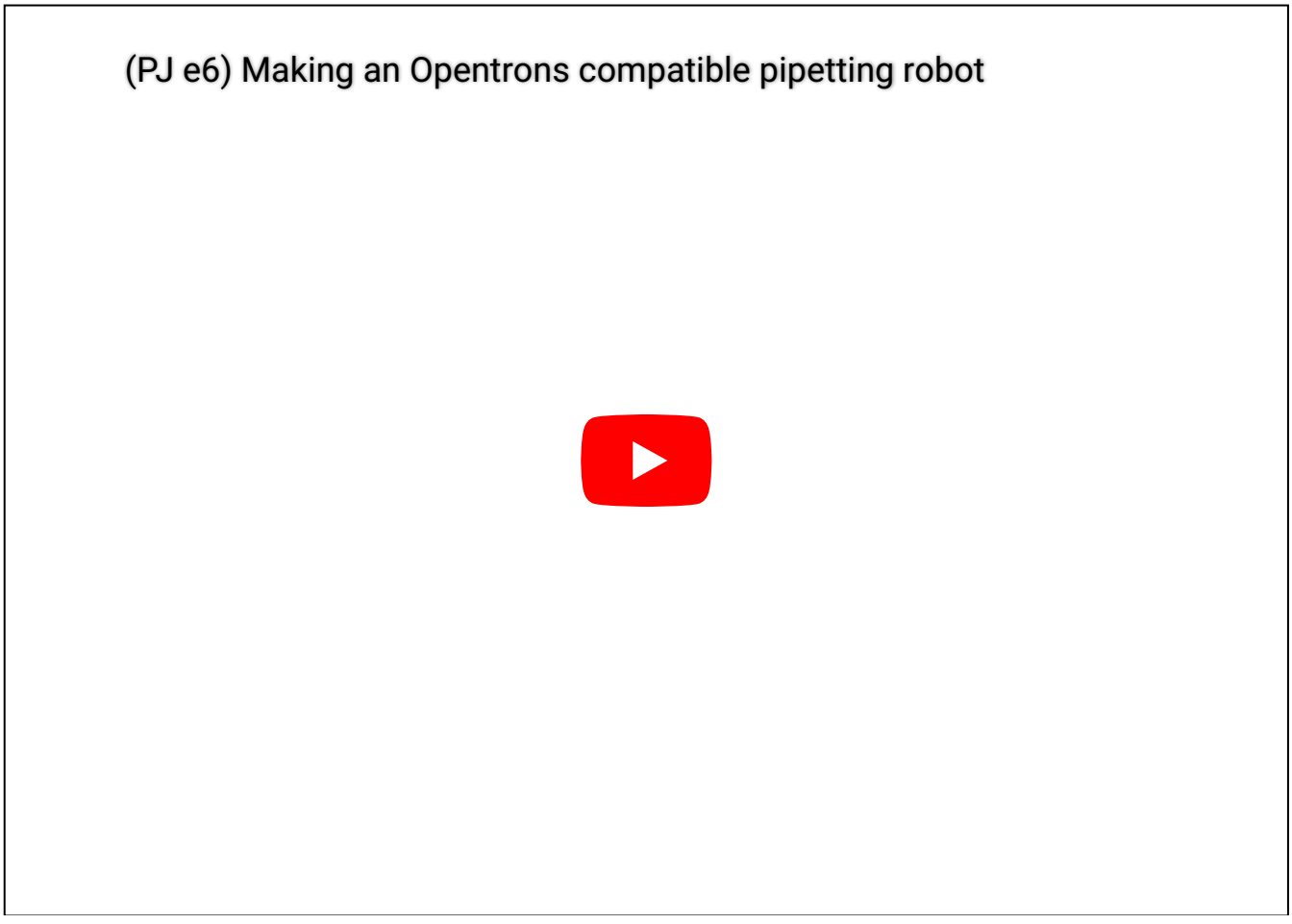
In the format provided by the authors and unedited

Pipette Jockey

I make, optimize and reverse engineer methods and reagents for the molecular biology lab. I also make other cool stuff like lab equipment. What follows are my adventures.

