Article title

Sidekick: A Low-Cost Open-Source 3D Printed Liquid Dispensing Robot

Authors

*Rodolfo Keesey, Robert LeSuer, Joshua Schrier**

Affiliations

Department of Chemistry, Fordham University, 441 E. Fordham Road, The Bronx, New York 10458, USA
Department of Chemistry and Biochemistry, SUNY Brockport, 350 New Campus Drive, Brockport, NY, 14420…address…

Corresponding author's email address and Twitter handle

jschrier@fordham.edu
@JoshuaSchrier

Abstract

The Sidekick is a desktop liquid dispensing robot, compatible with standard SBS microplates designed for accessible laboratory automation. It features an armature-based motion system and a fully 3D printed chassis to reduce overall mechanical complexity and accommodate user modification. Liquid dispensing is achieved with four solenoid driven positive displacement pumps that deliver 10 microliter increments. Control is provided over a USB serial interface using a simple vocabulary of commands, implemented in the form of a short Micro Python program. At a total price of $650, the Sidekick offers laboratories an easy to build, easily maintained, open-source liquid handling system for both research and pedagogical introductions to lab automation.

Keywords

Laboratory Automation, 3D Printing, Chemical Experimentation

Specifications table

Please replace the italicized instructions in the right column of the table with the relevant information about your hardware.

Hardware name	*The Sidekick liquid handler*
Subject area	• Chemistry and biochemistry

Hardware type	• Chemical sample handling and preparation
Closest commercial analog	Mantis Liquid Handler
Open source license	*All designs must be submitted under an open source license (for more details see the Guide for Authors). Please specify the open source license you've selected here.*
Cost of hardware	*$650*
Source file repository	*https://github.com/rodolfokeesey/Liquid-Handler*
OSHWA certification UID *(OPTIONAL)*	*We encourage you to apply for a free OSHWA Certification, which confirms your work is open-source compliant.* *If certification has been acquired, insert the OSHWA UID here. For example: "CH000005". In your OSHWA certification project description, include a link to your HardwareX publication and the tag #HX.* *If you haven't acquired certification, please delete this row of the specifications table.*

1. Hardware in context

Automation increases productivity and decreases human error. Chemical laboratory automation has a long history, dating back from hydraulic control devices in 1875 to mechanical, electromechanical, and robotic systems [1]. More recently, the idea of autonomous experimentation systems that combine automation with artificial intelligence has become an active area of research in chemistry and materials science [2], [3]. However, as devices became more complicated, it was no longer possible for benchtop scientists to build their own devices, and instrumentation became the realm of the equipment manufacturer [4]. Increased laboratory productivity and precision justified high-cost automation, leading to manufacturers focusing on more expensive but capable devices [5]. However, this can place such equipment out of reach for less resourced laboratories, particularly in basic research and educational settings, preventing the automation of routine tasks. Liquid handling is one such foundational task in chemical experimentation, and commercial liquid handlers often cost tens of thousands of dollars. Opentrons offers one of the least expensive commercial solutions for automation in the form of the Opentrons OT-2, but even this machine starts at $5000 [6]. This is where open-source hardware can bridge the financial gap to commercial grade equipment, allowing smaller labs to experiment with automation, or acquire hardware for pedagogical purposes.

A variety of open-source liquid handling machines seek to lower the upfront cost of automation [7]–[9]. The OTTO is a Cartesian liquid handling machine, that uses linear rail guided motion, and liquid aspiration/dispensing via micropipette. This design is entirely open source and assembled with off the shelf components for an estimated USD$1500 [7]. OTTO addresses the reliability concerns of DIY devices by adding additional sensors that detect possible alignment issues in sample preparation [4]. The FINDUS robot uses a similar Cartesian movement system guided by a less expensive linear bearing system. By combining this bearing system with 3D printed parts, the FINDUS reduces the reported build cost to less than USD $400 [8]. PHIL is a microscope stage-top liquid handling robot, designed for microfluidics and biological research. Unlike OTTO and FINDUS, PHIL uses a 3D printed robotic arm, and open-source peristaltic pumps [9].

While these designs have impressive capabilities at a low price, they often require tens of hours of build time, the sourcing of dozens of parts, and the construction and maintenance of the liquid dispensing systems [7], [8]. This is an obstacle to potential users, and both reduces adoption while increasing the true cost of the system. Additionally, both the OTTO and the FINDUS rely on a micropipette for liquid dispensing. The micropipette adds an extra hidden cost to the final project; a P200 micropipette from a reputable manufacturer ranges in price from $170 to $360 [10], [11], and limits the accurate volumetric range of the handler to the range of the micropipette. While PHIL sidesteps these issues with multichannel peristaltic pumps, the time to assemble and calibrate multiple microfluidic pumps may be intimidating to the end user. To address these issues, we developed the Sidekick, a fully 3D printed liquid dispensing robot designed for use in research and education. The Sidekick increases the approachability of open-source liquid handling automation by replacing the linear Cartesian movement system with a simpler armature-based system. It delivers liquids with four solenoid-driven displacement pumps, removing the need for assembly. These parts are combined with a fully 3D-printed chassis and a MicroPython control system. The Sidekick costs $650 to build and offers a balance between ease of use and affordability.

2. Hardware description

2.1 Hardware Overview

The Sidekick's design prioritizes simplicity of hardware, software, and assembly. Most open-source liquid handlers use a Cartesian gantry motion systems like those used for 3D printers [7], [8], [12], [13]. These designs are well suited for many automated liquid handling tasks, but their build complexity may be intimidating for labs looking to experiment with automation. These designs often use commercial micropipettes for liquid handling, adding additional maintenance and cost that is often not included in the bill of materials estimates [7], [8].Their Cartesian motion systems run G-Code (RS-274), a widely used programming language for automated machine tools. This requires the user to translate commands to G-Code, putting another layer of abstraction between instruction and movement [7].

The Sidekick addresses these issues by replacing the Cartesian kinematic system with an armature-based design, which lowers part number and increases the number of 3D-printable components. This kind of armature system was inspired by drawing robots and greatly decreases the number of components required for planar motion. The reduced mechanical complexity in turn reduces build time, machine size, and materials cost. The motion control system of the Sidekick costs less than $130, requires the end user to source 18 parts, and can be assembled in under two hours (excluding the time to 3D-print the parts).

To keep the liquid dispensing subsystem simple, the Sidekick uses four commercial micropumps that dispense digital increments of 10 uL upon a voltage pulse cycle. Using four pumps allows the Sidekick to dispense four different liquids. The primary advantage of commercial micropumps is that they require little calibration, greatly simplifying the initial construction and ongoing maintenance. Should the user damage the liquid handling system (e.g., by using corrosive materials), they are modular and can be replaced. These pumps are included in the overall price of the Sidekick, eliminating any hidden costs associated with adapting laboratory micropipettes. However, if flowrate or accuracy does not meet the user's needs, the Sidekick can accommodate other multiple open-source options, such as the Ender 3 adapted syringe pump [14], the Poseidon syringe pump [15], or the FAST peristaltic pump [16].

The Sidekick is controlled by a Raspberry Pi Pico and is programmed in MicroPython, a subbranch of Python designed for microcontrollers [17]. Kinematic calculations are calculated on the Pico and directly translated into armature angular position, eliminating the need for intermediary software to send or interpret G-Code. The Sidekick takes plain text commands via serial and can receive instruction from any programming language that has access to the serial port.

The chassis of the Sidekick is entirely 3D printed in PETG, reducing material costs, part sourcing time and allowing users to customize part dimensions to incorporate imaging, spectroscopy, or other functionality. Altogether, these design choices make the Sidekick a liquid handling robot with:

- A low mechanical complexity and part number
- An assembly time of under two hours
- A multichannel dispense system.
- A cost of under $650

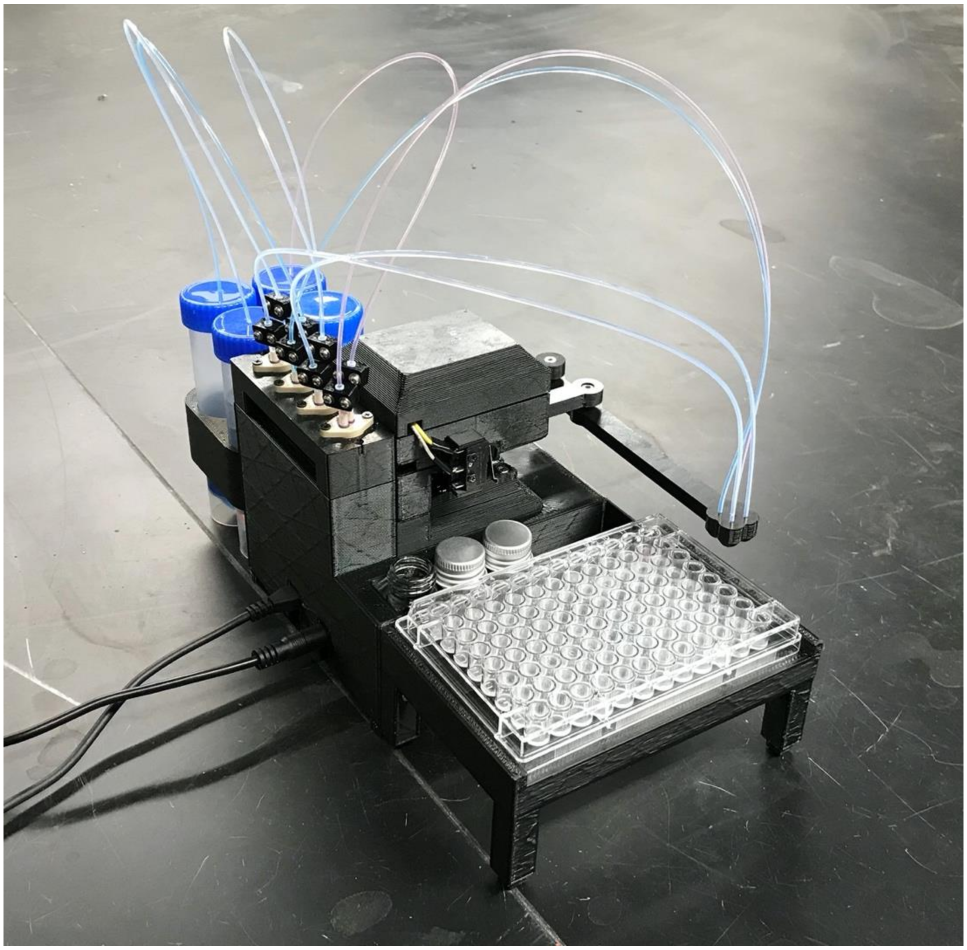


Figure 1: The Sidekick Liquid Handler

2.2: Dispense Armature

Armature Design: The Sidekick uses a selective compliance articulated robot arm (SCARA) inspired by designs like Lineus [18], Tracey.io [19], and the Mantis Liquid Dispenser [20]. These types of robots are mechanical simple and reduce cost and build time [21]. In contrast, most consumer grade 3D printers and open-source liquid handlers share the architecture of a 3-axis Cartesian robot [7], [8], [12], [13]. The Cartesian design maximizes stability and precision, and the orthogonal axis make kinematics calculations trivial. While the popularity of 3D printing hardware has lowered the cost to create these types of robots, they still have significant upfront cost and build time. Potential builders must source linear bearings, pulleys, and motors to accomplish linear motion. If a user is only dispensing onto a single plate, the creation of such a large machine is unnecessary. In contrast, the Sidekick's SCARA design requires four 3D printed legs, three ball bearings, and two stepper motors. Positional accuracy is achieved by implementing a homing system, high resolution stepper motors, and a software calibration step to ensure accurate plating.

This armature-based design is well suited to the dispensing into a single 96-well SBS microplate, and the increased complexity of motion programming inherent to SCARA designs are hidden to the end user. The inverse and direct kinematics functions are preprogrammed, and the experimentalist can simply direct the end effector to a desired position. The smaller operating area reduces the amplification of positional error that would occur with a longer armature. The lack of any z-axis motion like aspiration or tip ejection alleviates the need for strict end effector rigidity.

Figure 2 shows the usable area of the dispense arm, and Figure 3. shows a schematic of the dispense arm, which consists of four 3D printed legs, connected at each joint by 10mm ball bearings. The base of the armature is connected to two stepper motors. These motors are responsible for moving the dispense head across the useable area. Such a design allows for any reachable location to be described by the angular position of L1 and L2, without having to convert from rotational to linear motion.

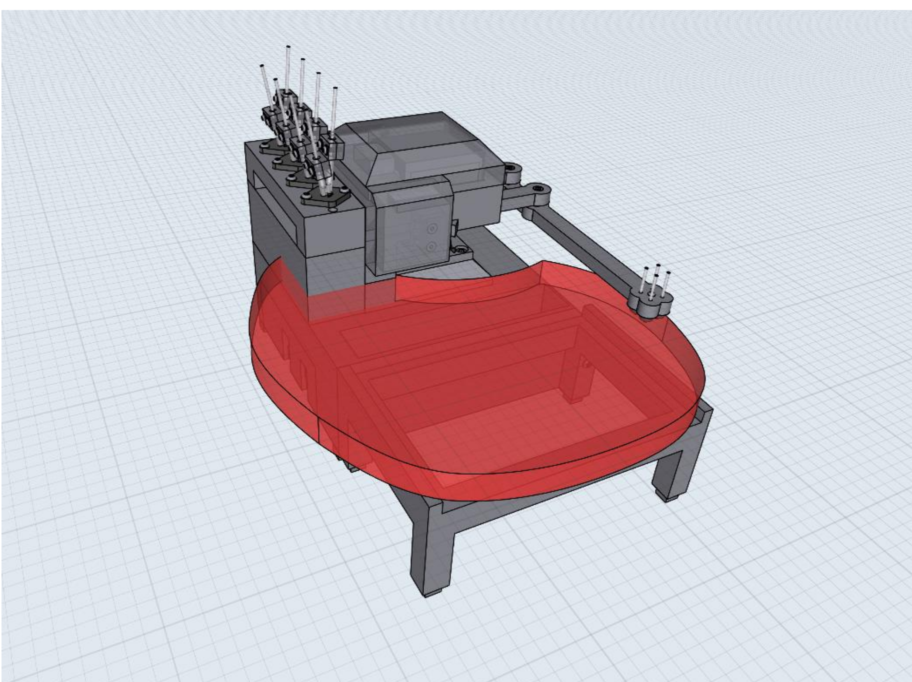


Figure 2: A visual representation of the Sidekick's usable area.

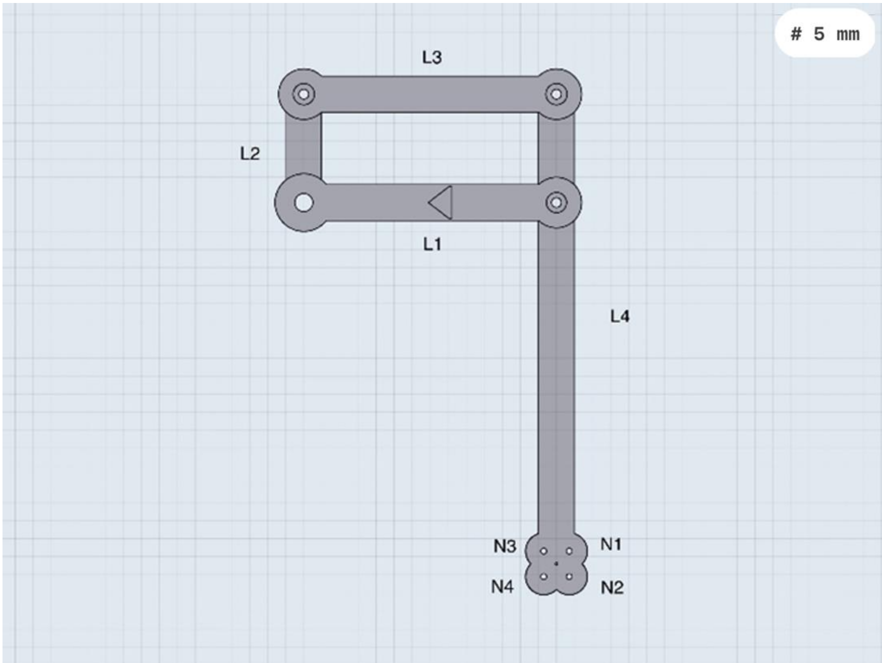


Figure 3: A Diagram of the Sidekick Dispense armature. Liquids can be dispensed from N1, N2, N3 or N4.

Motor Choice: The Sidekick motors must be both precise and accurate, as differences in angular position between wells can be as low as 3.2 degrees. We initially considered servo motors, as they simplify control and do not require any additional drivers or encoders, but we found hobbyist servo motors did not provide the requisite resolution. Backlash in the gearings, and an imprecision in the encoders of +-1.5 degrees precluded their use. We tested the MG90S and the SER0047 micro servos and found that their gearing backlash was greater than 1 degree. When attempting to position the armature over the 96 well plate, the end effector would vary from the center of each well by more than 3mm. The backlash and low angular resolution made it impossible to account for this inaccuracy by any computational method. We then tested the larger Feetech 3443, TowerPro SG5010, and the DF Robotics SER0044 servo motors. While these larger servos did not have any measurable backlash, their positional feedback suffered from noticeable non-linearity (Fig. 4). While this non-linearity could be accounted for by applying a regression and correcting the positional command, this would have to be calibrated by the end user, which seemed impractical. Larger servos with the required angular resolution, like the Dynamixel XL-320 are prohibitively expensive for this design and require a proprietary driver [22].

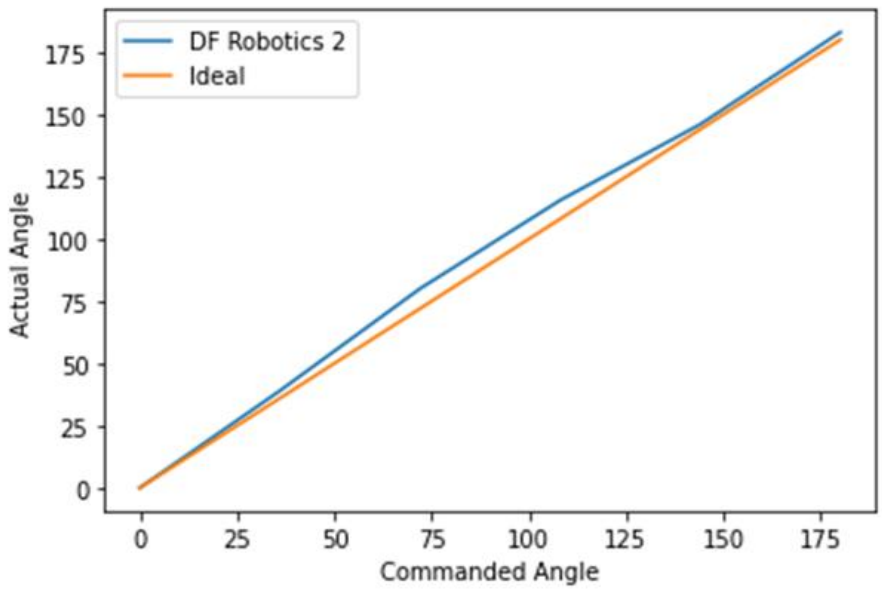


Figure 4: DF Robotics SER0044 Experimental vs. Commanded Angle, Nonlinearity

Despite the need for an additional driver and the loss of closed loop control, this led us to consider NEMA 17 Pancake stepper motors, as they offer precise motion and a small footprint. Choosing a stepper motor with a step-angle of 0.9-degrees offers a baseline angular resolution that meets the Sidekick's requirements. With the addition of micro-stepping, these steppers offer an even greater angular resolution, opening the possibility of using 1.8-degree stepper motors to further save on cost. In testing, these 0.9-degree steppers with 1/8 micro stepping reliably centered the end effector over each well with 0.1 mm precision. The loss of closed loop control, and the possibility of skipping steps are not concerns due to the low resistance to armature motion. The pancake stepper motors run on 12-24 volts, and are rated at 1.2A per phase, with a holding torque of 11 N/cm. The Sidekick drives these steppers at 12 volts at 1A per phase.

2.3: Motor Control

Driver Choice: Stepper drivers were chosen based upon footprint, power requirements, and operational noise. Several drivers were tested:

Adafruit TB6612: The Adafruit stepper/motor driver is a good hobbyist choice for board development. The driver is quiet in operation but lacks an adjustable V-Ref. Motor overheating was observed in 30-minute continuous operation, manifesting in skipped steps and high motor surface temperature.

Allegro A4988: The A4988 is a dedicated stepper driver, offering adjustable V-ref and higher micro stepping capabilities. While the adjustable max current solves the overheating problems, the A4988 suffers from high operational noise. The driver whine during movement and holding torque is very audible. While driver noise can be mitigated by reducing current and increasing micro stepping, neither option was able to reduce the noise to an acceptable level [23]. While older users may not experience discomfort, younger operators reported irritation.

TI DRV8825: The DRV8825 is also an acceptable option with smooth reliable motion and good thermals. However, it also suffers from high operational noise that cannot be mitigated by micro-stepping or reducing current. While it may be possible to increase micro stepping past the 1/64th steps that were tested, the Pico's processor becomes the limiting factor and skips steps. This can be solved with programmable IO pins, but with an increase in programming complexity.

TMC 2209: The TMC 2209 drivers were the best choice in thermals, noise, and accuracy. These drivers lack any noticeable noise in operation, offer high micro stepping capabilities, and adjustable V-ref. Additionally, these drivers feature stall protection, which allows for the possibility of adding feedback to increase reliability and remove limit switches during the homing process. These were the final choice for the Sidekick's drivers. These drivers are used in conjunction with two limit switches that mark the home position of the dispense armature to drive precise and accurate motion.

