Step-by-Step Guide to Building a Linear Translation Stage

Bill of Materials

The parts to build the linear stage came mainly from two suppliers: TechnobotsOnline.com and Adafruit.com. Because many parts are just standard fasteners, you may already have many of the screws, studding and nuts as well as some of the electronics lying around. Many of the components are also cheaply available on Amazon and Ebay. Included in the BoM below are links to product pages for each item. Note links last updated 29/08/13.

Here a few useful links to some of the suppliers we used.

Name	Info	Country
Technobots	Supplies an excellent range of mechanical and electrical components, particularly for robotics and CNC projects. UK based.	UK
Adafruit	A fantastic US website which produces educational electronics kits backed up with superb tutorials and resources	USA
Wickes	DIY and hardware store	UK
ThePlasticPeople	Offer cut-to-size engineering plastics including acrylic	UK
Hobbytronics	Supplies electronic components for robotics	UK
Phenoptix	UK distributor of many Adafruit products	UK

Item	Qty	Supplier
OpenBeam		
OpenBeam 300mm Al extrusion	4	Technobots
OpenBeam 150mm Al extrusion	12	Technobots
OpenBeam 60mm Al extrusion	2	Technobots
OpenBeam T-Bracket	12	Technobots
OpenBeam L-Bracket	8	Technobots
OpenBeam Shaft Clamps	4	Technobots
OpenBeam NEMA 17 Stepper Motor Mount	1	Technobots
OpenBeam Feet	4	Technobots
Mechanical Components		
Adafruit 12V Stepper Motor 200 steps/rev. (NEMA 17)	1	Adafruit
Aluminium Flex Shaft Coupler	1	Adafruit
LM8UU linear bearing	2	Adafruit
8mm Silver steel rod 333mm	2	Technobots
M8 270mm threaded rod (studding)	1	Wickes
M8 120mm threaded rod (studding)	1	Wickes
M6 150mm threaded rod (studding)	4	Technobots
Acrylic sheet for stage 100 x 140 x 5mm	1	ThePlasticPeople
Fasteners		
M2 12mm caphead screws	4	Technobots
M3 6mm buttonhead screws (pack of 50)	126	Technobots
M3 8mm caphead screws (pack of 10)	4	Technobots
M3 20mm caphead screws (pack of 10)	4	Technobots
M3 hex nuts (pack of 100)	142	Technobots
M2 hex nuts	4	Technobots
M3 washers (pack of 100)	24	Technobots
M6 hex nuts	14	Technobots
M8 hex nuts	3	Wickes
M2 washers	4	Technobots
M6 washers	14	Technobots
M8 washers	2	Wickes
Electronics		
Arduino Uno R3	1	Adafruit
Adafruit V2 Motorshield	1	Adafruit
Standard A-B USB cable	1	Adafruit
Set of stacking headers	2	Technobots
14mm lever microswitches V4	2	Technobots
Tactile switch 6x6mm	2	Technobots
Small breadboard	1	Adafruit
Electrical wire	-	Technobots

Tools	
2mm hex allen key (for M3 buttonhead screws)	Technobots
2.5mm hex allen key (for M3 caphead screws)	Technobots
Soldering iron	Technobots
Hacksaw	Technobots
MakerBot Replicator 2X	MakerBot
1.75mm ABS filament for 3D printer	MakerBot

Step 1 – Print 3D printed parts

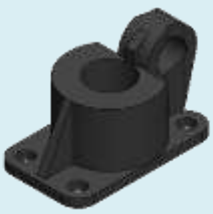
When building our prototype microscope, we were fortunate enough to have access to a MakerBot Replicator 2X 3D printer. This was a fantastic resource as it enabled us to create complex and bespoke parts very quickly. Most desktop printers can use either ABS or PVC plastic, both of which are stiff and versatile materials.