Making a Opentrons compatible liquid handling robot

Published: January 3rd, 2018 Last Modified: October 12th, 2018



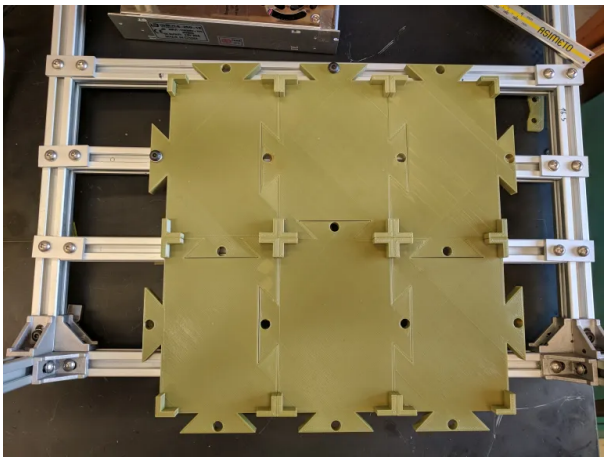
It's finally done, my very own liquid handling robot! The video goes over my trails and tribulations building this thing. Overall I'm glad I did it, the level of complexity forced me to challenge myself at every stage. This is definitely a step above assembling a ready to go 3D printer kit from ebay.

The next video will be doing some actual science with this thing...I think that was the reason to build it, right???

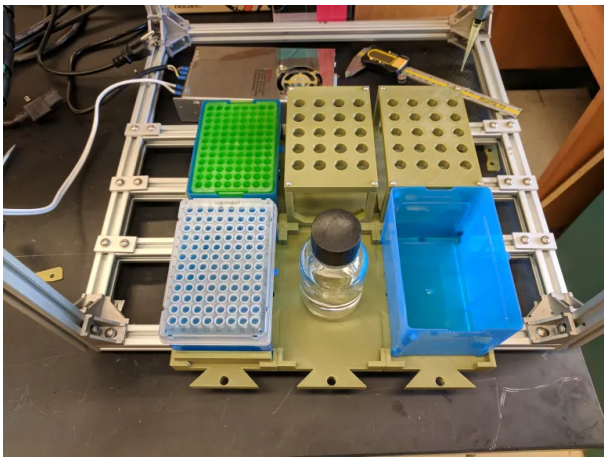
Thanks again to the Opentrons team for releasing their designs for us to play with 🙂

Additional improvements and pictures will go in the blog post as I make them.

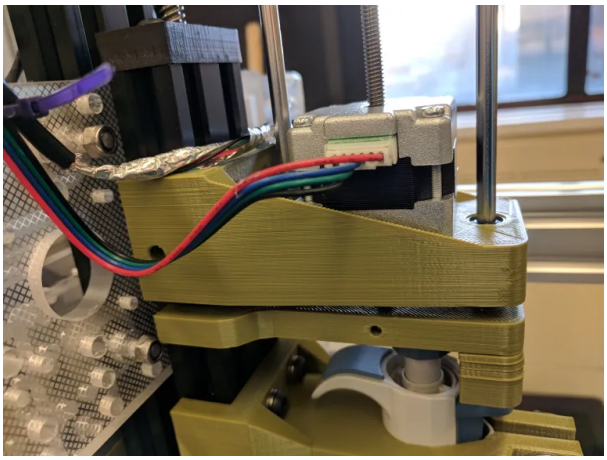
Finally finished the 3×2 deck since uploading the video, seems to be sturdy enough for the job and easy to mount solidly to the robot. One thing that could get your resident Biosafety officer excited is that 3D printed parts are hard to sterilize and clean, like wood. The layers create a rough surface that traps dirt, so I doubt this is suitable for sensitive cell culture work. Ideally it would be polished metal or smooth, chemical resistant plastic so you could douse it in decontamination mix.