Kinematics: The Sidekick's kinematics feature a preset lookup table that converts between 96-well plate locations and stepper motor angular position. To accommodate for variations in assembly and plate placement the Sidekick can generate plate maps for any N x M array of wells. This is used to either fine tune effector placement in case the preset lookup table is not sufficiently accurate, or to generate a new map for a different configuration of wells. Calibration is achieved by centering the effector over each of the four corner wells. A linear interpolation is then applied to determine the location of the remaining wells.

For fine tuning effector placement and manual movement, the Sidekick can locally calculate kinematics. This allows the use of non-standard plates or more bespoke applications that require precise locational placement. The kinematics are calculated for each of the four nozzles, and the center point of the effector. The kinematics algorithms are all written in MicroPython, and the Sidekick's motion is controlled via a library of commands sent by serial port.

2.4: Pump System

Four main options were considered when choosing pumps. Micropipette adaptations, open-source peristaltic pumps, open-source positive displacement pumps, and commercial positive displacement pumps. These routes offer the best balance between affordability and precision.

The adaptation of handheld micropipettes for liquid dispensing robots is a common design choice [7], [8]. While micropipettes offer excellent precision and accuracy, their adaption for automated use is both expensive and calibration heavy. Additionally, to aspirate, dispense, and change tips requires z-axis mobility. This would add additional cost to the motion system and increase calibration times for the end user.

Peristaltic pumps are ubiquitous in open-source lab automation hardware. Their use can be seen in the PHIL liquid handler [9], computational chemistry applications and their development is often shared in standalone hardware articles [16], [24], [25]. Given their support and performance, they were a strong contender for the multichannel pumping system. These pumps can be purchased, but there are also open-source pump designs, like the FAST, offer flow rates as low as 0.7 µL/min [16], [24]. In theory, this low flow rate would allow for microliter level accuracy during dispensing. Ultimately, the time required to build and calibrate these peristaltic pumps was the deciding factor. As the Sidekick is equipped with four channels, these pumps would have to be built and calibrated multiple times. At a price of $50 per pump, the savings did not justify the increase in complexity and build time [24].

Open-source positive displacement pumps offer wider functionality than pumps. Their ability to aspirate through a disposable pipette tip allows for the intake of liquids and widens functionality beyond dispensing. There are numerous open-source vacuum displacement pump options, but the current open-source designs are rarely tested to volumes under 10 µL [14], [15]. These pumps often utilize glass syringes or plastic coupled with a stepper motor and a lead screw to drive linear motion, so they are less mechanically complex than peristaltic pumps [14], [15], [25], [26].

The positive displacement pumps from the Lee Company were ultimately chosen for their best out of the box performance. The LPM series of pumps are a range of chemically inert, solenoid driven positive displacement pumps. The pumps are energized by a 12-volt square wave, and aspirate fluid from the inlet when energized then dispense when de-energized. The wetted materials of the pumps are a combination of polyether ether ketone (PEEK) and fluoro elastomer (FKM) [27]. The Sidekick is equipped with four of the 10µL variant (LPMA1250110L). As we wanted to focus on accessibility and cost, the use of calibration free commercial displacement pumps offering 10 microliter aliquots with +-15% accuracy struck a balance between performance and ease of use [28]. Given the comparable performance and cost of open-source peristaltic pumps, the LPM pumps offer the requisite accuracy while skipping the development and calibration of assembling an open-source design. The 10-microliter aliquot offers sufficient granularity to perform a wide variety of dispensing tasks. The LPM pumps are also self-priming and do not require additional motions in the z-axis to aspirate and dispense. This complements the armature motion system and mitigates the need for assembly or calibration.

The Sidekick uses 1/32" ID PTFE tubing for liquid pathing from the reservoir and dispensing into the target well. The 1/32" ID tubing has an internal diameter of .79", comparable to a 20-gauge PTFE lined dispense needle [29]. This narrow tubing serves two purposes. The narrower tubing reduces cross sectional area at the dispense tip, decreasing the possibility of dripping during travel motions and is more flexible, reducing the resistance on the armature when bending the tubing during travel.

2.4: Electronic Components

The Sidekick is controlled by a Raspberry Pi Pico running MicroPython. The Pico is mounted on a custom PCB designed in Fritzing [30]. The PCB accommodates the remaining electrical components, and header pins for hardware connection. Each of the two stepper motors are driven by a TMC 2209 stepper driver. The four pumps are driven by a ULN2803A Darlington Array. The remaining unused pins of the Pico are routed out to accessible header pin through holes, so that the end user can add additional hardware. The Pico is powered by the 5V USB bus, and the stepper motors and pumps are powered by a generic 12-volt, 5-amp power supply.

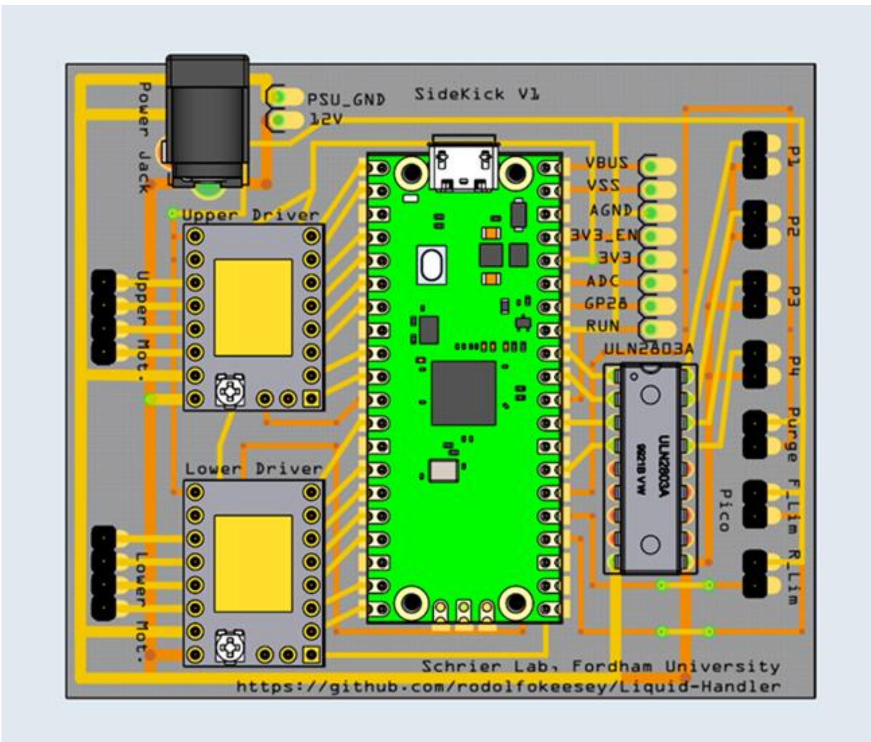


Figure 5: A rendering of the Sidekick's electrical components.

3. Design files summary

Design file name	File type	Open source license	Location of the file
Base	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MainBody
Button Housing	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MainBody
Button Housing Front	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MainBody
Foot	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MainBody
PCB Tray	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MainBody
Leg One	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/Armature
Leg Two	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/Armature
Leg Three	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/Armature
Leg Four	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/Armature
Upper Motor Mount	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MotorMounts
Lower Motor Mount	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MotorMounts
Motor Cap	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MotorMounts
Front Switch Mount	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MotorMounts
Rear Switch Mount	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/MotorMounts
Adapter Clamp Front	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/Pumps s
Adapter Clamp Rear	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/Pumps
Pump Mount	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/Pumps
96 Well Plate Holder	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/96WellTrayHolder_Reservoir
50ml Tube Holder	STL		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/3D_Assets/96WellTrayHolder_Reservoir
SideKickV3	fzz		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/PCB

SideKickV3	zip		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/PCB
Pico Snapshot	folder		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/Pico_Snapshot_10_6_21
Assembly Instructions	pptx		https://github.com/rodolfokeesey/Liquid-Handler/tree/main/Supporting_Documentation

Item Descriptions:

Base: The base of the Sidekick liquid handler. The purge button, motor assembly, pump assembly, and plate tray are attached to the base.

Button Housing: Houses the purge button and attaches to the base of the Sidekick.

Button Housing Front: Covers the front of the purge button and secures it into place.

Foot: Standoff feet that attach to the bottom of the Sidekick's base and plate holder.

PCB Tray: Holds the PCB and connects to the bottom of the Sidekick's base.

Leg One: The first leg of the armature movement system, this leg is attached to the upper stepper motor.

Leg Two: The second leg of the armature movement system, this leg is attached to the lower stepper motor.

Leg Three: The third leg of the armature movement system, this leg is attached to legs two and four.

Leg Four: The fourth leg of the armature movement system, this leg is attached to legs three and one. The end of this leg houses the multichannel dispenser.

Upper Motor Mount: Houses the upper stepper motor, and the front and rear limit switches.

Lower Motor Mount: Houses the lower stepper motor and attaches directly to the base of the Sidekick.

Motor Cap: Sits on top of the upper motor mount. This is mainly on aesthetic piece to hide exposed wiring.

Front Switch Mount: Houses the front limit switch, attaches to the upper motor mount.

Rear Switch Mount: Houses the rear limit switch, attaches to the upper motor mount.

Adapter Clamp Front: Front clamp to adapt the pump outlet to the smaller diameter output/input tubing.

Adapter Clamp Rear: Rear clamp to adapt the pump outlet to the smaller diameter output/input tubing.**Pump Mount:** Houses up to four LPL Dispense pumps and attaches them to the base of the Sidekick.

96 Well Plate Holder: Holds a 96 well plate for liquid dispensing and attaches to the base of the Sidekick.

50ml Tube Holder: Holds four 50ml falcon tubes, used as a reservoir for dispensing.

SideKickV3.fzz: A PCB blueprint in the Fritzing software's proprietary format.

SideKickV3.zip: The PCB files in a compressed format. This is uploaded to a PCB manufacturing service for production.

Pico Snapshot: A folder containing all the Python scripts necessary for running the Sidekick. These files are uploaded onto the Pico

Assembly Instructions.pptx: A slide deck of instructions for assembling the Sidekick.

4. Bill of materials summary

See Attached Excel Spreadsheet

5. Build instructions

5A: Software/Hardware Set Up

Step 1: Install Micropython onto Pico

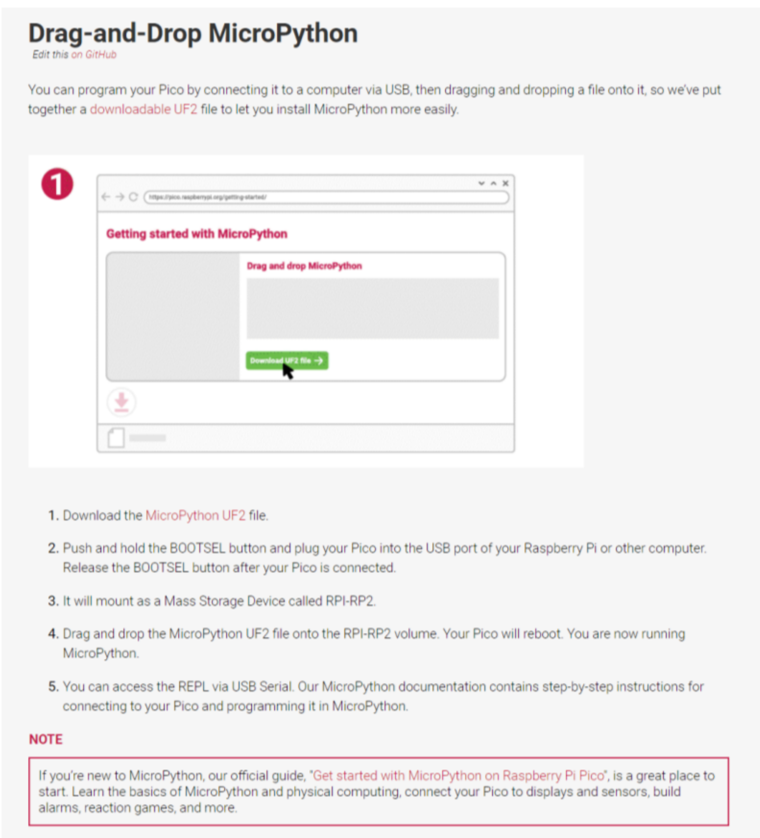


Figure 2: Flashing Micropython onto the Pico

Flash the Raspberry Pi Pico with MicroPython. Follow the directions in this link: https://www.raspberrypi.org/documentation/microcontrollers/micropython.html

Step 2: Install Thonny IDE

Figure 3: The Thonny UI

After flashing MicroPython onto the Pico, we need to upload the SideKick’s code to the microcontroller.

Download the Thonny IDE. This will be used to load the provided code onto the Pico.
https://thonny.org/

Once downloaded, install, and follow the prompts for set up with MicroPython and the Raspberry Pi Pico. Once finished, your Thonny should look like **Figure 3.**

Step 3: Download the Sidekick snapshot

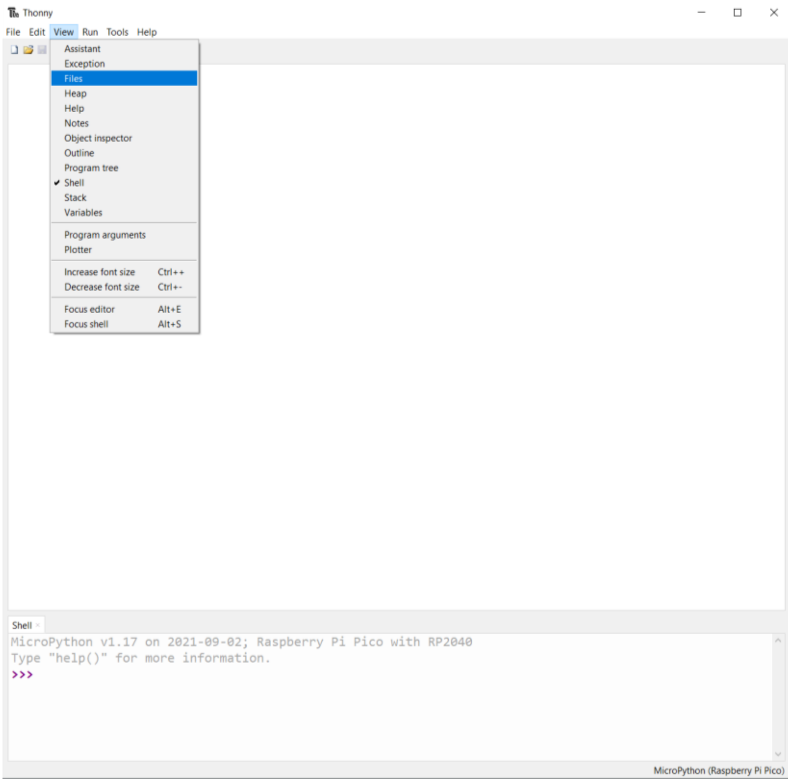


Figure 4: Navigating to the file viewer in Thonny

Go to the SideKick GitHub page to download all the necessary files:

https://github.com/rodolfokeesey/Liquid-Handler

Go to the folder marked “Pico Snapshot” and download the contents. Now return to Thonny. Navigate to View -> Files (Fig. 4).

Step 4: Upload the Sidekick snapshot to Pico

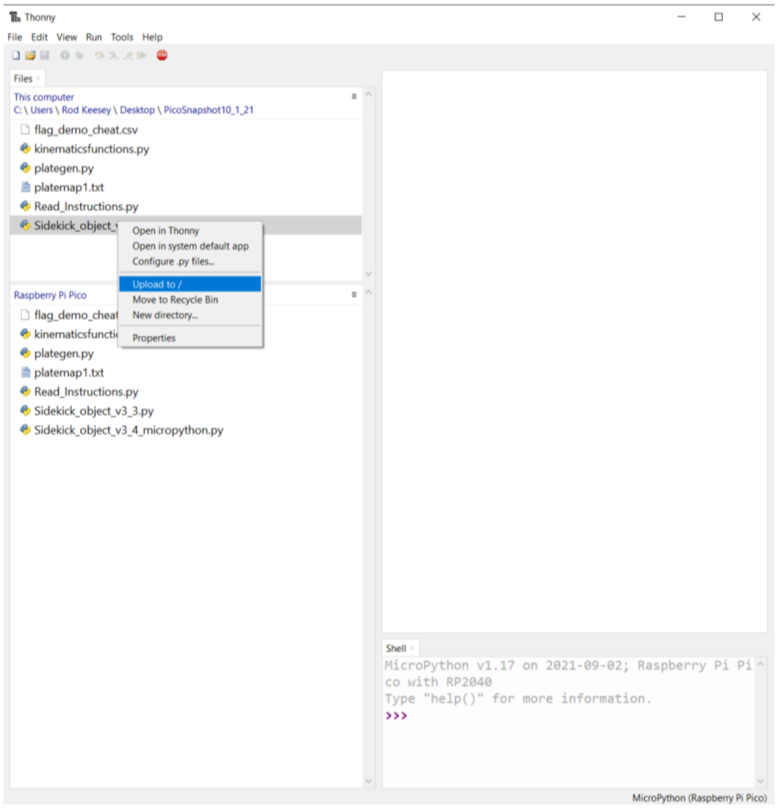


Figure 5: Uploading the Sidekick snapshot

In the upper panel, navigate to where you downloaded the snapshot files. On this machine it's under:

C:\Users\Rod Keesey\Desktop\PicoSnapshot10_1_21

Then, right click on each item in the snapshot, and click "Upload to /" in the dropdown. This saves each of the files onto the Pico (Fig. 5).

Step 5: Opening the main loop

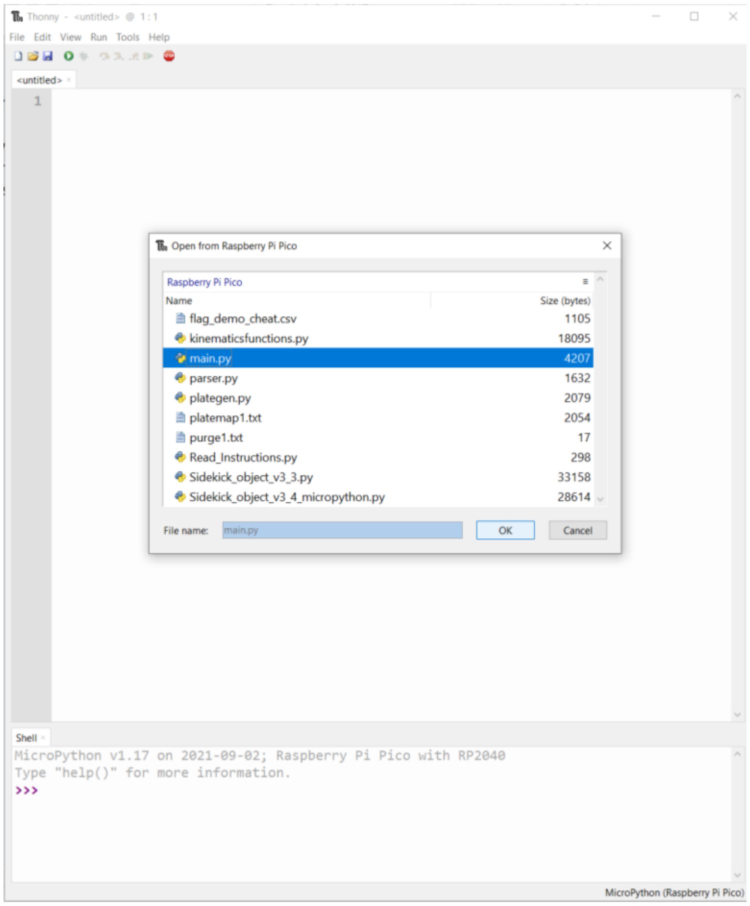


Figure 6: Opening the main.py file to run the Sidekick main loop on the Pico

Once all the files are uploaded, hit the "Open Icon" then select the Raspberry Pi Pico. Open the main.py file.

Step 6: Running the main loop

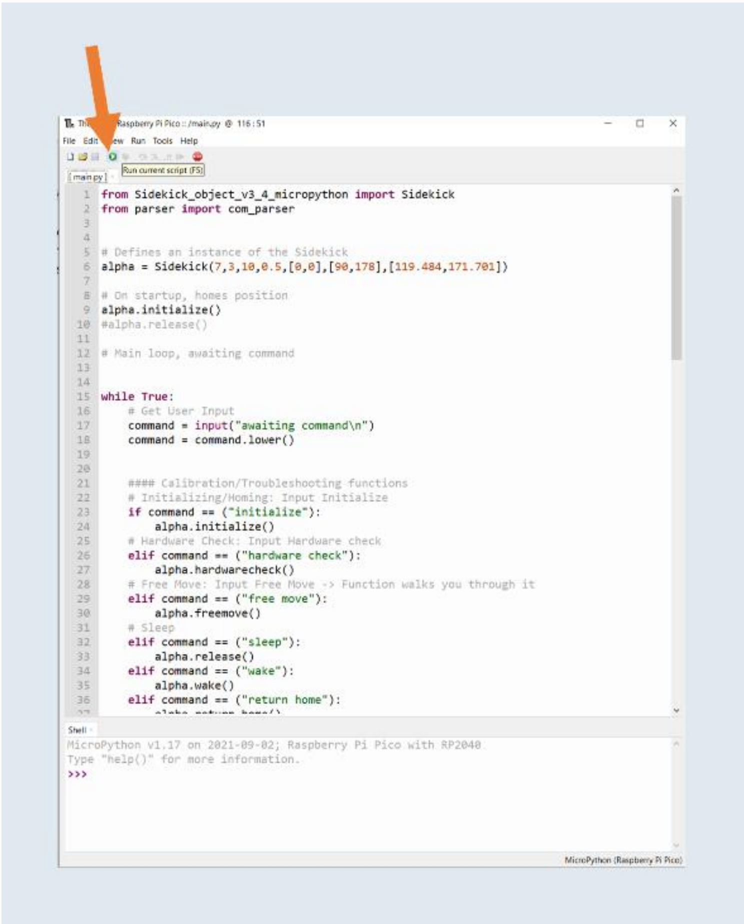


Figure 7: Running main.py

Once opened, hit the green “run current script” button (Fig. 7). This initializes the main loop of the robot. Because the SideKick is not currently attached to any hardware, nothing will happen. Just disconnect the Pico from the computer. Do not hit the “Stop” button in Thonny. The main loop should be left running. The Pico is now flashed with the SideKick’s code.