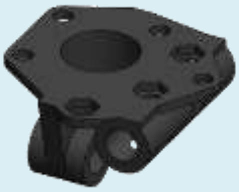
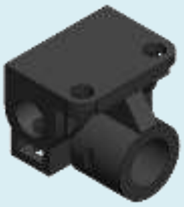
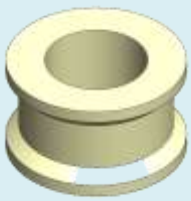
If you don't have access to a 3D printer, you can send your designs to a 3D printer company who will manufacture your parts for as little as 50p per item. Some online companies which offer this service include:

https://www.ponoko.com/
http://i.materialise.com/
http://www.shapeways.com/
http://www.sculpteo.com/en/

The .STL files for all the 3D printed parts can be found on the OpenLab Tools Github repository here.

3D printing is quite a slow process and some of these parts may take up to an hour to print, so move on to the next section whilst your 3D parts are printing!

	**2 x Endstop Mounts**	Supports: Print time: Volume:	No 11mins 1157mm^3
	**Optics Clamp (Optional)**	Supports: Print time: Volume:	No 1hr 7mins 6921mm^3
	**Lead Nut Mount - Top**	Supports: Print time: Volume:	Yes 41mins 4183mm^3

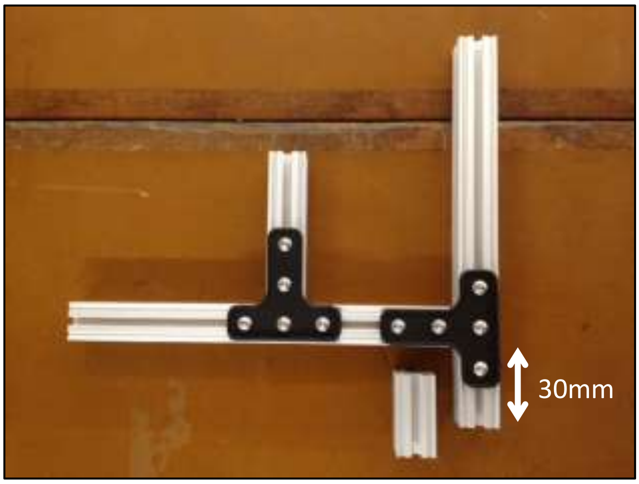
	**Lead Nut Mount - Bottom**	Supports: Print time: Volume:	Yes 46mins 4694mm^3
	**Bearing Mount Left - Top**	Supports: Print time: Volume:	Yes 50mins 5169mm^3
	**Bearing Mount Left - Bottom**	Supports: Print time: Volume:	Yes 1hr 7mins 6924mm^3
	**Bearing Mount Right - Top**	Supports: Print time: Volume:	Yes 50mins 5169mm^3
	**Bearing Mount Right - Bottom**	Supports: Print time: Volume:	Yes 1hr 7mins 6924mm^3
	**2 x Front Stage Mounts**	Supports: Print time: Volume:	Yes 44mins 4548mm^3
	**4 x M3 Spacer**	Supports: Print time: Volume:	No 2mins 78mm^3

Step 2 – Base and stepper motor mount

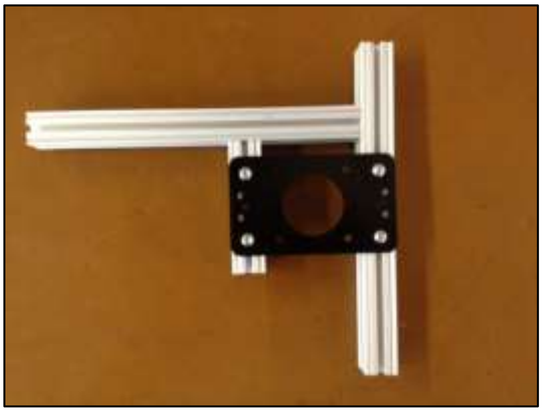
Begin by putting 6mm buttonhead screws through each of the holes in 2 of the OpenBeam T-brackets, and attach nuts to the other side leaving 2 or 3mm between the nut and the bracket to allow the nuts to slide into the Al extrusion. If you are short of screws it is sufficient to omit the central screw, so long as each beam section is connected to the bracket by at least 2 screws.