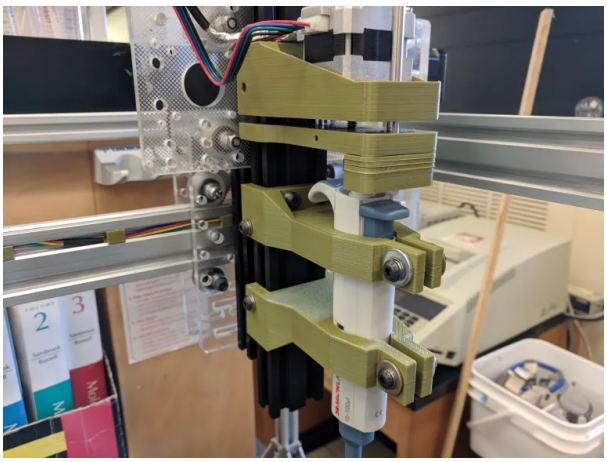


Here it is loaded up with different containers, I 3D printed the 4×5 1.5 mL tube holders in two parts and finally made some nice wedges to keep odd shape stuff snugly in each slot. Pretty reasonable amount of working room I think, we will see if I run into any trouble on that front.



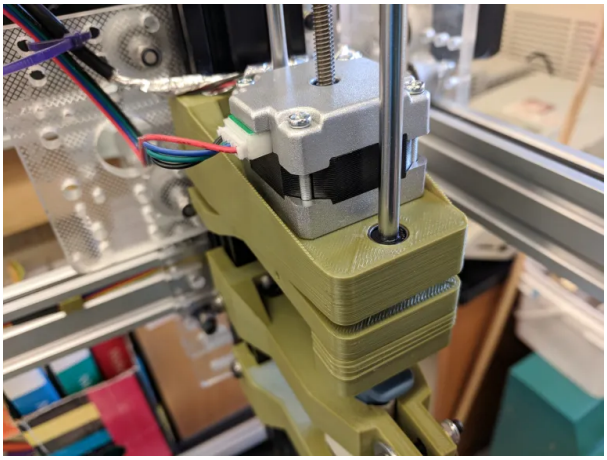
I've had a bit of issue with certain pipette tips being difficult to eject, I solved this issue by beefing up the pipette and motor mounts on the A axis. As well, I increased the length of the wedge that actuates the pipette eject.





Unfortunately the 5 mm linear bearings that support the 5 mm rods may be inferior to the bronze bushings in the Opentrons robot. The linear bearings fit the rod tightly, however a moderate amount of force is enough to allow the rods to bend out

of alignment a bit, which robs the motor of force needed to eject stubborn pipette tips.

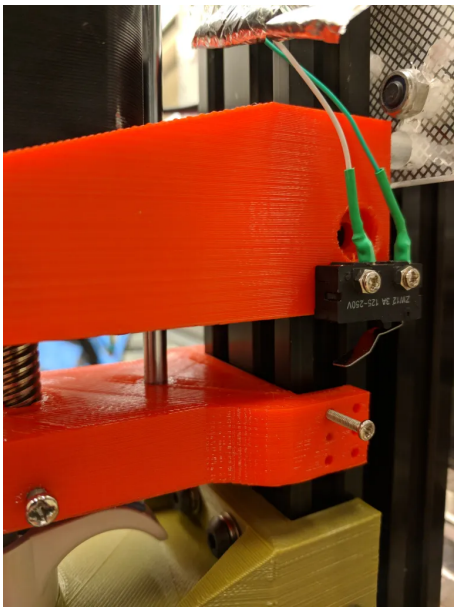


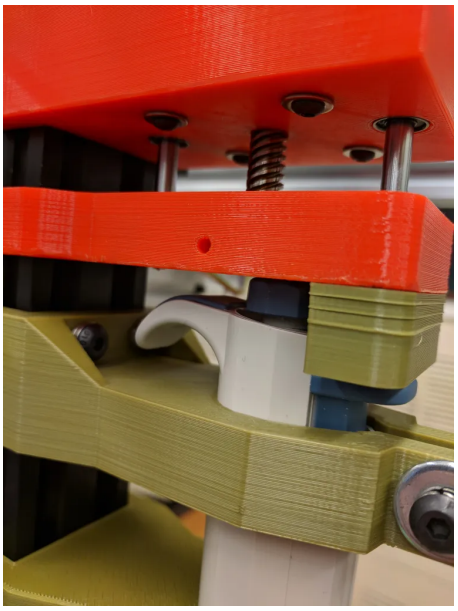
Update 1:

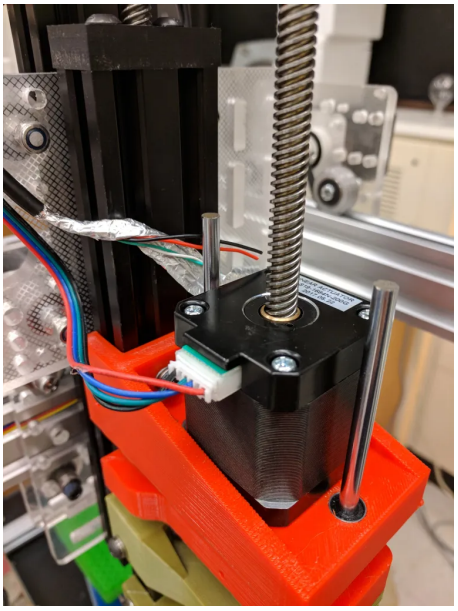
Been benchmarking this to see what kind of accuracy I can get. On a 10-100 uL Dragonlab pipette I can get >1% to 0.5% when pipetting the full 100 uL, not a problem. However, as soon as I try to pipette 50 uL, I'm getting more like 3-4% error. 10 uL? 10-20%. Absolutely useless for preparing qPCRs or even a garden variety BSA gradient.

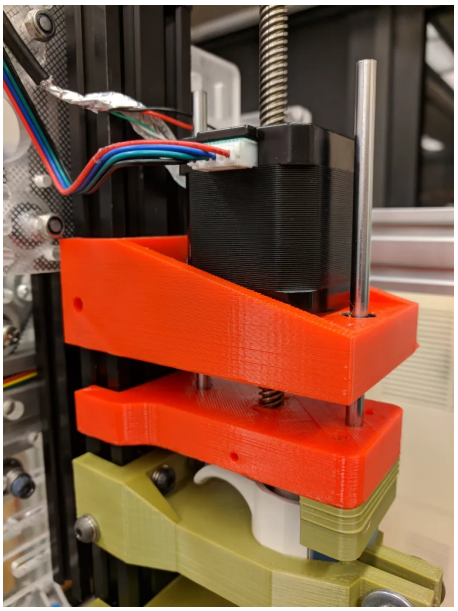
Why so poor? Well, when I pipette by hand, I can get the dragonlab within quoted specs. As soon as it goes into the machine, no good. So, something in the pipetting mechanism is introducing significant error. One clue is that the accuracy is fine at 100 uL, but drops the lower you go. From what I can tell, the lead screw of the non-captive stepper motor was never designed for anything approaching precise work, it's a 5 mm threaded rod. Basically, the further down it plunges, it does a little shaking dance. Looking at the opentrons robot, they use a non-captive stepper with an acme threaded rod, much more precise. As soon as I get my hands on one I'll try again.

Update 2:









Received the non-captive steppers with Tr8-8 acme threaded rods, printed out new motor mounts out of PETG, should be better suited for the job as the motor heats up. Made sure that the threaded rod is supported within the mount. Kept 5mm guide rods, could potentially go for 8mm.

Performance significantly better, plunger platform no longer dances when it actuates all the way down. I can now get easily within, or better than quoted specs of Dragonlab pipettes. Dispensing 20 uL with the 10-100uL pipette I was getting a standard error of (+/-) ~1.5%, not too shabby for that pipette. As well, keep in mind that the lower in volume you go, the hairier it gets in actually benchmarking a pipette. The old Mettler Toledo AB104S we have is quite good, but it's possible that the scale is introducing significant error into the measurements. So it's possible the robot is more accurate then I can test with the scale I have.