Step 7: Ordering the custom PCB

Product Detail

Gerber file:	SideKickV3_Y4	Build Time:	1-2 days
Layers:	2	Dimension:	81.8 mm* 69.3 mm ~~60mm* 82mm~~
PCB Qty:	5	Different Design:	1
Delivery Fomat:	Single PCB	PCB Thickness:	1.6
Impedance:	no	Layer stackup:	
PCB Color:	Green	Silkscreen:	White
Surface Finish:	LeadFree HASL-RoHS	Deburring/Edge rounding:	No
Outer Copper Weight:	1	Gold Fingers:	No
Flying Probe Test:	Fully Test	Castellated Holes:	no
Remove Order Number:	No	4-Wire Kelvin Test:	No
Material Type:	FR4-Standard Tg 130-140C	Paper between PCBs:	No
Appearance Quality:	IPC Class 2 Standard	Confirm Production file:	No

Figure 8: The JLCPCB order options for the Sidekick PCB

The SideKick has a custom PCB designed in Fritzing. The files for the PCB are in the "PCB" folder on the SideKick's GitHub page. We used JLCPCB to manufacture a set of 5 boards (https://jlcpcb.com/). Upload the SideKickV3.zip file to JLCPCB. The settings for the board are pictured in **Figure 8**.

Step 8: 3D Printing Sidekick components

Go to the folder marked "3D Assets" and download all files. Print the models in PETG. Print 8 copies of the foot.stl file. Print 4 copies of Adapter Clamp Front, and 4 copies of Adapter Clamp Rear.
If you do not have access to PETG, you may print them in PLA, but monitor the armature for slippage against the shaft due to motor heat.

Print with the following settings:

15% Infill
Support Everywhere
15% Support Density
3 Wall Perimeter, or equivalent 1.2 mm Perimeter

Our prototype was printed on a Creality CR10S Pro V2, using Overture PETG. The layer heights were set to .42mm on a .6mm nozzle, and printed with a 80°C bed temperature and 240 °C nozzle temperature.

5B: Preliminary Wiring

Step 1: Prepare wires

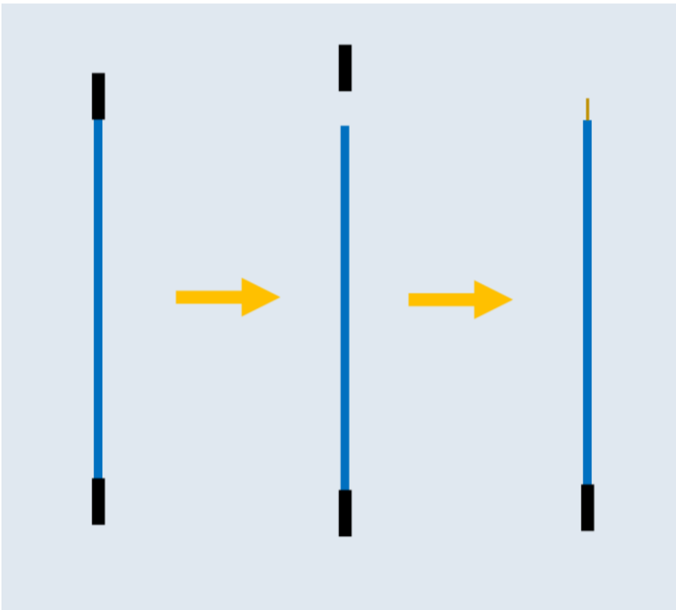


Figure 9: Female-female Dupont Cables

Cut one of the ends from a female-female Dupont connector, then strip the remaining wire to expose the bare wire (Fig. 9). Use these stripped wires for steps 1-3. **Be sure to match the wire colors in the diagram.** This will allow you to easily follow the instructions for connecting the cables into the PCB.

Step 2: Splice Dupont Connectors to Stepper Motors

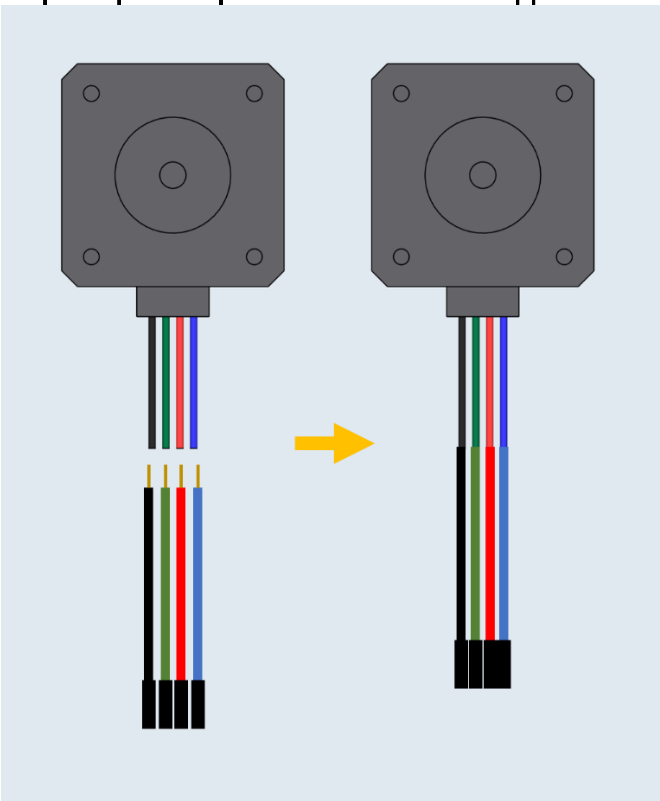


Figure 10: Nema 17 Pancake Stepper Motor, Stripped Dupont Cables

Use wire strippers to strip away the insulation from the ends of the stepper motor wires. Then, splice matching colored Dupont cables prepared in Step 1. This elongates the stepper motor cables and allows them to connect to the header pins of the PCB. Repeat for the second stepper motor.

Step 3: Connecting wires to the Limit Switches

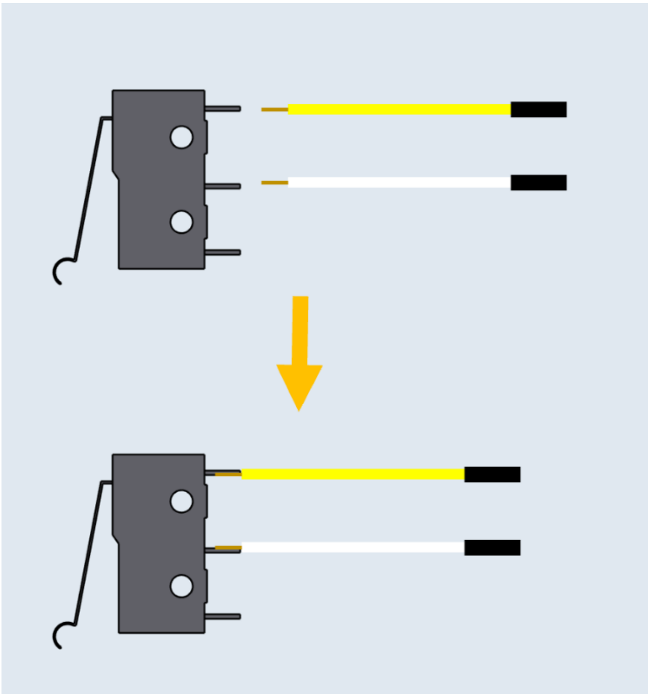


Figure 11: Limit Switch, Stripped Dupont Cables

Solder the Dupont cables from Step 1 onto the contacts of the limit switch (Fig. 11). Repeat for the second limit switch.

Step 4: Wiring the Purge Button

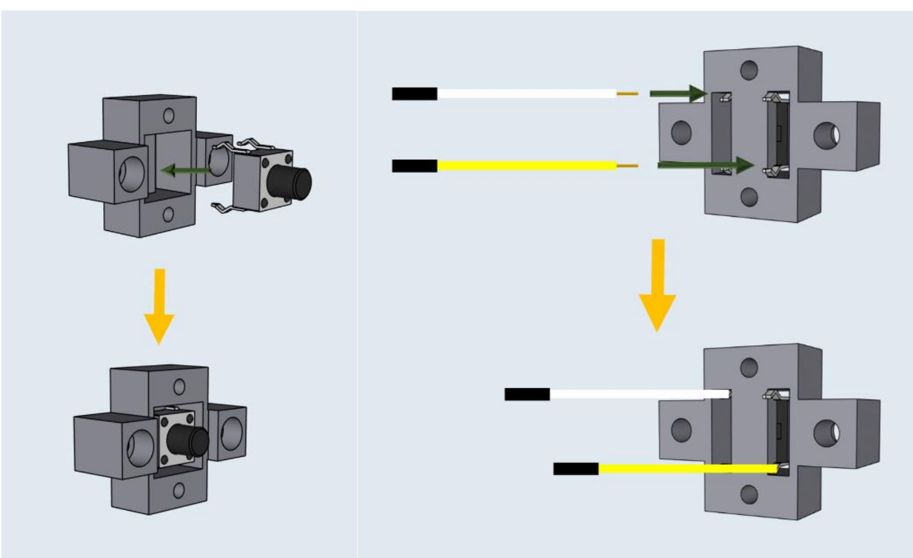


Figure 12a: Button Housing, Button *Figure 12b: Button Housing, Stripped Dupont Cables*

Insert the purge button into the printed purge button holder (Fig. 12a). Then solder the wires from Step 1 onto the contacts of the purge button (Fig. 12b).

Step 6: Attaching Dupont Cables to LPL Pumps

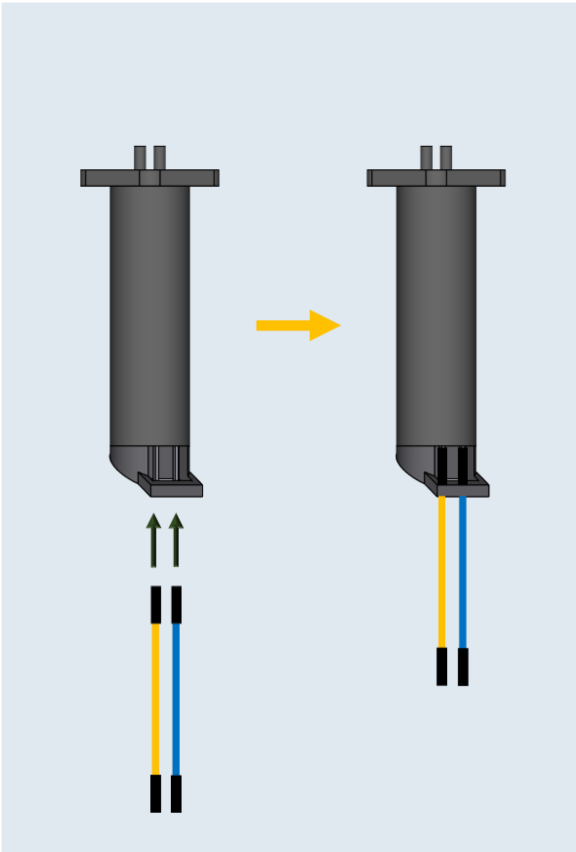


Figure 13: LPL Dispense Pump, Female-Female Dupont Cables

Attach two uncut female-female Dupont connectors onto the two contacts of the LPL pump. Repeat three more times for the rest of the pumps.

5C: Armature

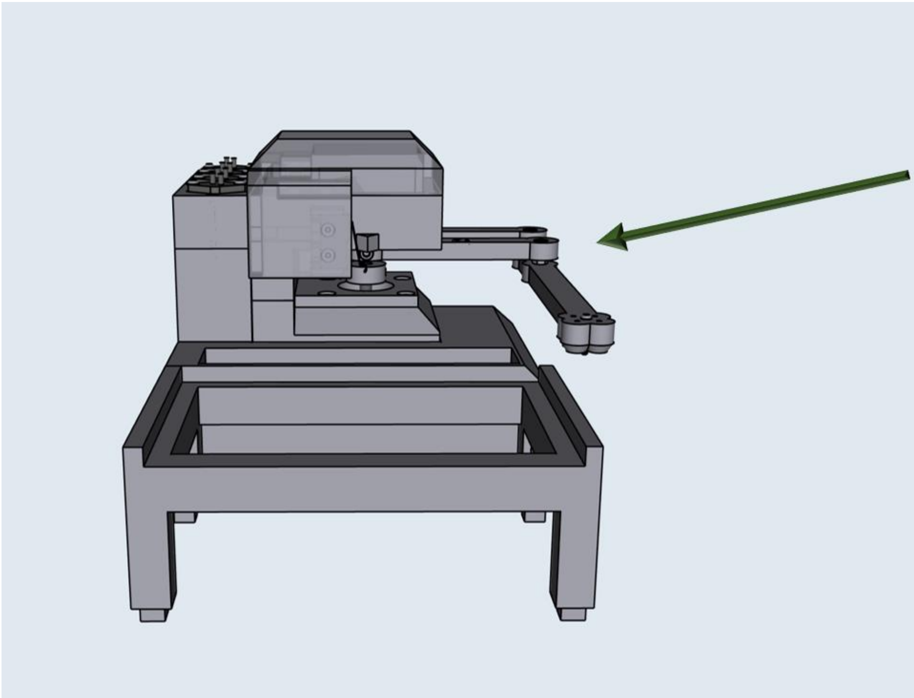


Figure 14: Assembling the Sidekick Armature

Step 1: Assembling Leg One

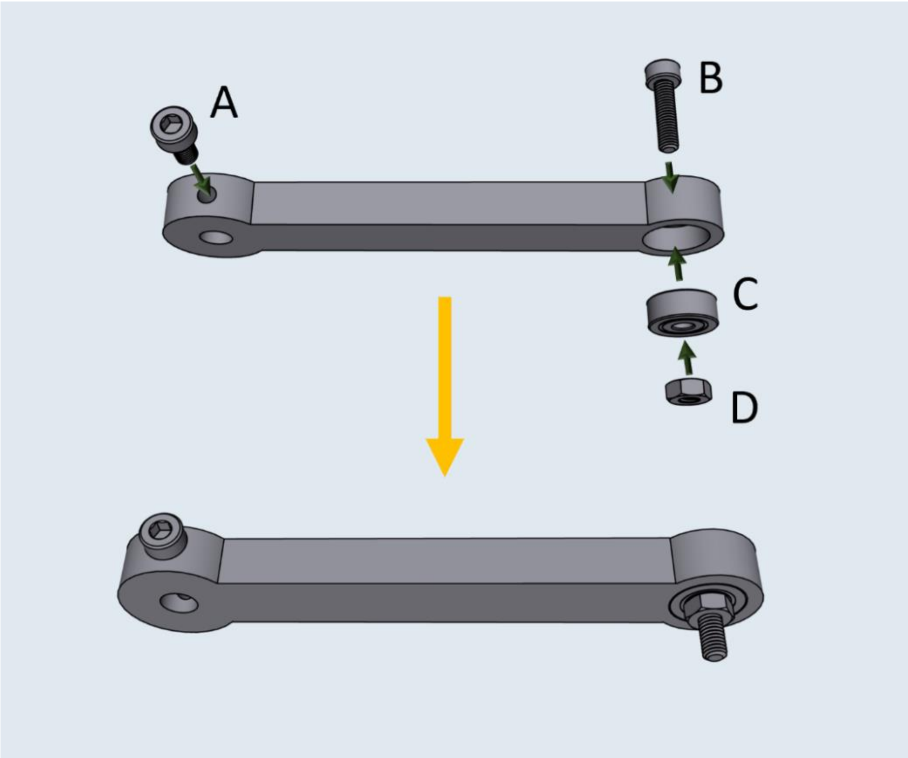


Figure 15: Assembling Leg One. A: M3 x 6 screw. B: M3 x 12 screw. C: 623-2Z Ball Bearing. D: M3 Hex nut.

Thread an M3 x 6 (A) screw into leg one. Press fit the ball bearing (C) into the housing in leg one, then pass the M3 x 12 screw through the bearing, and thread an M3 hex onto the end (Fig. 15). The hex nut should be touching the bearing and screwed tightly onto the M3 x 12 screw.

Step 2: Assembling Leg Two

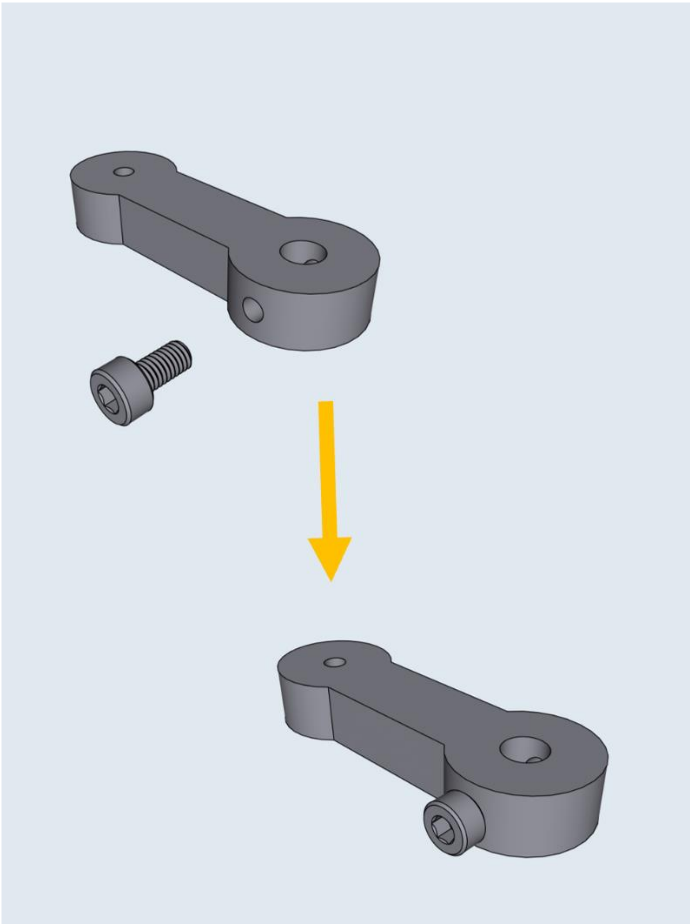


Figure 16: M3 x 6 screw and L2

Assemble leg two by threading in an M3 x 6 mm screw (Fig. 16).

Step 3: Assembling Leg Three

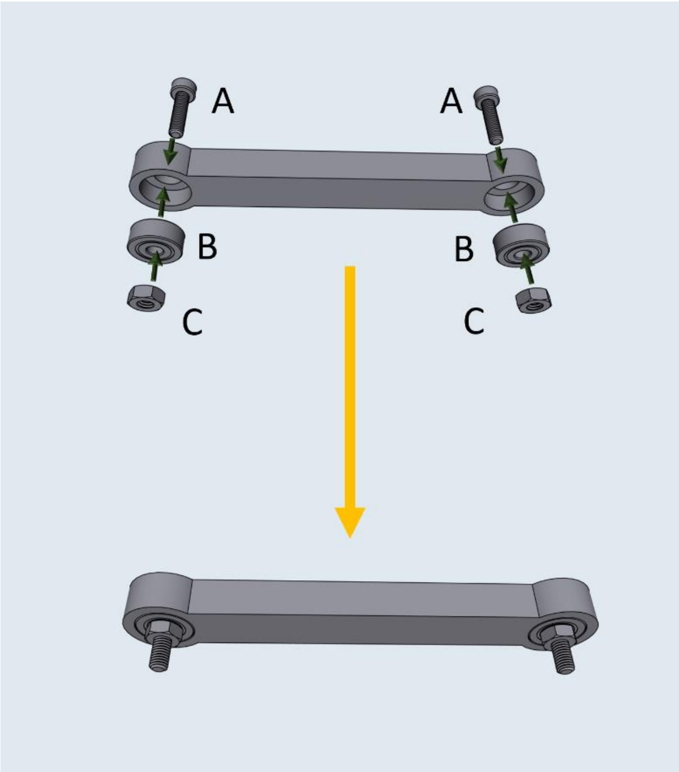


Figure 17: A: M3 x 12 screw. B: 623-2Z Ball Bearing. C: M3 Hex nut.

Press fit the ball bearing (B) into the housings, then pass the M3 x 12 screw (A) through each bearing, and thread an M3 hex nut (C) onto the ends (Fig. 17). The hex nuts should be touching the bearing and screwed tightly onto the M3 x 12 screw.