Slide the top of the 'T' into one of the 150mm aluminium sections, and the bottom into the 60mm section.

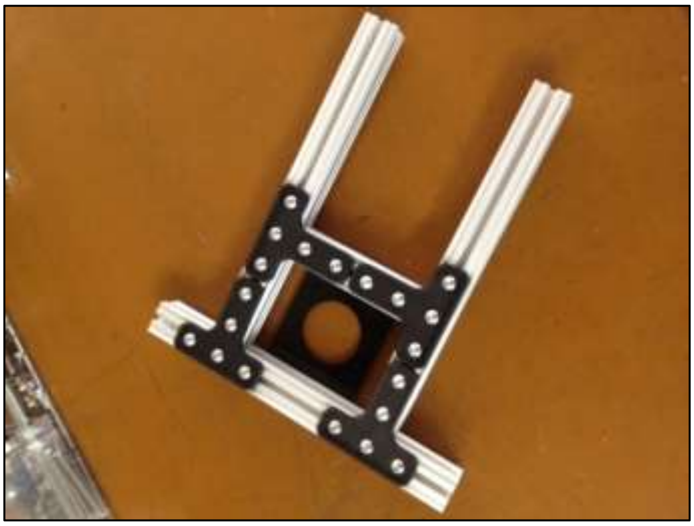
Tighten up the bolts on the 60mm section but for now leave the bolts at the top of the ‘T’ loose – this will make it easier to attach the motor mount.



Attach a second 150mm section perpendicular to the first with a T-bracket. There should be a 30mm gap between the end of the new piece and the side of the original, as shown. If you got hold of an OpenBeam starter kit, you can use the 30mm pieces to measure.



Now flip the assembly over. Insert screws into the four outermost holes of the stepper motor mount, and loosely attach screws on the underside. Slide the mount plate onto the parallel 60mm and 150mm sections, such that the mount is centred on the 60mm section.



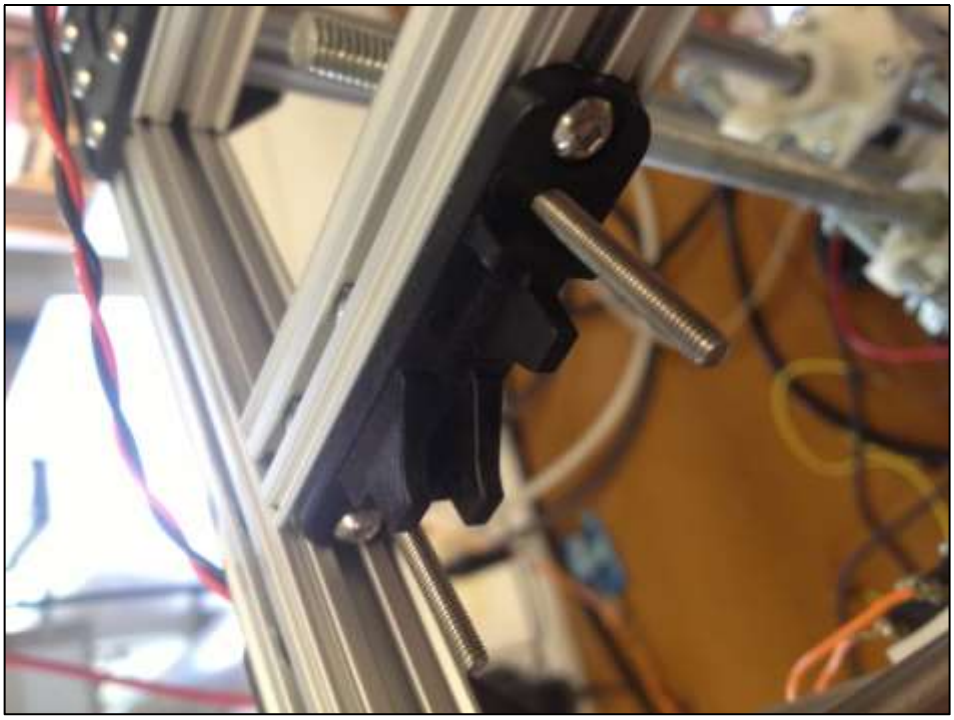
Flip the assembly back over onto the side with the brackets, and close off the open side with another 150mm section. Check that the gap between the side of the new piece and the end of the perpendicular piece is 30mm.