Also, in benchmarking the opentrons clone, I'm starting to get a feel for the quirks of the system and its limitations. Basically, the opentrons pipetting robots reminds me a lot of the difference between hobbyist 3D printers and professional 3D printers from say, Stratasys. A 3D printer kit off gearbest is cheap, but you can pretty much expect that you will be tinkering with it from day 1. On top of that, there will be reliability issues which you yourself will have to fix. Trying a new filament? Expect a learning curve. Stratasys printers are too expensive for most individuals, but they're infinitely more reliable, and since you will be buying filament material straight from

the company, there will be no tinkering for the average stratasys client, it's all taken care of in the price.

Here, opentrons would be the hobbyist equivalent of machines like the Beckman Biomek. How much is a new Biomek? Hard to find a specific quote, but I'm gonna go real wide, somewhere between 30,000$-200,000$ easily, not to mention proprietary tips.

An opentrons robot runs for 3-4000$ish, ~1000$ for a DIY build. Cheaper, but guess what? There's tinkering and reliability issues. Had to change tip brands? Centrifuge tubes? Don't expect your old calibration or program to work without adjustments. Don't you even *think* of running a freshly written program on your precious samples and expect everything to go smoothly the first time. Developing even a single protocol that runs reliably every time can take anywhere from 30 minutes to hours! Not sure how much easier protocol development would be with the Biomek software, but there's no mucking around in python for one. Also, the reliability and repeatability would likely be significantly better with consistent tips and plates straight from Beckman. Anyway, I'm close to actually running an experiment with this beastie, just wanted everyone to know the realities of this class of pipetting robots.

Rough cost estimation

(Had most of it all on hand, this is what I remember paying for parts. Some I manufactured, some I scavenged)

12 x 420mm long 2020 aluminum extrusion —– 60-100$ CAD
3 x 400-500 mm long 2040 v-slot extrusion —– 50-75$ CAD
8-10 mm thick polycarbonate for plates —– Free, from someone's old hood or something? maybe 75-100$ CAD new?
20 x 2020 aluminum corner pieces —– 1.50 – 3$ a piece locally, buy in bulk off ebay for 0.50 a piece, ~15$ CAD
4 x Nema 17 steppers —– 80-100$ CAD, had these from scrapped 3D printers
1 x Nema 17 non-captive stepper —– Uncommon, 30-50$ CAD each
16 x Polycarb V-wheel kit —– 6-10$ CAD a piece locally, worth it to buy from overseas in bulk for ~2$ a piece, ~40$ CAD total

2 Kgs of 1.75 mm PLA for 3D printed parts —– 50$ CAD
3 Dragonlab pipettes —– 40$ CAD a piece, but customs destroyed me, brought it up to 80$ per pipette. Avoid Fedex I guess? 240$ CAD
2 x 5 mm linear bearings —– 10$ CAD
A length of 5 mm rod —– 15$ CAD
MKS sbase 1.3 board —– 70$ CAD
Wiring (Wires, connectors, etc) —– 80$ CAD
Fasteners (Nuts, bolts, washers, etc) —– Mostly M3 and M5 size stuff, as well as 2020 extrusion T-nuts. 50-100$ CAD, buy in bulk and save for other projects

Time —– Nuff'. 3ish months when I was free. How much is an hour of your free time worth?

Running total: 500-600ish CAD with stuff I had on hand, upwards of 1000$ if I had to buy everything new.

Consider that the Opentrons uses a legit smoothieboard (250-300$ CAD), 3030 extrusion for the frame and aluminum plates.

9 thoughts on "Making a Opentrons compatible liquid handling robot"

DerAndere

October 4, 2018 at 9:23 pm

Hi,
You might be interested in the program GGCGen (https://gitlab.com/RobotsByDerAndere/CartesianRobotsByDerAndere/GGCGen/). It is a Graphical G-code Generator so it is maybe more flexible (but less mature) than

the “Protocoll Designer” of the OT App in conjugation with the “Opentrons API Server”. GGCGen features a graphical user interface for creation of G-code scripts for lab robots. The G-code could be directly interpreted by a G-code interpreter. Everyone is invited to fork my project and modify it to include the Opentrons layout. So far it was only used for my own DIY lab robot, PipetBot-A8 (https://it-by-derandere.blogspot.com/p/pipetbot-a8.html) that runs Marlin firmware.