Step 4: Connecting Legs Two and Four

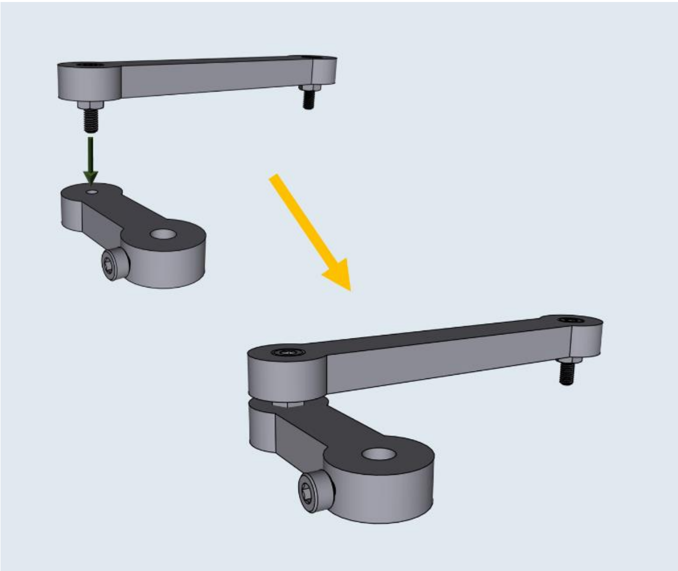


Figure 18: Leg Two, Leg Three

Thread the remaining length of the M3 x 12 screw of L3 onto L2 (Fig. 18). Both ends of L3 are identical, so it does not matter which side is attached to L2. After attaching, rotate L3 to check for any binding. The leg should be able to rotate freely.

Step 5: Prepare the center point of Leg Four

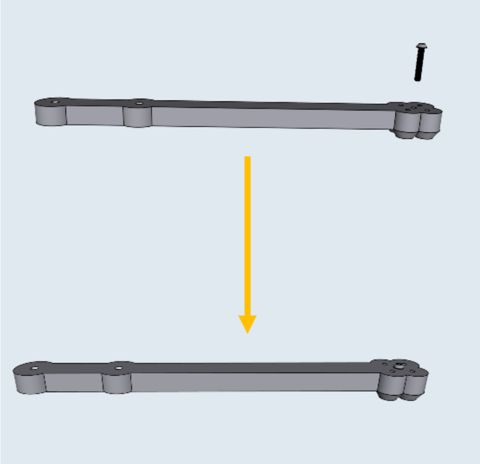


Figure 19a: M3 x 12 screw, Leg Four

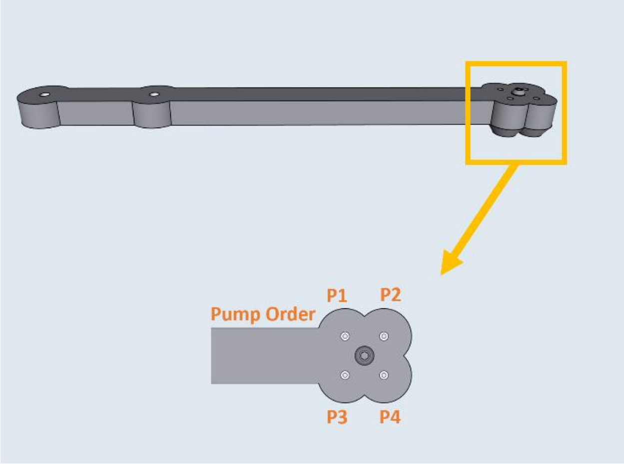


Figure 19b: A diagram of the dispense nozzle

Thread the M3 x 12 screw into the center point of L4, as indicated by (Fig. 19a).

5D: Motor Mounts

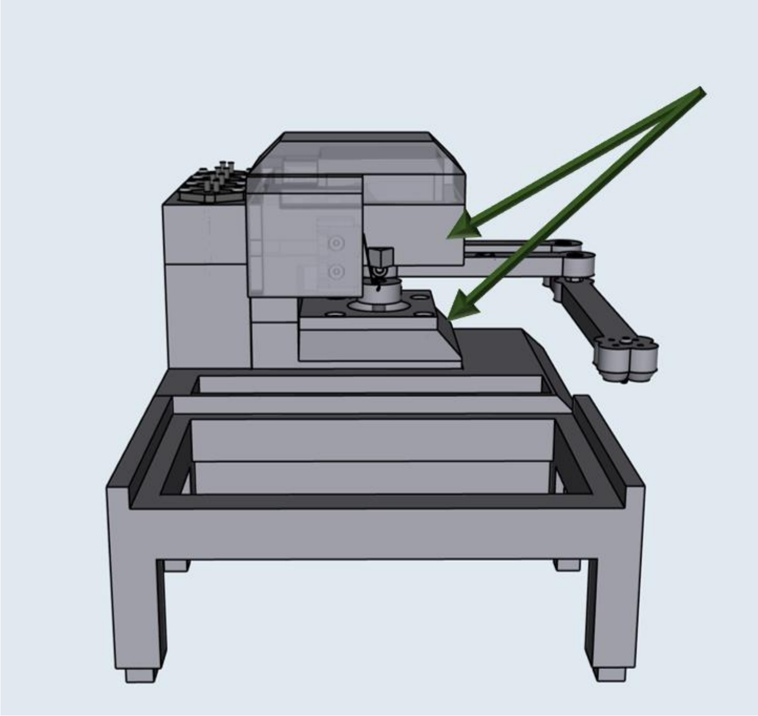


Figure 20: The Sidekick Motor Mount Assembly

Step 1: Mounting the Top Stepper Motor

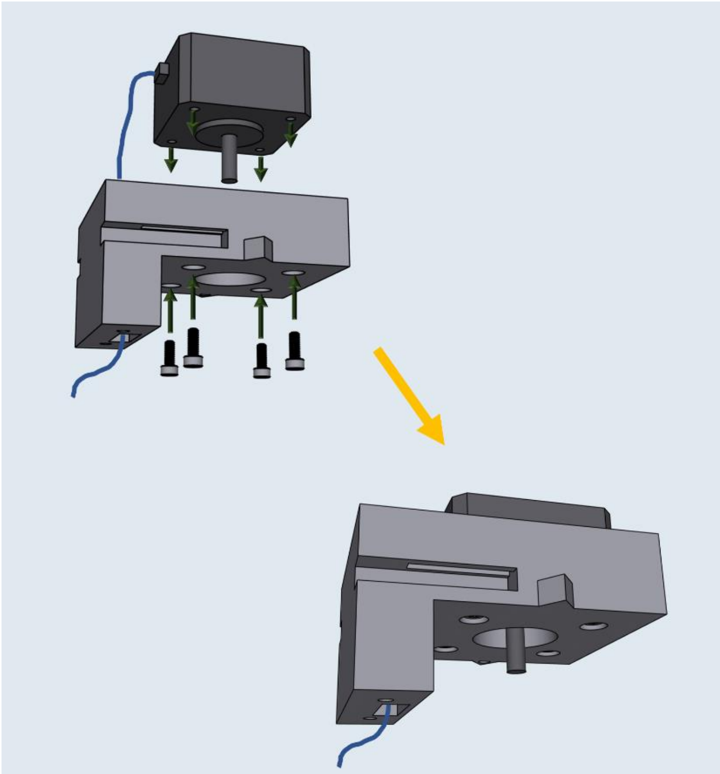


Figure 21: M3 x 8 mm screws, Stepper Motor, and Upper Stepper Motor Mount.

Place the stepper motor into the top mount, with the stepper motor wires facing the cable management channel. Then secure the motor in place with four M3 x 8 screws (Fig. 21).

Step 2: Attaching Limit Switches to Switch Mounts

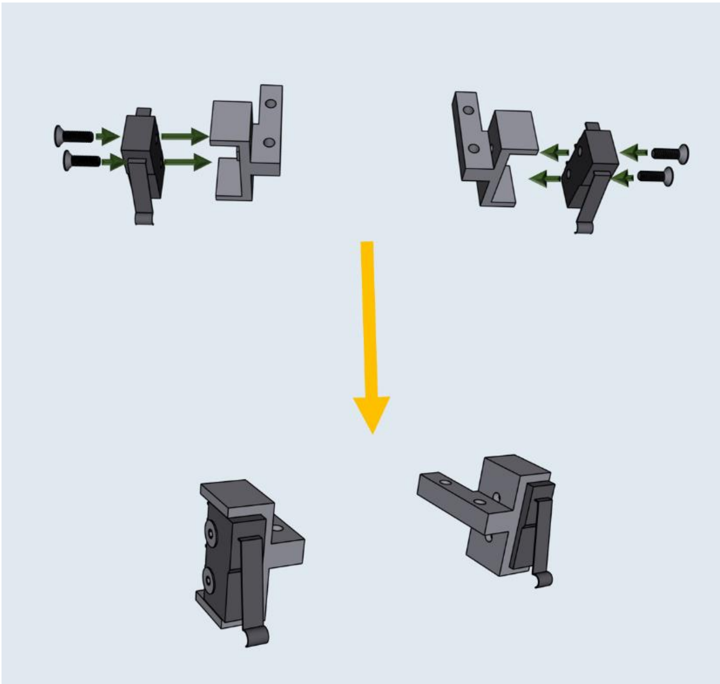


Figure 22: M2.5 x 8 screws. Limit Switches, Limit Switch Mounts

Mount the Left and Right Limit Switches on the Limit Switch Mounts and secure them in place with two M2.5 x 8 screws (Fig. 22).

Step 3: Attaching Limit Switch Mounts onto Top Motor Mount

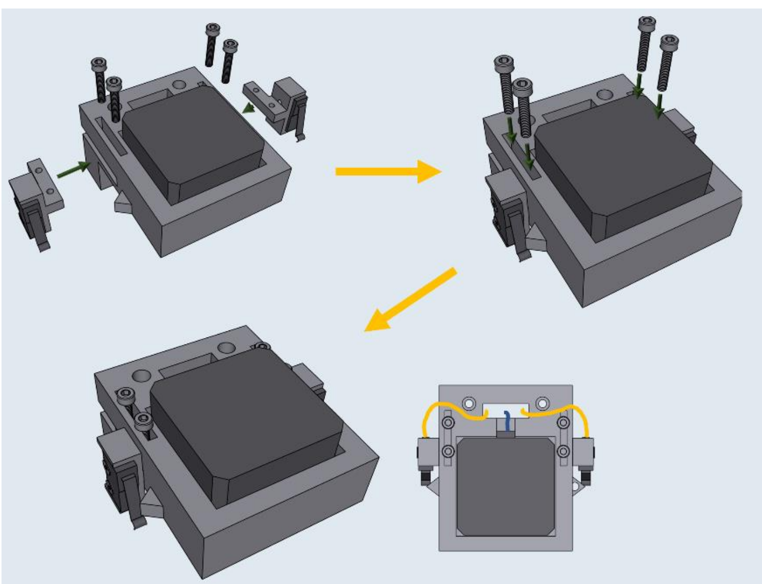


Figure 23: M3x16 screws. Limit Switch and Mounts, Upper Motor Mount.

Mount the Limit Switches into the notches of the Upper Motor Mount. Then screw two M3 x 16 screws onto the Limit Switch mounts to secure them to the Upper Motor Mount. The colored lines indicate the cable management pathing (Fig. 23).

Step 4: Attaching Leg One onto the Upper Motor Assembly

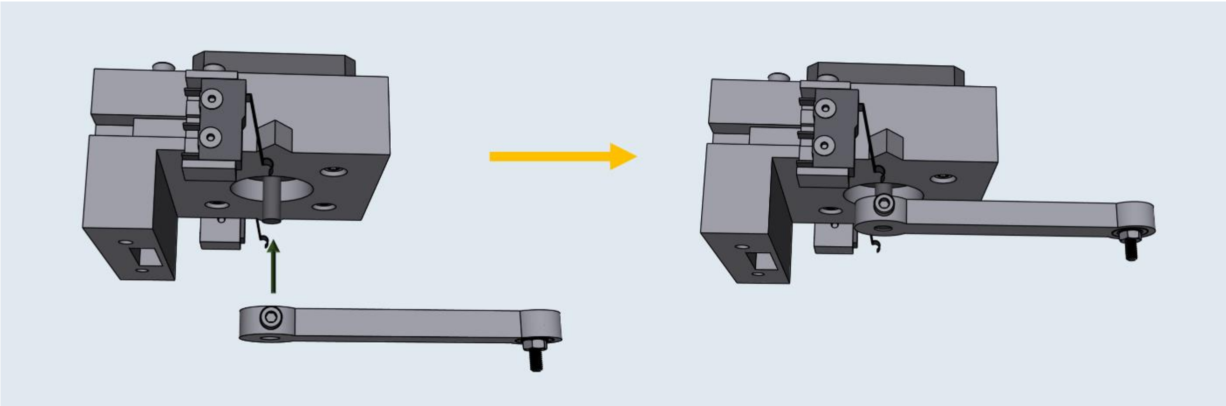


Figure 24: Upper Motor Assembly, Leg One

Press fit Leg One onto the shaft of the Upper Stepper motor as indicated in (Fig. 24). The leg should be able to rotate and engage the limit switches, without scraping against the top of the motor mount.

Step 5: Tightening Leg One onto the Upper Motor Assembly

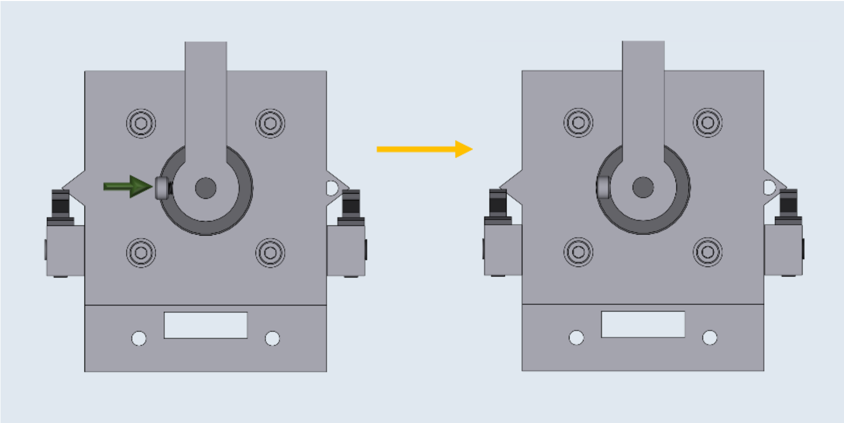


Figure 25: Tightening the M3 x 6 screw

Once satisfied with the position of Leg One, tighten the M3 x 6 screw on Leg One to secure it to the motor shaft (Fig 25).

Step 6: Setting Home Position for Leg One

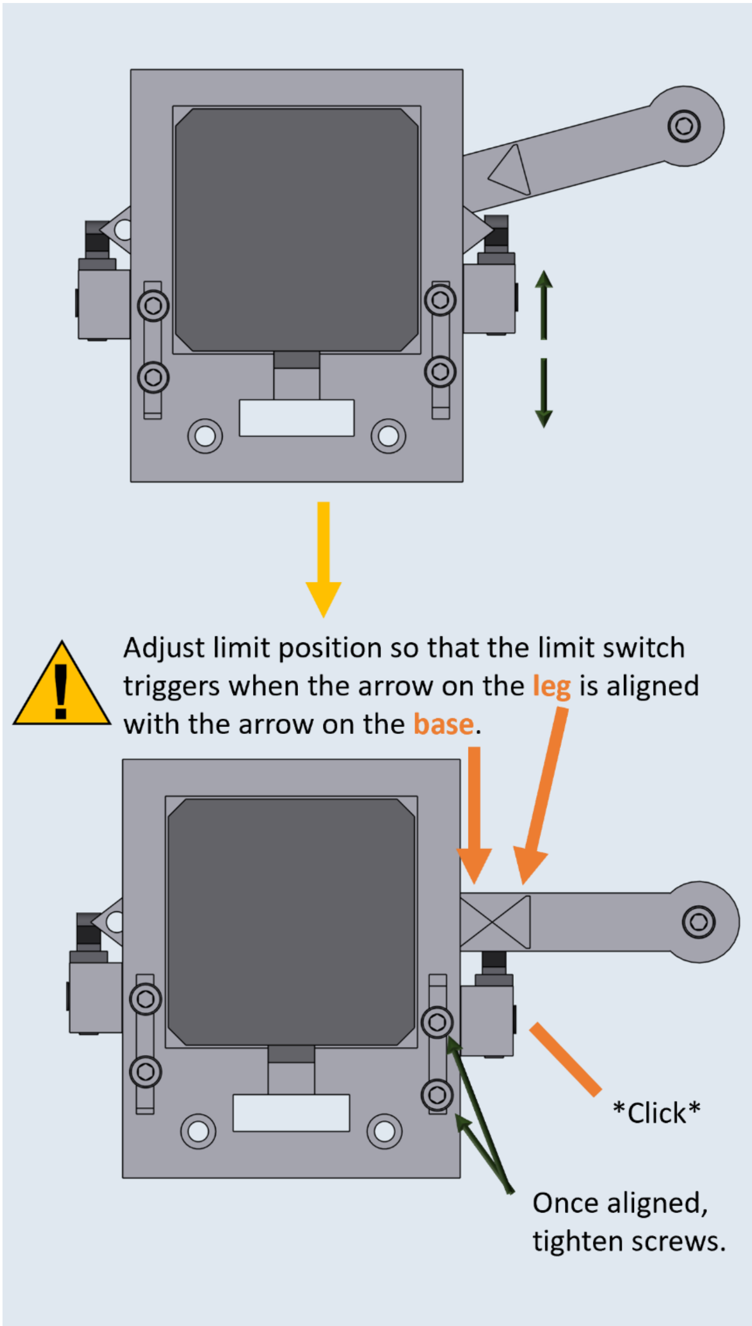


Figure 26: Adjusting the Front Limit Switch Home Position

Adjust the Front Limit switch mount, so that the limit switch is engaged once Leg One is at the correct zero position, indicated by the alignment of the triangle on the Upper Motor Mount (Fig. 26). Once the Front Limit Switch Mount is in the correct position, tighten the two M3x16 screws.

Step 7: Mounting the Lower Stepper Motor

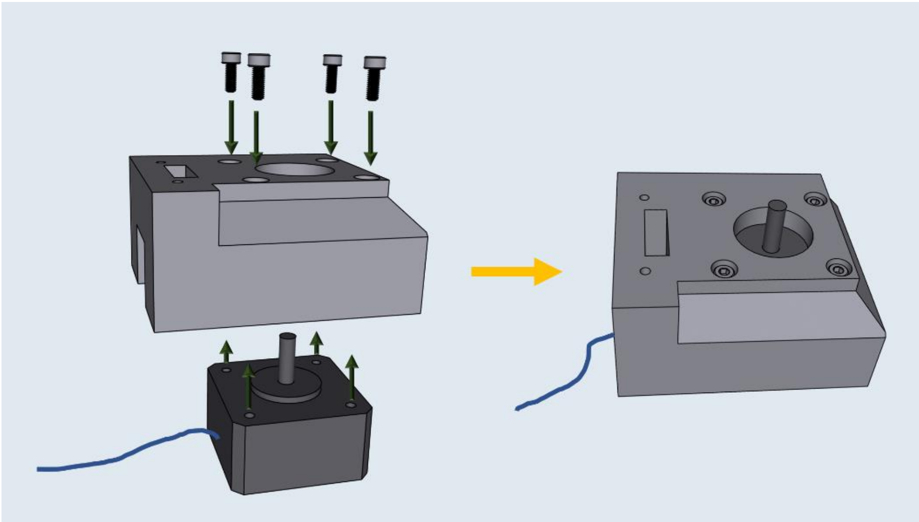


Figure 27: M3x8 screws, Stepper motor, Lower Stepper Motor mount

Place the stepper motor into the lower mount, with the stepper motor wires facing the cable management channel. Then secure the motor in place with four M3 x 8 screws (Fig. 21).

Step 8: Attaching Leg Two and Three to the Lower Motor Assembly

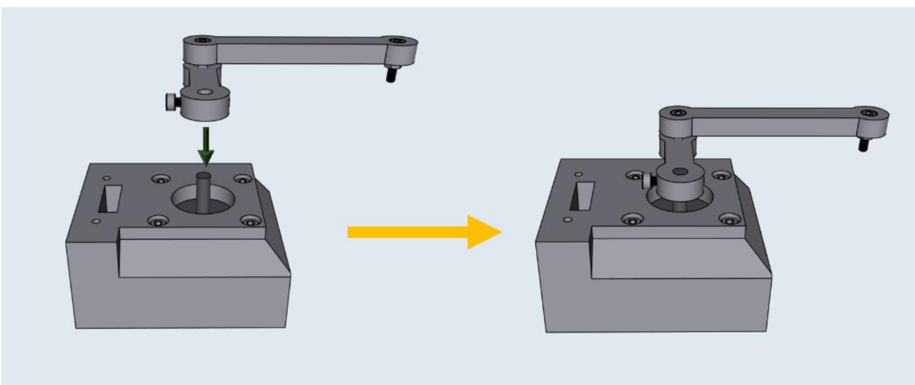


Figure 28: Lower Motor Mount Assembly, Legs Two and Three

Press fit Leg Two onto the stepper motor shaft with Leg Three facing upwards, as indicated in (Fig. 28). The legs should be able to rotate freely without catching on anything.