Regards, DerAndere

Pipette Jockey

October 4, 2018 at 9:34 pm

Incredible work! Thank you for sharing! I will definitely check out your method when I eventually revisit the pipetting robot project. I kind of burnt myself out a little to be honest on that one!

What kind of pipetting module are you using with your A8 robot?

Also, do you mind if I link to your blog on the pipetting robot posts? I think it will be useful for readers 🙂

DerAndere

October 7, 2018 at 3:07 pm

You are welcome to add links to my blog in your posts. Don’t be too enthusiatic though: If you look at the GGCGen source code, you will see it is not designed well and error checking is missing. It is my first software ever, after all. I will try to refactor the code, to at least replace some hardcoded xy-offset-constants by variables. Then it would be a bit easier to adjust the code for another working space layout. The Opentrons App is certainly designed 1000x better. Did the OT team do the all the work of adapting their software and the user interface for your robot? Or do you simply ignore grid positions past the 2×3 positions

available with your robot layout? For me the OT source code was too complicated to start adating it.
Hardware-wise I am equaly “lacking behind”: I am simply planning to use a floor-mounted DIY 1ml syringe pump (https://www.pinterest.de/pin/348888302370315553/), connected via tubings, because I do not want to spend > 180 € total. If I get an old hamilton syringe, I may design for more precision. Tip removal is done like in another project (http://www.robotc.net/blog/2010/05/01/cool-project-liquid-handling-robot/)
Keep up the great work on your homepage!

DerAndere
PS: +1 on how time-intensive these projects are 😀

Pipette Jockey

October 9, 2018 at 4:16 pm

Sent you an email, I could help you with the syringe situation 🙂

One of the OT team gave me some broad strokes instructions on how to set up the software, but other then that I was on my own, I felt kind of weird taking up their time asking about how to get around paying 4000$ for their robot, you know? Yes, you basically don’t really have to worry about where a container is, as long as it’s in the workable volume of the robot since you calibrate the position of each container at a specific X, Y coordinate. You do this every time you run a program, or bad things happen! Now, with a strong programming background you could modify the firmware and tailor it to your liking, however as a neophyte like myself I found it pretty difficult to set up linux quite the way its supposed to, and everything just collapsed with a pile of missing dependencies…I was pretty done after that XD

DerAndere

October 11, 2018 at 11:53 pm

Much appreciated!

You deserved to get help because you were brave enough to ask. Plus the open-mindedness of their team is part of the commercial Opentrons value: If potential customers read about your Aliqbot but then decide that they are not willed to fiddle around with a selfmade robot, they might prefer an original Opentrons robot over some other brand because they know that they can then profit from both, customizable open source software and solid official support.
Anyways: I will have some more fun fiddling around with DIY robots, now! 🙂

DerAndere

Gernot

March 6, 2019 at 8:49 am

Klasse – will have a look at it

DerAndere

September 18, 2019 at 10:30 am

My graphical protocol designer GGCGen (see above) now outputs Python protocols that are partially compatible with opentrons APIv2 . I had to work around the tight integration of Smoothie in Opentrons software. So I made a seperate backend that translates Python protocols written for opentrons robots to G-code and can control robots that run Smoothie or Marlin firmware. All work in progress and highly experimental. Hope this gives curious people more options. Still, X000$ for an original Opentrons OT robot is the way to go. Best regards!

Coris

June 3, 2020 at 7:36 am

Hi.
You are doing a great job.
I am working in the development of a project like yours, inspired from opentrons also, but based on Derandere program.
Anyway i am stuck in to implementing the software.
Are you able to share the way how you manage to implement opentrons into your project?
Thx.

venu

November 3, 2020 at 4:55 pm

hi sir good job sir. please send me the arduino codes for liquid handling machine...