Step 9: Tightening Leg Two on the Lower Stepper Motor

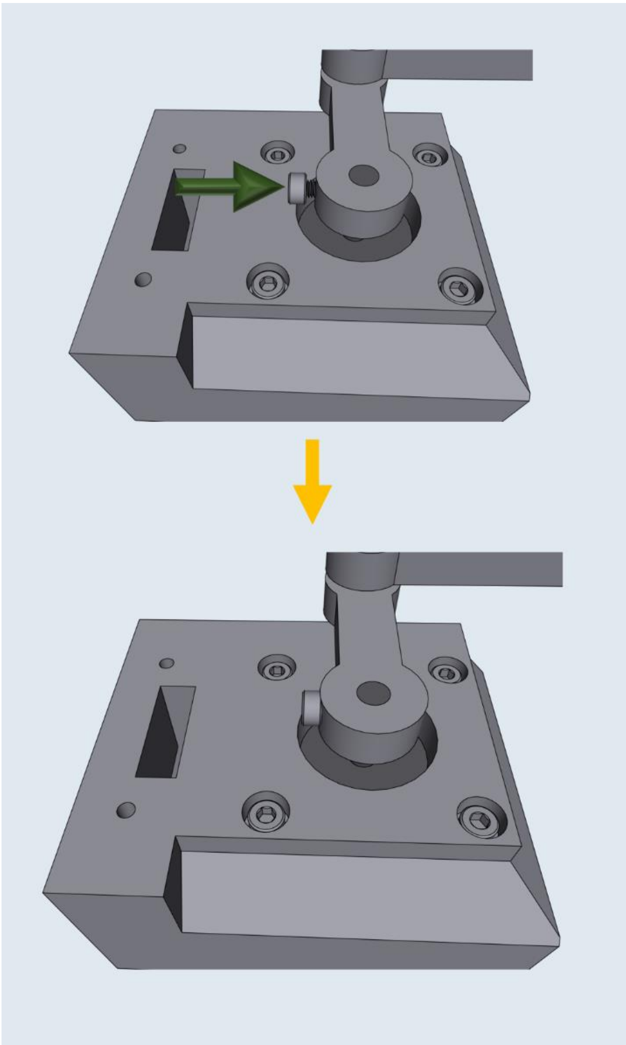


Figure 29: Tightening Legs Two and Three onto the Lower Motor Assembly

Once satisfied with the position of Leg Two, tighten the M3 x 6 screw on Leg Two to secure it to the motor shaft (Fig. 29).

Step 10: Attaching the Upper Motor Assembly to the Lower Motor Assembly

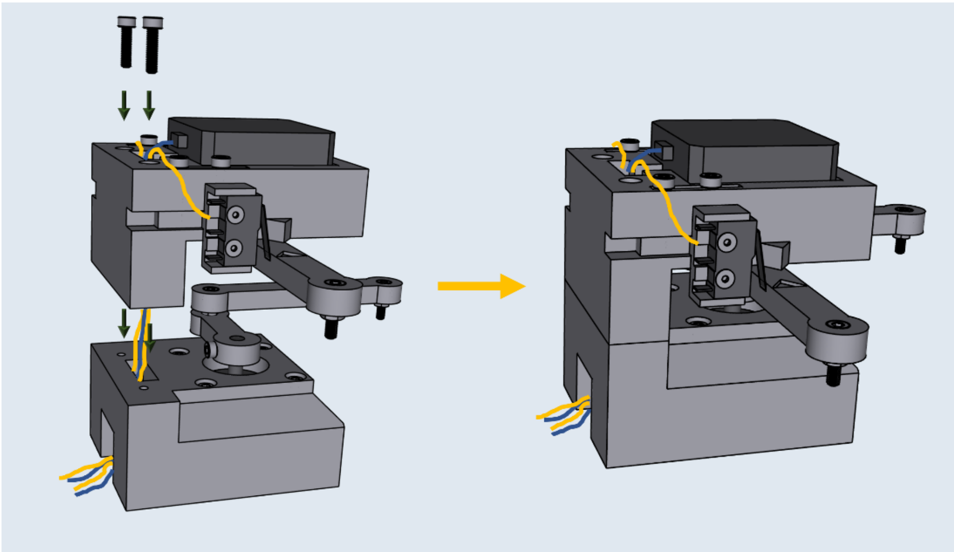


Figure 30: M3x12 screws

Thread the cables from electrical components on the Upper Motor Assembly through the cable management tunnel of the Lower Motor Assembly. Then secure the two together with two M3 x 12 screws.

Step 11: Setting Home Position for Leg Two

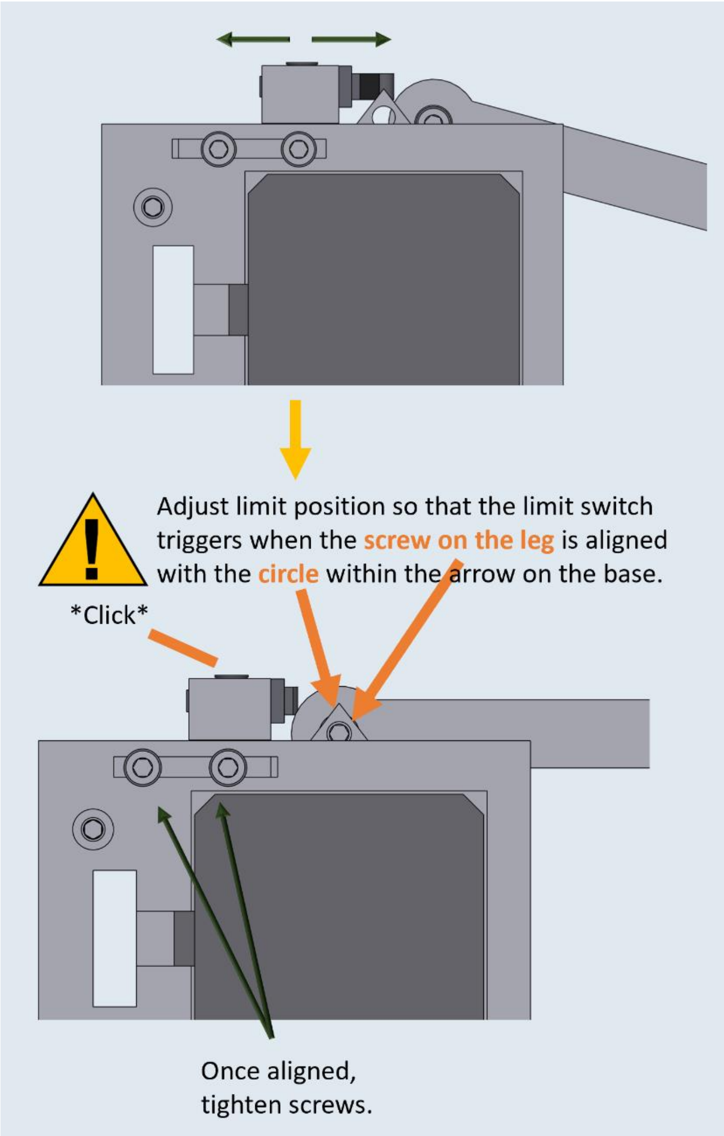


Figure 31: Adjusting the Rear Limit Switch Home Position

Adjust the Rear Limit switch mount, so that the limit switch is engaged once Leg Two is at the correct zero position, indicated by the alignment of screw on the leg aligning with the inscribed circle on the Upper Motor Mount (Fig. 26). Once the Rear Limit Switch Mount is in the correct position, tighten the two M3 x 16 screws.

Step 12: Setting the gap between Leg One and Leg Two

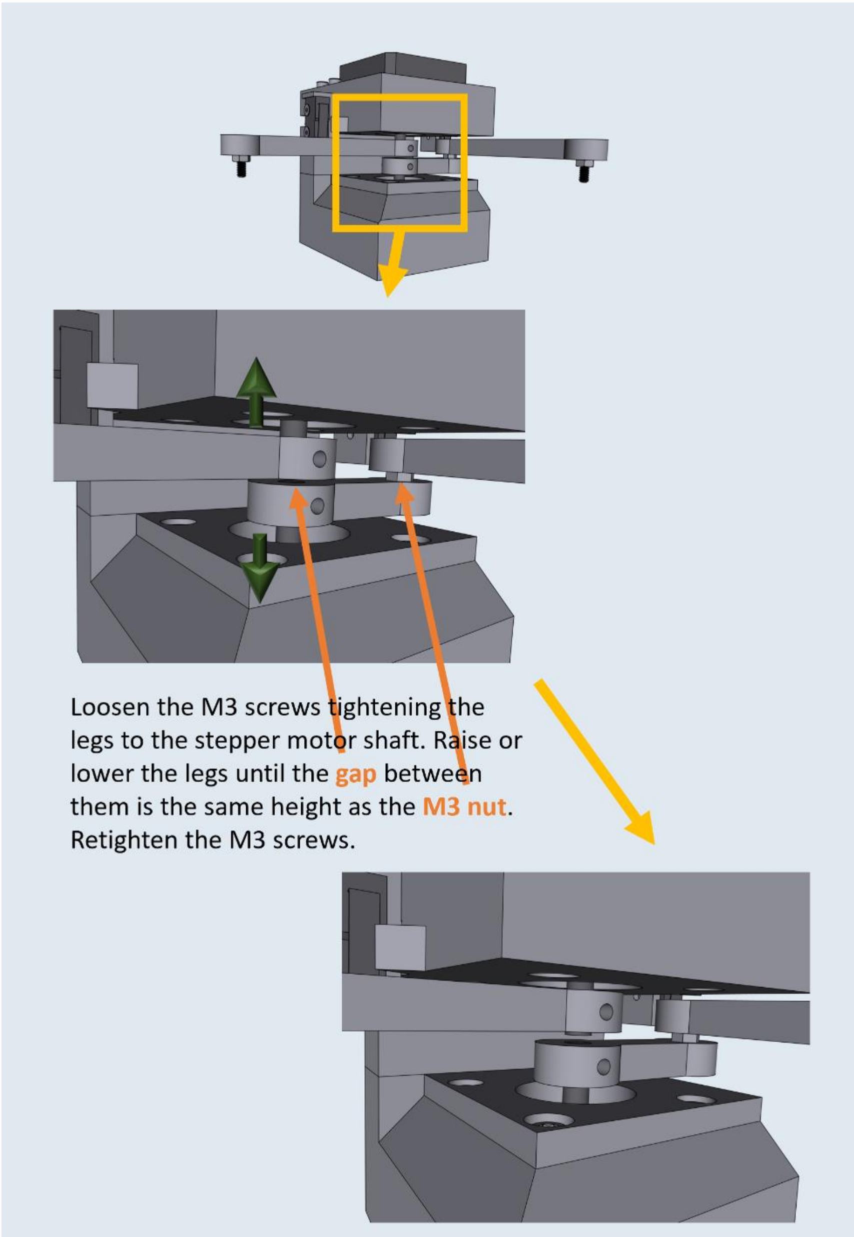


Figure 32: Setting the gap between Legs One and Two

Adjust the gap between Leg One and Leg Two, so that the distance between the two is equal to the height of the M3 nut between Leg Two and Leg Three. This is best done by loosening the two M3 screws holding the legs against the motor shafts and sliding a flat head screwdriver into the gap. When gapping the two legs, be sure that neither leg brushes against the upper or lower motor mount. Once satisfied with the gap, retighten the M3 screws to secure the legs.

Step 13: Attaching Leg Four

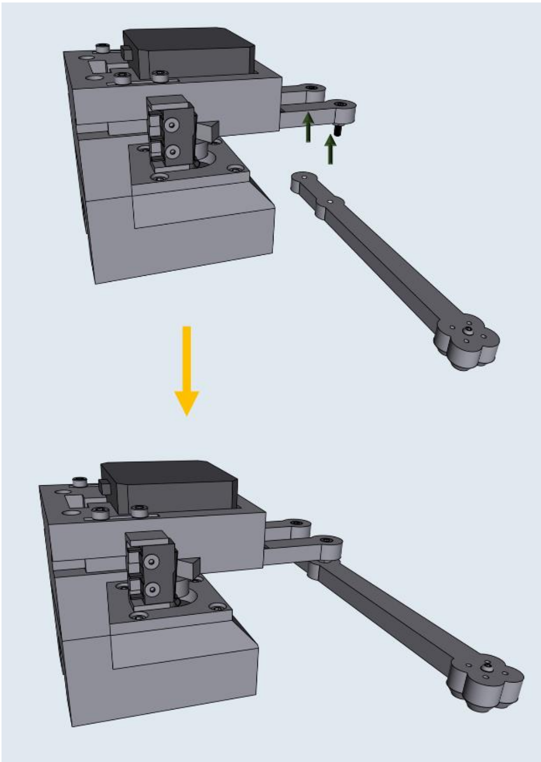


Figure 33: Motor Mount Assembly, Leg Four

Attach Leg Four to Leg One and Leg Three with the nozzle cones facing down (Fig. 33). Thread the M3 screws into Leg Four until the M3 nuts are pressed against Leg Four.

5E: Base Assembly

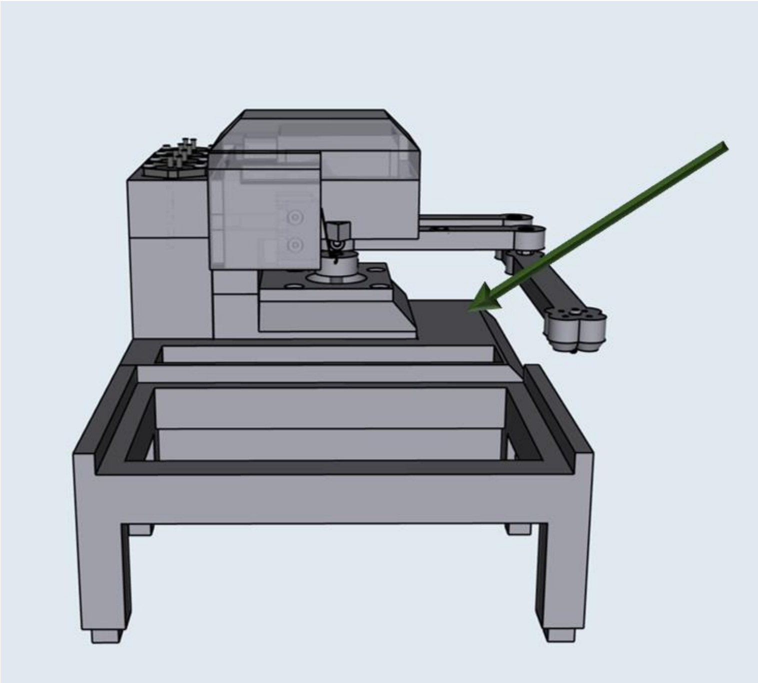


Figure 34: The Sidekick Base Assembly

Step 1: Purge Button Assembly

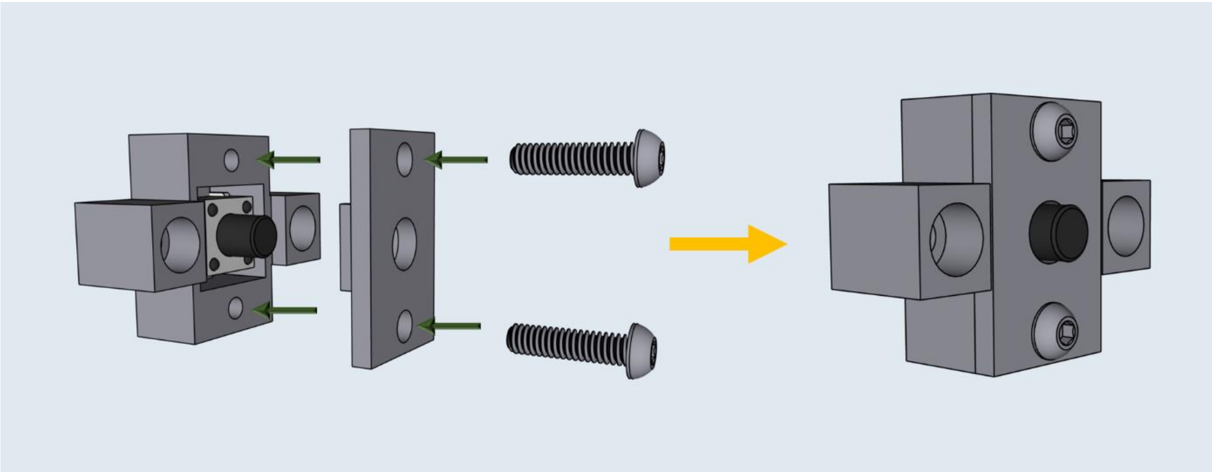


Figure 35: M2 x 8, Button Housing, Button Housing Front

Finish assembling the Button Housing wired in 5B, Step 4 by placing the Button Housing front over the housing and securing with two M2 x 8 screws (Fig. 35).

Step 2: Joining the Button Assembly to the Base

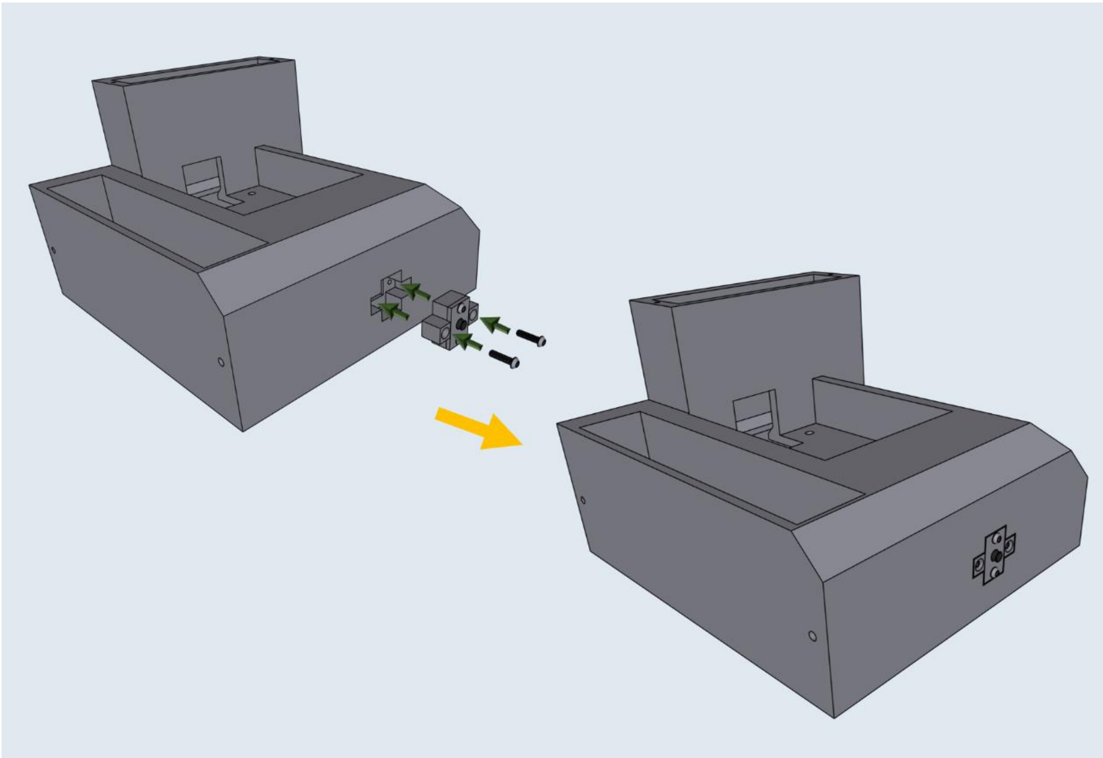


Figure 36: M2 x 8, Button Assembly, Base

Thread the button wires in the cable management slot in the base. Then press fit the Button Assembly into the Base and secure the assembly with two M2 x 8 screws.

5F: Base, Final Assembly

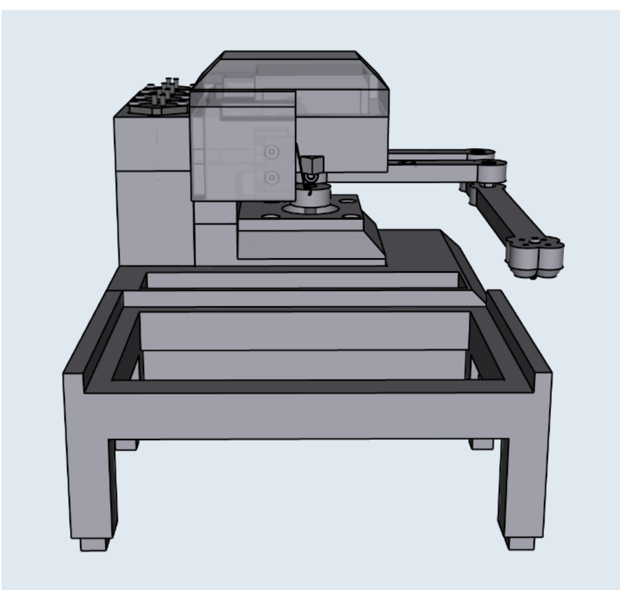


Figure 37: Final Assembly of the Sidekick base

Step 3: Securing Pumps to Pump Mounts

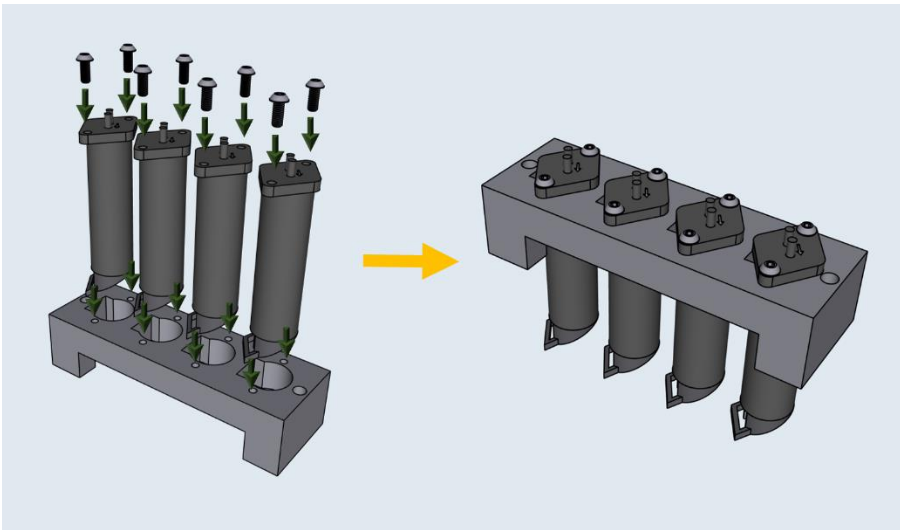


Figure 38: M2.5 x 6, LPL Pumps, Pump Mount

Mount and secure the Pumps to the Pump Mount with M2.5 x 6 screws (Fig. 38).

Step 4: Attaching the Pump Assembly to the Base

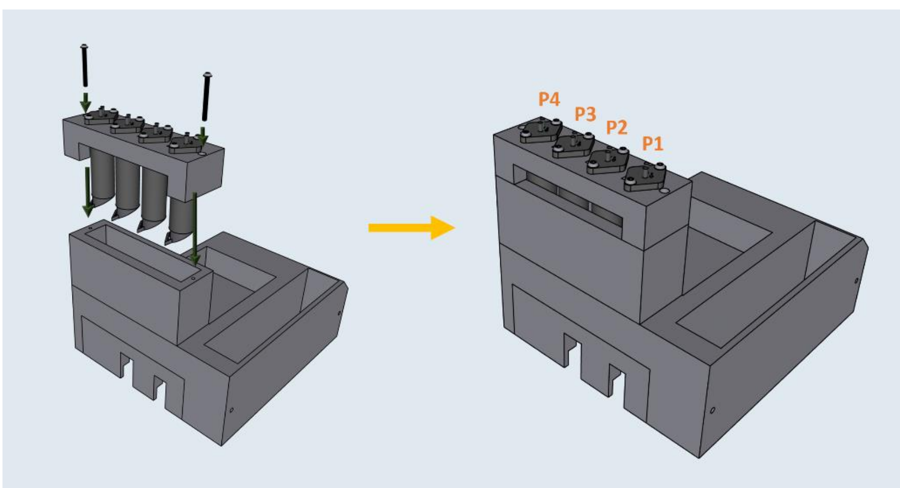


Figure 39: M2 x 20 screws, Pump Assembly, Base

Mount the Pump Assembly to the Base using two M2 x 20 screws (Fig. 39).

Step 5: Attach the Motor/Armature Assembly to the Base

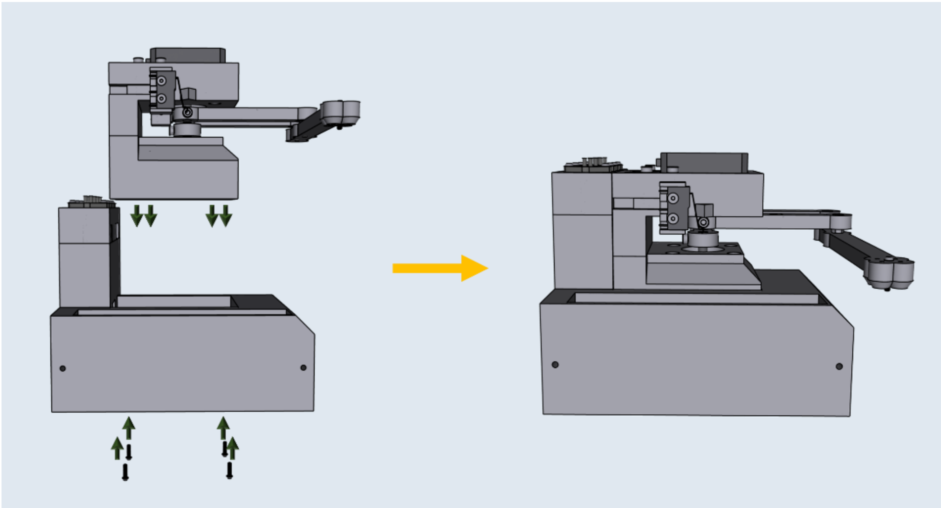


Figure 40: M3 x 8 screws, Motor/Armature Assembly, Base

Mount the Motor/Armature Assembly onto the Base with 4 M3 x 8 screws from the underside of the base (Fig. 40).

5G: PCB Wiring/Assembly

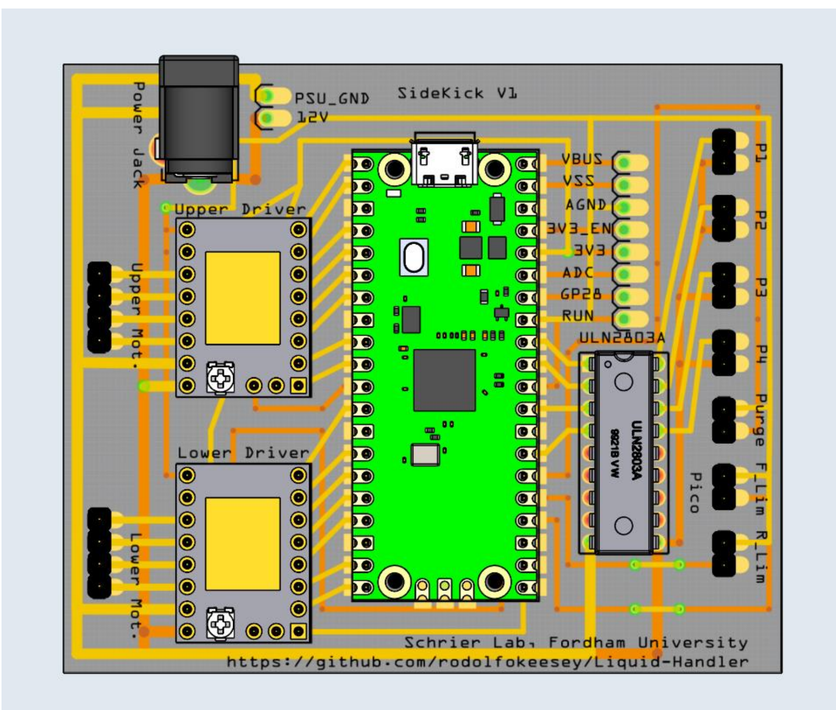


Figure 41: The Sidekick PCB

Step 1: Soldering in the Electrical Components

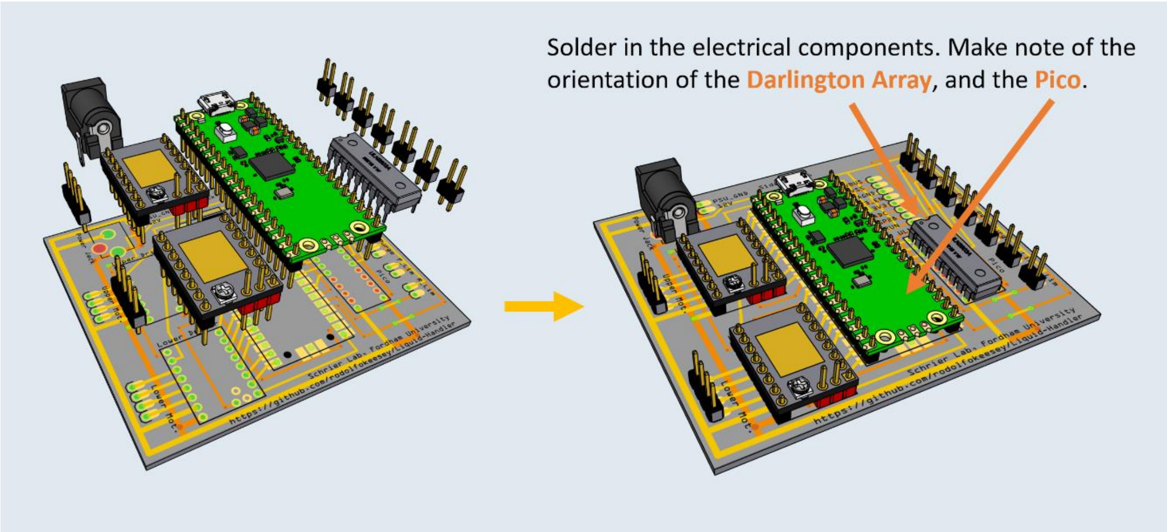


Figure 42: TMC 2209 Stepper Drivers, Raspberry Pi Pico, Darlington Array, Power Barrel, Header Pins and PCB

Through-Pin solder the two TMC 2209 Stepper Drivers, the Raspberry Pi Pico, Darlington Array, Power Barrel, and Header Pins onto the PCB (Fig. 42).

Step 2: Trim the electrical component pins

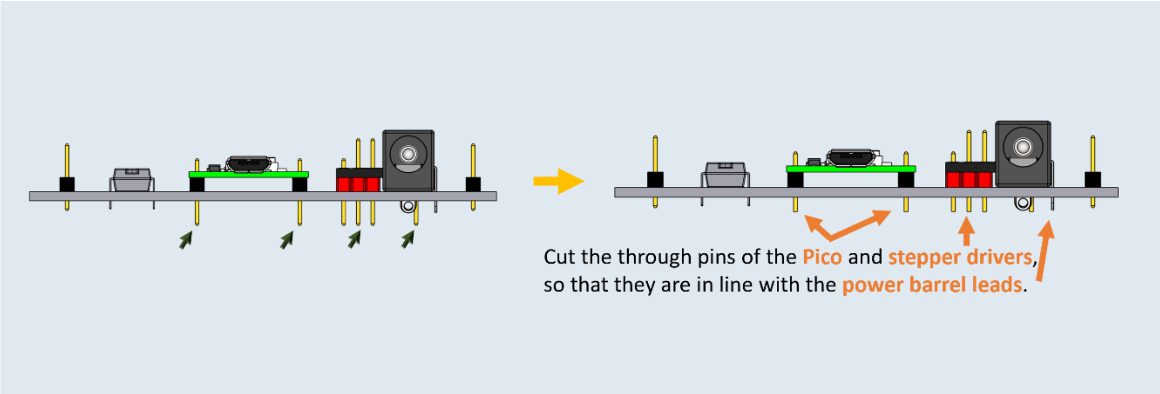


Figure 43: Trimming excess pin length from the electrical components

Cut the excess pin length from the Raspberry Pi Pico, and the two Stepper Drivers (Fig. 43). They should be the length of the Power Barrel leads. This allows for the PCB to slide into the PCB Tray.

Step 3: Wiring the Hardware to the PCB

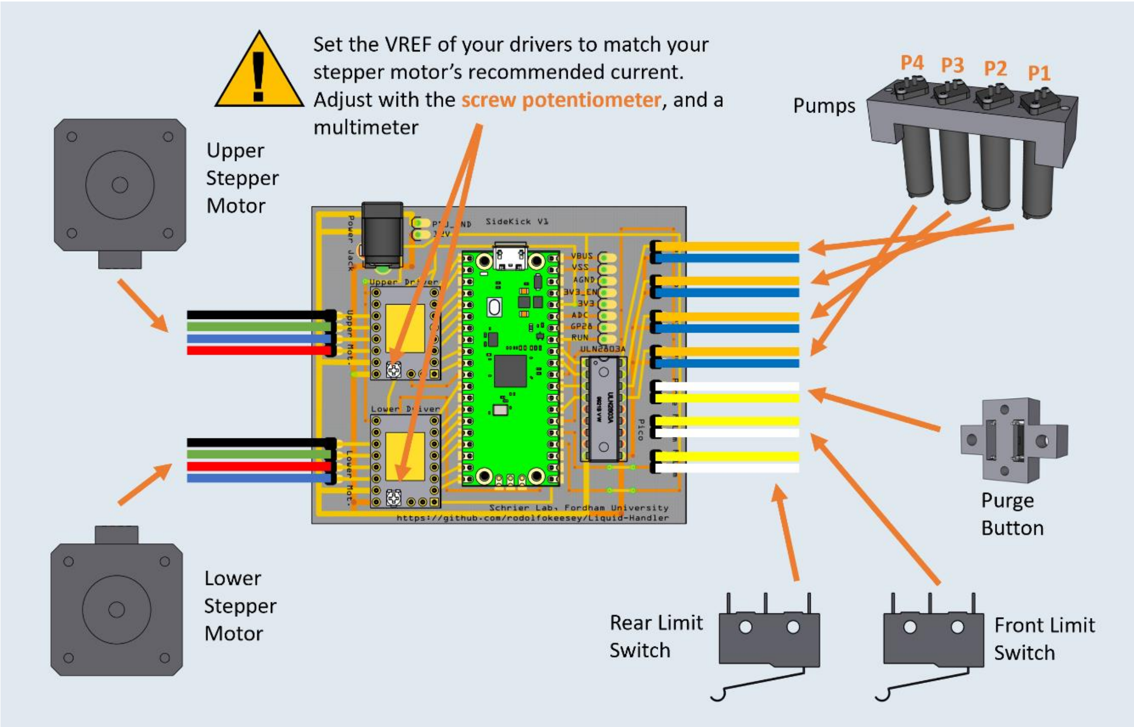


Figure 44: A wiring diagram for connecting the Sidekick's hardware to the PCB

Route all wires from underneath the base of the Sidekick. Plug the female ends of the Dupont connectors into the PCB as diagramed in **Figure 44**. Be sure to match the wiring colors to the pins correctly. After wiring all the hardware, set the VREF on the stepper motors to match the current limits of your chosen stepper motors.

Step 4: Slide the Wired PCB into the PCB Tray

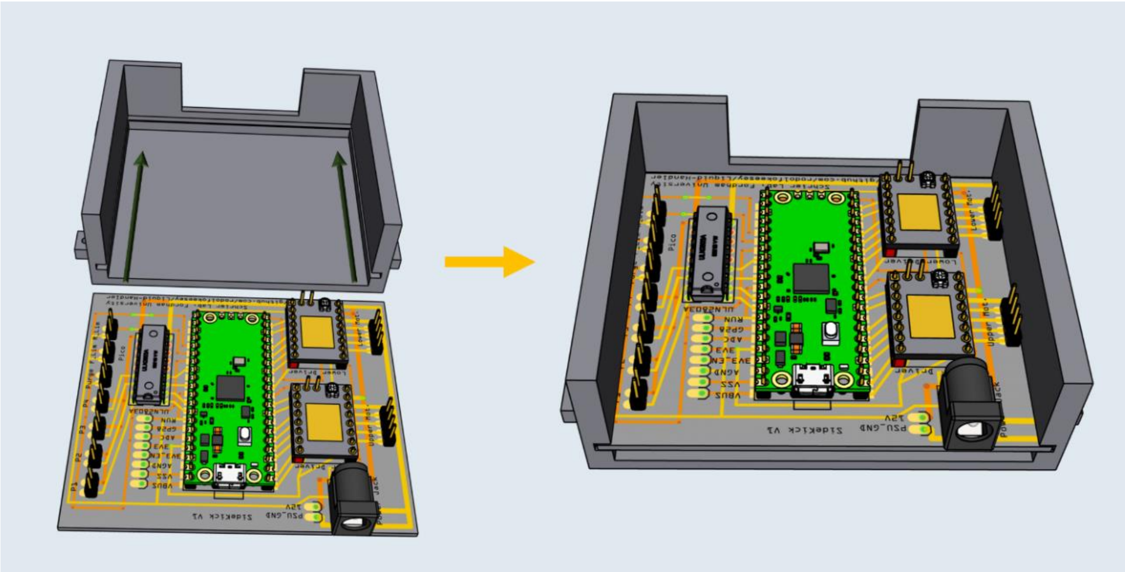


Figure 45: PCB, PCB Tray

Slide the PCB into the PCB Tray (Fig. 45). Route the wires into the gap at the rear of the tray.

Step 5: Ensuring Correct Wiring

Before continuing, make sure that all hardware is wired correctly. Plug in the PCB to power supply, and the USB to a computer. Once the Sidekick is plugged in, it should immediately start to home against the limit switches. If this does not occur, first check the wiring for the limit switches and motors, as the Sidekick will not continue if it cannot home properly.

Next, open Thonny and type into the command line, "hardware check" followed by a return). The Sidekick should home against each limit switch, park the nozzle at a 90° angle, and then cycle through energizing each of the pumps. When a pump is energized, it will make an audible clicking sound. The Sidekick will then ask you to press the purge button. Once pressed, it will output "pressed" and when let go, it will output "released". Once the correct wiring is validated, continue to step 6. If any of the components did not behave as described, inspect their associated wiring.

Step 6: Mounting PCB Assembly to Base

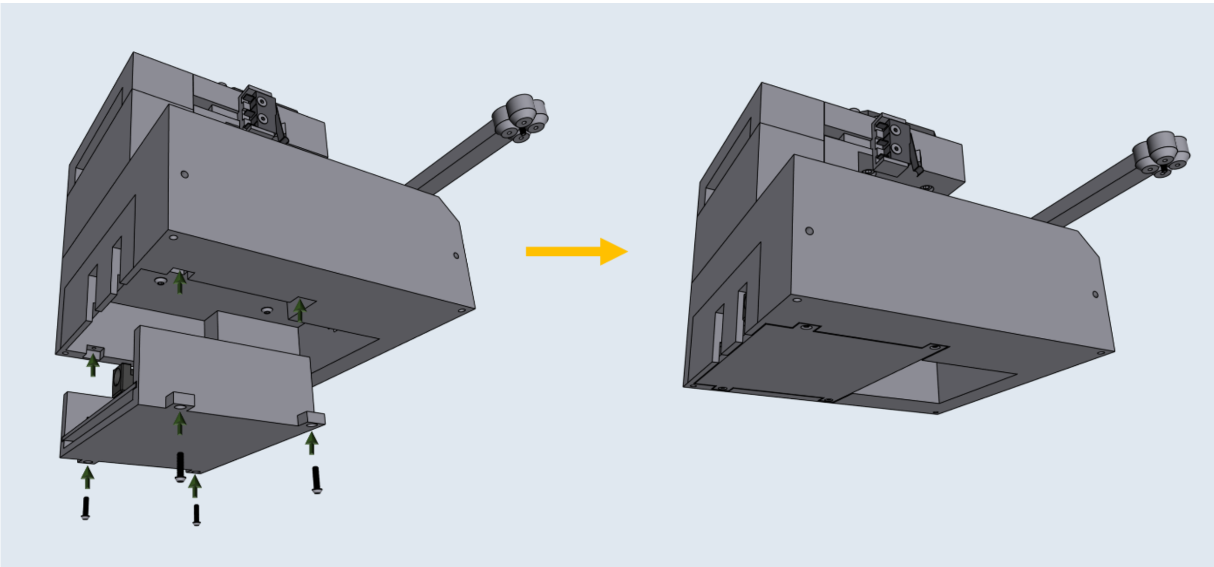


Figure 46: M2 x 8 screw, PCB Assembly, Base Assembly

Route all the wires to the gap in the PCB Tray, and the associated gap in the Base. Then, fit the tray into the base slowly, making sure to keep all the wires routed into the cable management gap. Then thread four M2 x 8 screws to secure the tray in place. This is easiest to do when the Base Assembly is resting on its side, with the underside facing the assembler.

5H: Plate Holder and Feet

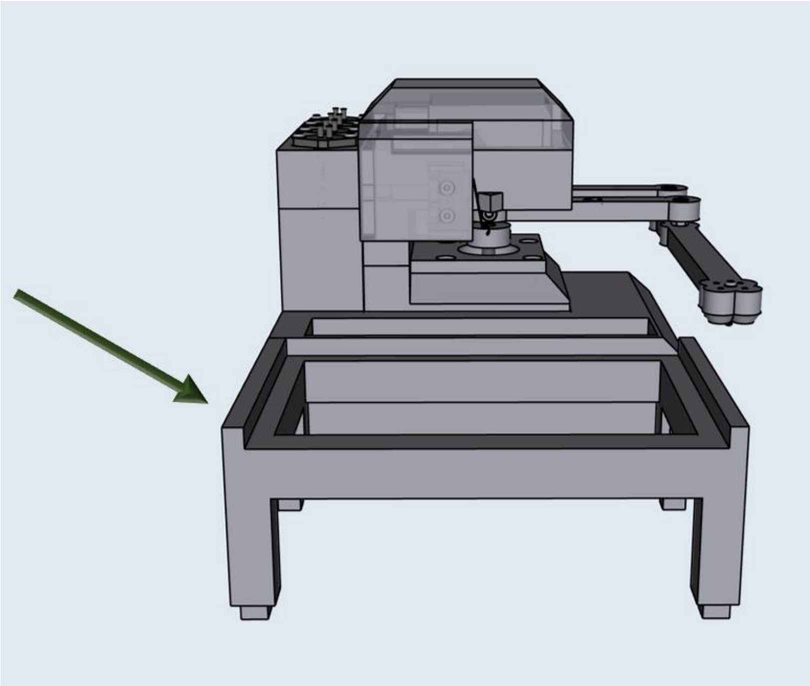


Figure 47: The Sidekick Plate Holder

Step 1: Attach Feet to Base Assembly

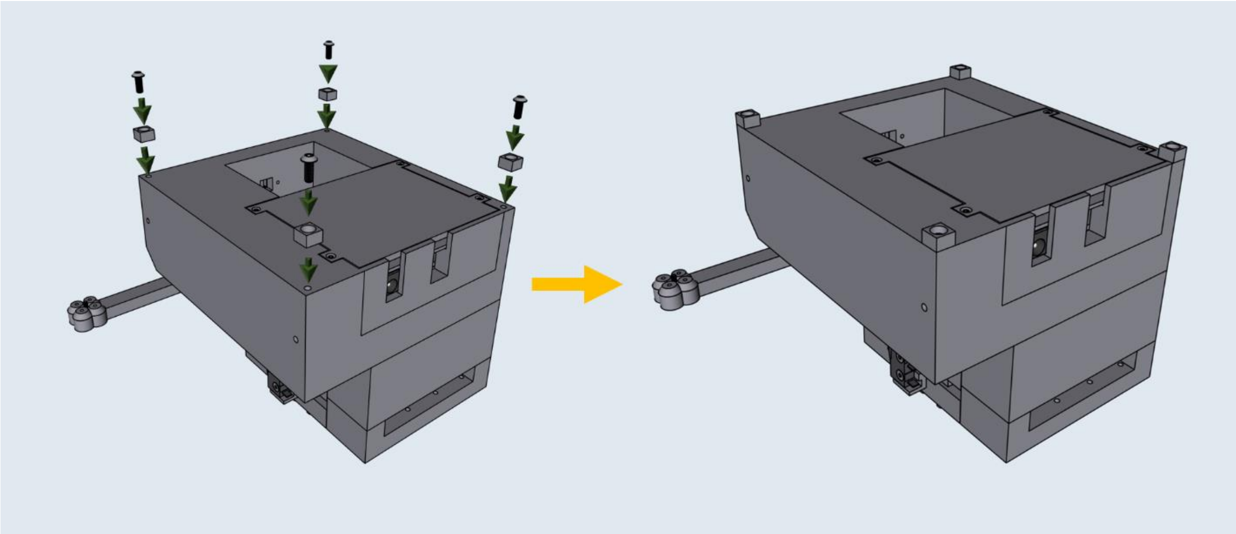


Figure 48: M3 x 8 screws, Foot (1-4), Base Assembly

Attach four Feet to the Base Assembly with four M3 x 8 screws (Fig. 48).

Step 2: Attach Feet to the Plate Holder

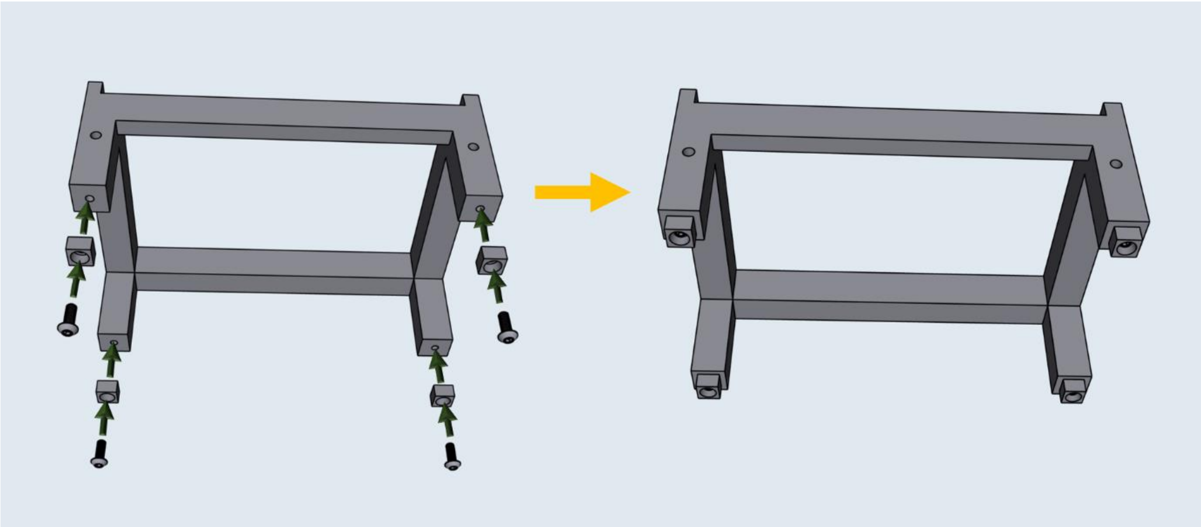


Figure 49: M3 x 8 screws, Foot(5-8), Plate Holder

Attach four Feet to the 96 Well Plate Holder with four M3 x 8 screws (Fig. 48).

Step 3: Attach Plate Holder to Base Assembly

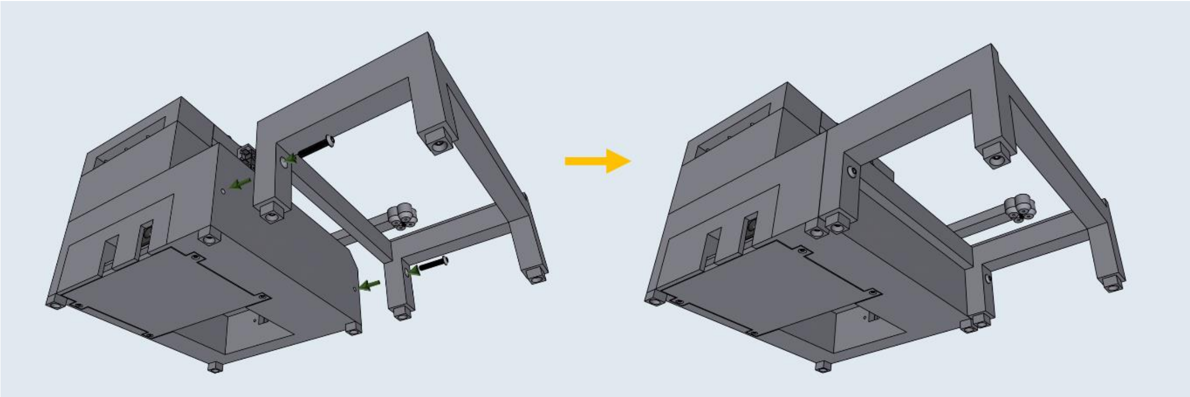


Figure 50: M3 x 16 screws, 96 Well Plate Holder, Base Assembly

Fasten the 96 Well Plate Holder to the Base assembly using two M3 x 16 screws.

5I: Assembling the Pump Tubing

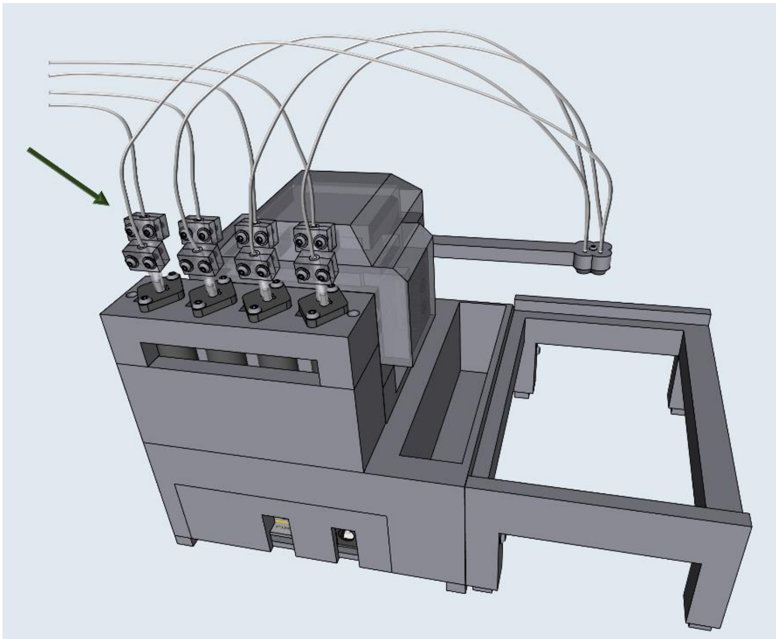


Figure 51: The Sidekick Tubing

Step 1: Assemble the Tubing Adapter Clamp

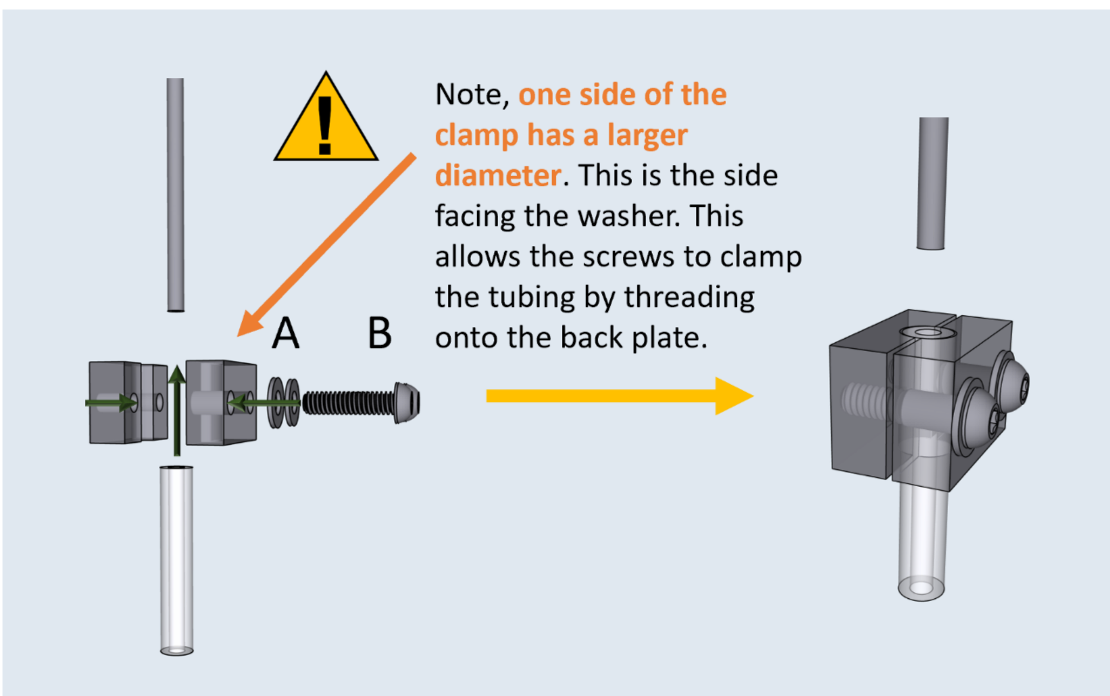


Figure 52: A) M2 Washer, B) M2 x 8 screw, 1/8" OD PTFE Tubing, Adapter Clamp Front, Adapter Clamp Rear

Sandwich the tip of the prepared segment of 1/8" tubing between the Front Adapter Clamp, and the Rear Adapter Clamp. You can tell the two clamps apart by the larger hole diameter of the Front Adapter Clamp.

Pass the M2 x 8 screws through the M2 x 8 washers, then through the Front Adapter Clamp and screw it into the Rear Adapter Clamp. Loosely tighten the screws (Fig. 52).

Step 2: Sleeving the 1/16" PTFE Tubing

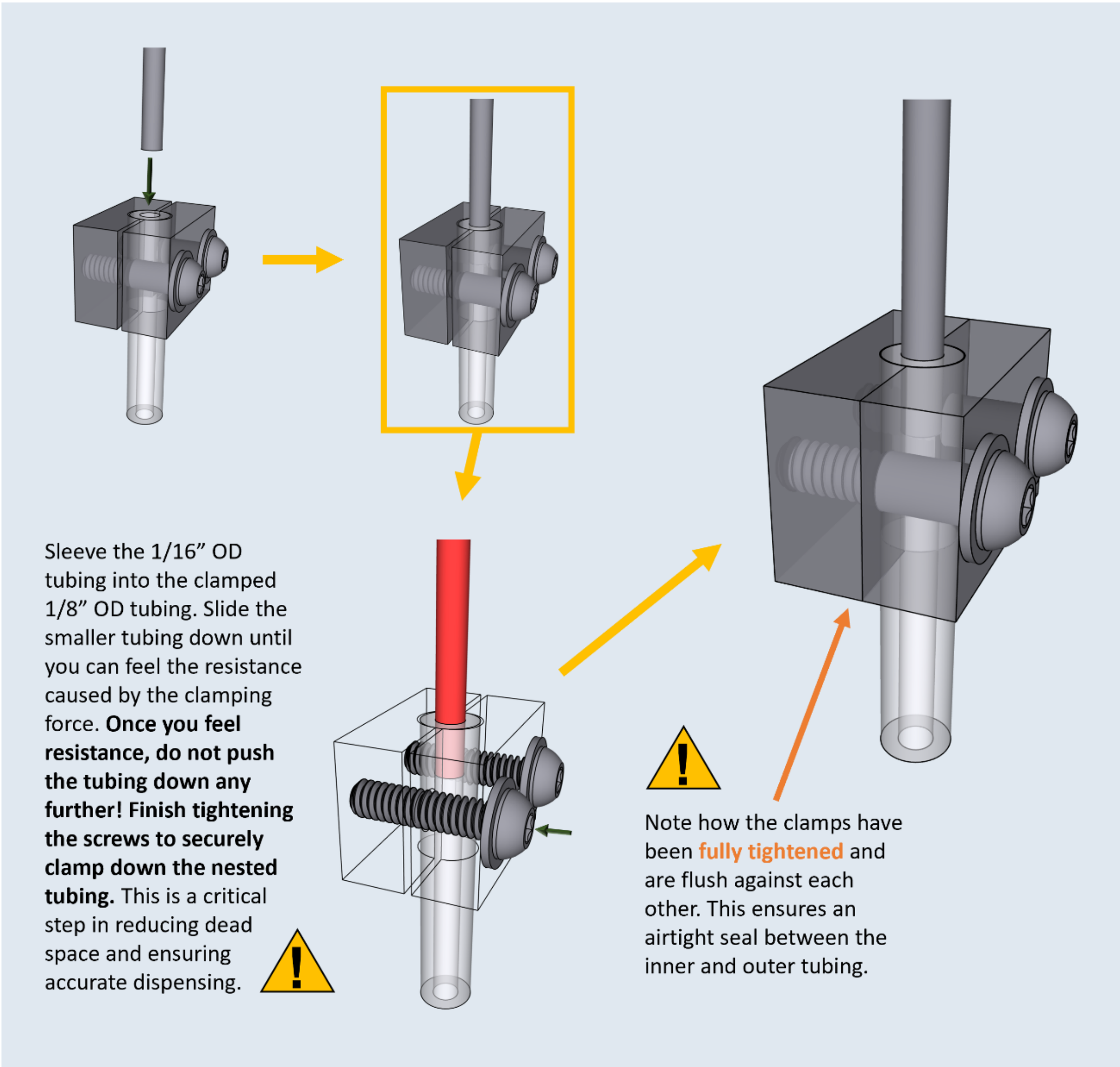


Figure 53: Adapter Clamp sleeving protocol.

Insert the 1/16" OD PTFE Tubing into the clamped 1/8" Tubing, then fully tighten the M2 screws. Closely follow the directions in **Figure 53**, as this is an important step for accurate liquid dispensing.

Step 3: Repeat for Inlet and Outlet Adapters

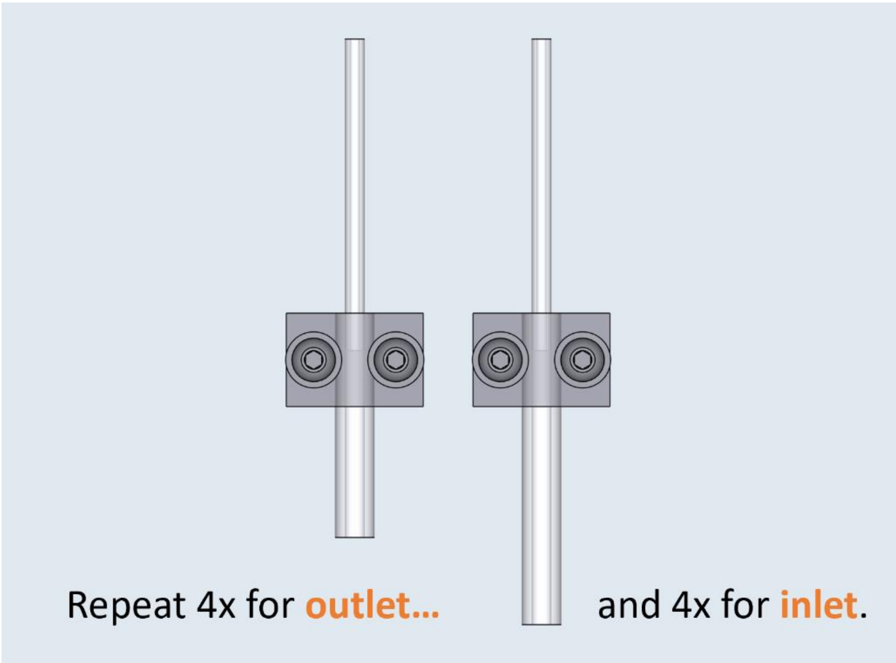


Figure 54: Inlet and outlet adapter clamps

Repeat Steps 1-2 four times for the outlet adapter tubing (18 mm long), and four times for the inlet adapter tubing (25 mm long).

Step 4: Attaching the tubing to the Pumps

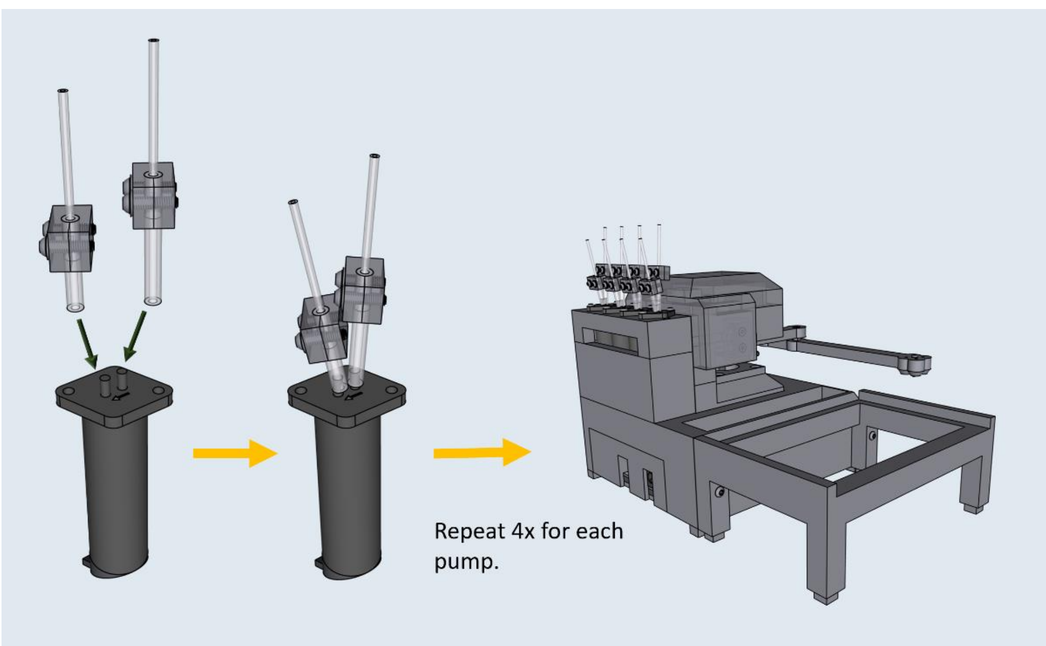


Figure 55: Inlet Tubing Assembly, Outlet Tubing Assembly, LPM Dispense Pump

Press fit the Tubing Assemblies over the inlet and outlet ports of the pump. The pump has an arrow by the ports that indicates the direction of liquid travel. Place the Inlet Tubing Clamp (longer tubing) over the inlet port, place the Outlet Tubing Clamp (shorter tubing) over the outlet port (Fig. 55). Repeat four times for each pump.

Step 5: Connect the outlet tubing into the Armature

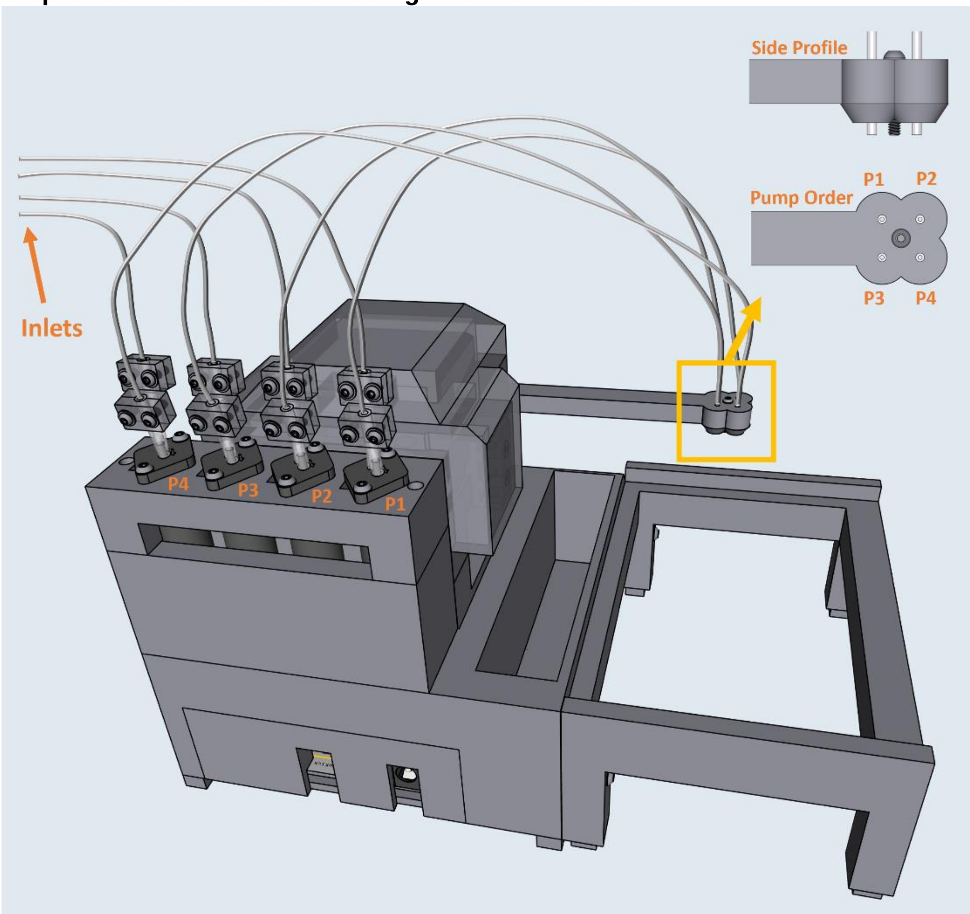


Figure 56: Tubing diagram, connecting each pump to the proper nozzle location

Connect the outlet tubing of each pump into the end effector. Reference **Figure 56** to match each pump to their end effector position. Push each tubing end out to the length of the center screw.

With this step, the mechanical assembly of the Sidekick is complete.

6. Operation instructions

Step 1: Determining the COM port

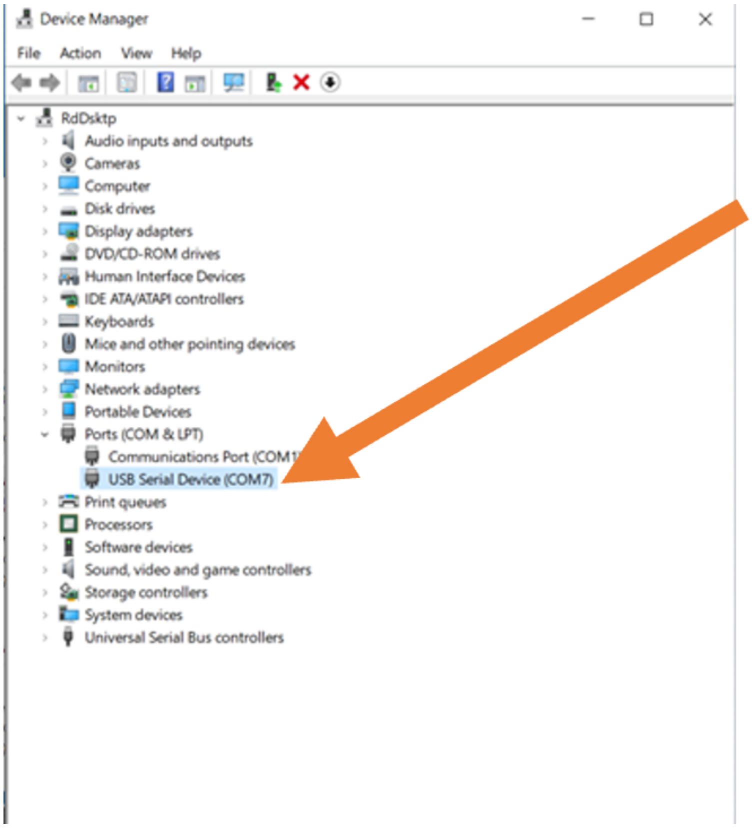
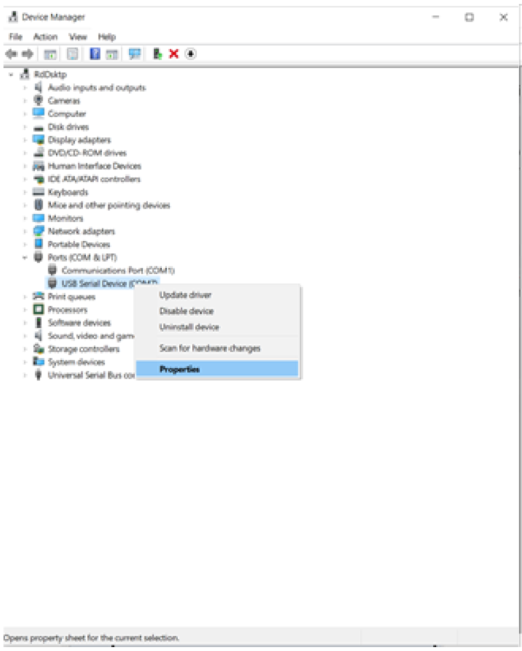


Figure 57: Checking the Sidekick COM port in Windows 10 device manager

After assembling the Sidekick and validating the correct wiring, reconnect the Sidekick to the USB port of the controlling computer, and connect it to the power supply. The Sidekick can be commanded via serial port. Any application that supports outputting via the serial port can communicate with liquid handler. We recommend using Thonny or a serial terminal program such as Putty (on Windows), or the built-in “screen” command line program (on Mac/Linux). If using Thonny, skip to Step 5.

To use Putty, first find the SideKick’s com port, and baud rate. These are easy to find in Windows with the device manager. Go to Device Manager -> Ports -> USB Serial Device (Fig. 57). Our Sidekick is located at COM7 (on Windows) or /dev/tty.usbmodem0000000000001 (on Mac/Linux). If you are unsure which port your Sidekick occupies, unplug it, and see which port disappears.

Step 2: Determining Baud



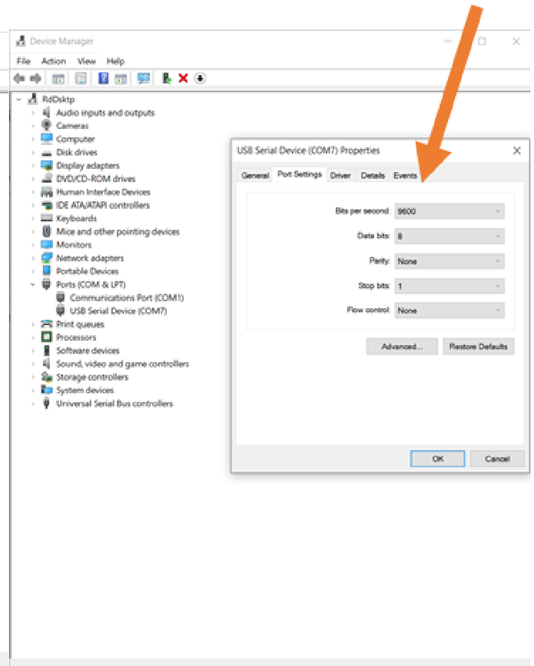


Figure 57a: Device Manager drop down

Figure 57b: Sidekick Baud in Device Manager Properties

Right click on the USB Serial Device. Click “Properties” in the drop down (Fig. 57a). Then click port settings. Make a note of the indicated bits per second, or baud (Fig. 57b).

Step 3: Connecting Via Putty

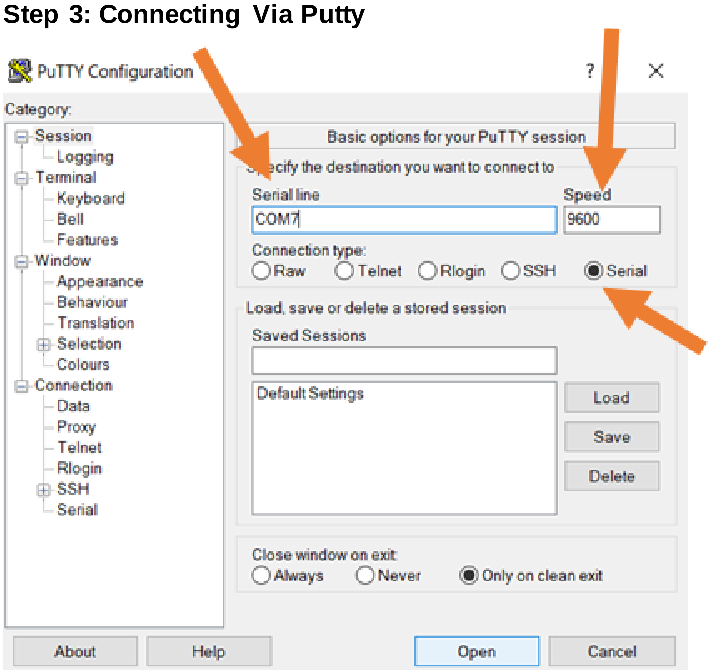


Figure 58: Putty startup UI

Now that we have the com port and the baud, we can use Putty to open a serial connection. Download and install Putty: https://www.putty.org/

Once Putty is installed, open the application. Hit the “Serial” option, and then input the correct com port we found in Step 1 under “Serial line”. Then input the baud rate we found in Step 3 under “Speed”.

Step 4: Opening Putty and Sending Commands

Figure 59a: Putty terminal

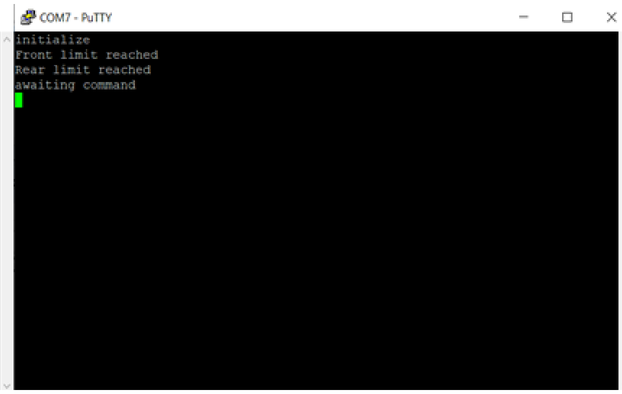


Figure 59b: Sending the Sidekick an “initialize” command

After hitting open, the command line screen will appear (Fig. 59a). Type “initialize” (followed by return) to test communication with the SideKick (Fig. 59b). The SideKick should home against the limit switches, and output to the terminal. The SideKick is now online and prepared to receive commands.

Step 5: Calibrating the Plate Map, and setting the Purge Location

Before doing anything with the SideKick, first set up the plate map and a purge location. Setting up a plate map is important for locational accuracy. Slight variations in assembly may invalidate the SideKick’s preloaded map for a standard 96-well plate. To calibrate the SideKick, load a 96-well SBS microplate into the tray. Then type the command: “remap” (return) and follow the prompts. You can remap as many times as you wish.

The purge location is any arbitrary place that you can use to purge the pump lines and dump excess liquid. The SideKick has been designed with a tray to accommodate small waste vials, but you can easily set up a purge location elsewhere. To set up a purge location, type the command: “set purge” (return) then follow the prompts. The purge location can always be reset.

Once the purge location has been set, and the plate map recalibrated, your SideKick is now ready to dispense to a plate. Before dispensing from a pump, clear the air from the lines. This can be done with the “manual purge” command. Repeat this process after leaving liquid in the lines for an extended period, or to clean the lines after changing liquids.

6.2 Command Library:

Dispensing and Moving: A dispense or move command should be a single line indicating four things:

Desired Pump: Ex. “Pump 1”
Desired Action: “dispense” or “move”
Target Location: Ex. “Well A6” or “purge”
Target Volume (if dispensing): Ex. “200 Microliters”

This is achieved by the following command:

- “Pump 1 dispense 200 microliters into well A3”

The order does not matter, so long as you indicate these four things. The following commands all mean the same thing.

- “Dispense 200 microliters from Pump 1 into A3”
- “dispense 200 microliters from pump 1 into a3”
- “Dispense 200 Pump 1 well A3”
- “Pump 1 Dispense 200 well A3”

If moving to a well, no volume needs to be indicated, for example:

- Pump 1 move well A3

Things to note:

- Make sure to type “well” if targeting a well. Ex. “well a6” not just “a6”.
- Case does not matter.
- Prepositions do not matter.
- Order does not matter.
- Volume is given in microliters and rounded to the nearest 10 uL increment.

Command List:

Initialize: Homes the armature against the limit switches. Use if the SideKick has bumped against something and skipped steps.
Hardware check: Runs through all the hardware to validate that everything has been wired correctly. Use after wiring the SideKick, or for troubleshooting.
Free move: Allows the user to freely move the armature.
Sleep: De-energizes the motors, allowing the armature to move freely.
Wake: Re-energizes the motors so that the armature can move again. Use after the “sleep” command.
Return home: Returns the armature to the home location.
Manual purge: Used to purge the liquid lines. Use after swapping reagents, or for cleaning the lines.
Remap: Used to calibrate a new plate. Use if swapping in a plate with different well locations.
Set purge: Sets the purge location. Use if changing the location of the purge tray/vial.

7. Validation and characterization

*** Work in progress***

Sidekick's accuracy and precision was tested through both gravimetric and absorbance analysis.

In a simple gravimetric test of Sidekick's multichannel accuracy, 10 microliters of water (the lowest volume the pumps are rated for) was pipetted by each pump, and the resulting mass was recorded. The Lee Company pumps had high precision across all channels. The preliminary data for each of the pumps 1-3 are given below.

However, we can see accuracy in a single cycle was found to vary +- 20%, larger than the manufacturers rating of +- 15%. While some pumps delivered the desired 10 microliters, others delivered closer to 12 microliters. This is likely due to variances in the bore of the solenoid pumps.

To accommodate these variances in pump accuracy, we can leverage their high precision to computationally adjust the number of cycles necessary to reach the desired volume. When implementing this calibration, accuracy for larger dispense volumes were within 6 microliters of target. Given the high repeatability of the Lee Company micropumps, this will ensure that with calibration all volumes pipetted will be within 6 microliters of target. An example of this data can be seen below, as the accuracy of each channel is evaluated at 10, 50, 100, and 500 microliters

As a more comprehensive examination of Sidekick's performance over 96 well plate experiments, an absorbance test was run using colored dye. The Sidekick is programmed to plate a gradient of mixed red and green dyed water, ranging from 100% concentration of red dye to 100% concentration green dye. The absorbances are recorded in a plate reader at 420 and 650 nanometers.

Ethics statements

Nothing to report

CRediT author statement

Rodolfo Keesey: Investigation, Software, Visualization, Investigation, Writing – Original Draft

Joshua Schrier: Conceptualization, Supervision, Project Administration, Funding Acquisition, Writing – Review and Editing

Acknowledgments

This study is based upon work supported by the Henry Dreyfus Teacher-Scholar Award (TH-14-010) and the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001118C0036. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA. JS acknowledges the and resources of the MERCURY consortium (http://mercuryconsortium.org/) under NSF Grant No. CNS-2018427.

References:

[1] K. Olsen, “The First 110 Years of Laboratory Automation: Technologies, Applications, and the Creative Scientist,” *J. Lab. Autom.*, vol. 17, no. 6, pp. 469–480, Dec. 2012, doi: 10.1177/2211068212455631.

[2] M. M. Flores-Leonar *et al.*, “Materials Acceleration Platforms: On the way to autonomous experimentation,” *Curr. Opin. Green Sustain. Chem.*, vol. 25, p. 100370, Oct. 2020, doi: 10.1016/j.cogsc.2020.100370.

[3] E. Stach *et al.*, “Autonomous experimentation systems for materials development: A community perspective,” *Matter*, vol. 4, no. 9, Art. no. 9, Sep. 2021.

[4] J. Boyd, “Robotic Laboratory Automation,” *Science*, vol. 295, no. 5554, pp. 517–518, Jan. 2002, doi: 10.1126/science.295.5554.517.

[5] R. S. Markin and S. A. Whalen, “Laboratory Automation: Trajectory, Technology, and Tactics,” *Clin. Chem.*, vol. 46, no. 5, pp. 764–771, May 2000, doi: 10.1093/clinchem/46.5.764.

[6] “Opentrons | Open-source Lab Automation, starting at $5,000.” https://opentrons.com/ (accessed Nov. 30, 2021).

[7] D. C. Florian, M. Odziomek, C. L. Ock, H. Chen, and S. A. Guelcher, “Principles of computer-controlled linear motion applied to an open-source affordable liquid handler for automated micropipetting,” *Sci. Rep.*, vol. 10, no. 1, p. 13663, Aug. 2020, doi: 10.1038/s41598-020-70465-5.

[8] F. Barthels, U. Barthels, M. Schwickert, and T. Schirmeister, “FINDUS: An Open-Source 3D Printable Liquid-Handling Workstation for Laboratory Automation in Life Sciences,” *SLAS Technol. Transl. Life Sci. Innov.*, vol. 25, no. 2, pp. 190–199, Apr. 2020, doi: 10.1177/2472630319877374.

[9] P. Dettinger *et al.*, “Open-source personal pipetting robots with live-cell incubation and microscopy compatibility,” Jul. 2021. doi: 10.1101/2021.07.04.448641.

[10] “P200 Pipette, Single Channel, 20-200µl | BT Lab Systems,” *BenchTop Lab Systems*. https://www.btlabsystems.com/P200_Pipette (accessed Nov. 02, 2021).

[11] “PIPETMAN Classic P200.” https://www.gilson.com/default/pipetman-classic-p200.html (accessed Nov. 02, 2021).

[12] M. C. Carvalho and R. H. Murray, “Osmar, the open-source microsyringe autosampler,” *HardwareX*, vol. 3, pp. 10–38, Apr. 2018, doi: 10.1016/j.ohx.2018.01.001.

[13] A. Faiña, B. Nejati, and K. Stoy, “EvoBot: An Open-Source, Modular, Liquid Handling Robot for Scientific Experiments,” *Appl. Sci.*, vol. 10, no. 3, Art. no. 3, Jan. 2020, doi: 10.3390/app10030814.

[14] S. Baas and V. Saggiomo, “Ender3 3D printer kit transformed into open, programmable syringe pump set,” *HardwareX*, vol. 10, Oct. 2021, doi: 10.1016/j.ohx.2021.e00219.

[15] A. S. Booeshaghi, E. da V. Beltrame, D. Bannon, J. Gehring, and L. Pachter, “Principles of open source bioinstrumentation applied to the poseidon syringe pump system,” *Sci. Rep.*, vol. 9, p. 12385, Aug. 2019, doi: 10.1038/s41598-019-48815-9.

[16] A. Jönsson, A. Toppi, and M. Dufva, “The FAST Pump, a low-cost, easy to fabricate, SLA-3D-printed peristaltic pump for multi-channel systems in any lab,” *HardwareX*, vol. 8, Oct. 2020, doi: 10.1016/j.ohx.2020.e00115.

[17] “MicroPython - Python for microcontrollers.” http://micropython.org/ (accessed Nov. 30, 2021).

[18] “Line-us,” *Line-us*. https://www.line-us.com/ (accessed Nov. 02, 2021).

[19] BitSand, “Tracey - Drawing Machine,” *Instructables*. https://www.instructables.com/Tracey-Drawing-Machine/ (accessed Nov. 02, 2021).

[20] “MANTIS - Liquid Handler,” *FORMULATRIX®*. https://formulatrix.com/liquid-handling-systems/mantis-liquid-handler/ (accessed Nov. 02, 2021).

[21] G. Wu and H. Shen, “Design Optimization of Parallel PnP Robots,” in *Parallel PnP Robots: Parametric Modeling, Performance Evaluation and Design Optimization*, G. Wu and H. Shen, Eds. Singapore: Springer, 2021, pp. 191–220. doi: 10.1007/978-981-15-6671-4_8.

[22] “OpenCM9.04-C (with onboard XL-type connectors) - ROBOTIS.” https://www.robotis.us/opencm9-04-c-with-onboard-xl-type-connectors/ (accessed Nov. 30, 2021).

[23] M. Eaker and D. Mitra, “How to Reduce Audible Noise in Stepper Motors,” Texas Instruments, Application Report SLVAES8, May 2020. Accessed: Nov. 30, 2021. [Online]. Available: https://www.ti.com/lit/an/slvaes8/slvaes8.pdf?ts=1630503659521

[24] T. Ching *et al.*, “Highly-customizable 3D-printed peristaltic pump kit,” *HardwareX*, vol. 10, p. e00202, Oct. 2021, doi: 10.1016/j.ohx.2021.e00202.

[25] M. R. Behrens *et al.*, “Open-source, 3D-printed Peristaltic Pumps for Small Volume Point-of-Care Liquid Handling,” *Sci. Rep.*, vol. 10, no. 1, p. 1543, Jan. 2020, doi: 10.1038/s41598-020-58246-6.

[26] M. S. Cubberley and W. A. Hess, “An Inexpensive Programmable Dual-Syringe Pump for the Chemistry Laboratory,” *J. Chem. Educ.*, vol. 94, no. 1, pp. 72–74, Jan. 2017, doi: 10.1021/acs.jchemed.6b00598.

[27] “LPM Series Pumps,” *The Lee Company*. https://www.theleeco.com/products/electro-fluidic-systems/dispense-pumps/fixed-volume-dispense-pumps/lpm-series-pumps/ (accessed Nov. 02, 2021).

[28] “LPM Dispense Pump Data Sheet,” *The Lee Company*. https://www.theleeco.com/resources/lpm-dispense-pump-data-sheet/ (accessed Nov. 02, 2021).

[29] “20 Gauge2 inPTFE Disposable Luer Lock Needle For Use With Syringes and Dispensing Machines,” *Grainger*. https://www.grainger.com/product/GRAINGER-APPROVED-20-Gauge2-inPTFE-Disposable-5FVL4 (accessed Nov. 30, 2021).

[30] “Fritzing.” http://fritzing.org/ (accessed Nov. 30, 2021).