In this step

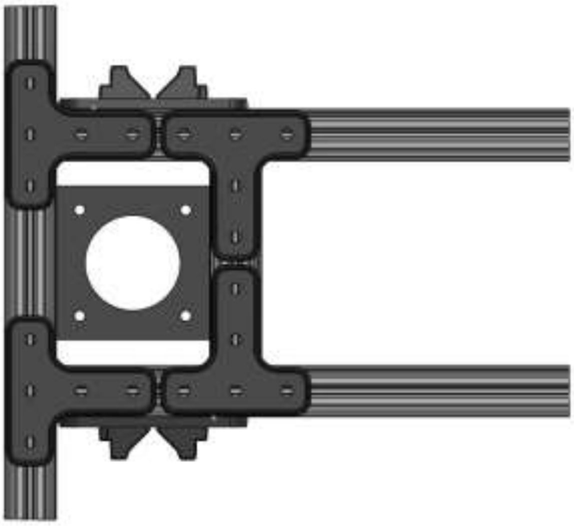
3	OpenBeam 150mm aluminium extrusion
1	OpenBeam 60mm aluminium extrusion
4	OpenBeam T-Brackets
1	OpenBeam NEMA 17 stepper motor mount
24	M3 6mm buttonhead screws
24	M3 hex nuts

Step 3 – Shaft clamps

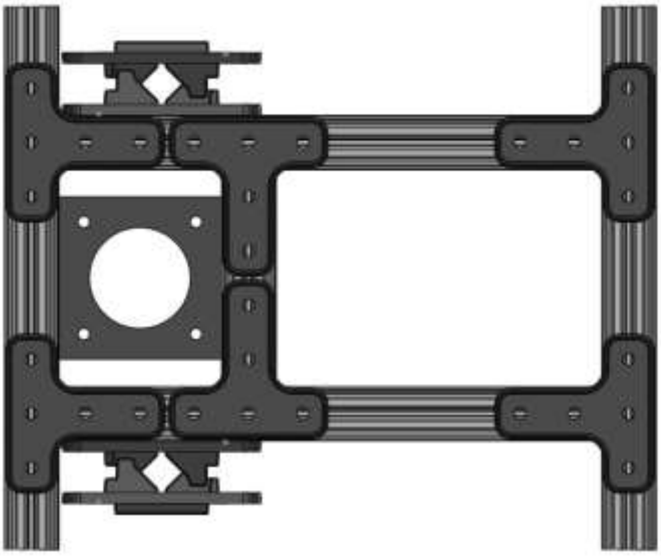
The OpenBeam shaft clamps consist of two identical pieces which mesh together. If you bought a shaft clamp set from Technobots this includes 4 of these pieces, enough to build 2 shaft clamps. The sets also contain two different types of screw: 25mm and 35mm M3 hex-head screws. We will be using the 25mm screws.



Into the two outermost holes, insert 6mm screws and nuts, with the nuts on the opposite side to the triangular teeth. Now from the back of the half-clamp, insert the 25mm screws into the inner holes, such that the hexagonal heads line up with the hex nuts.



Slide one half-clamp onto each of the 150mm beams, pushing them right to the end of the section so that they are just touching the section at the back. Once in place, tighten the buttonhead screws, and slot the other half of each clamp onto the 25mm screw. Secure with a M3 washer and bolt, but leave the clamp loose.



Finally the base can be completed by closing off the base with a final 150mm Al section. As before, the base must be completely symmetrical so ensure a 30mm gap on either side.

In this step

1	OpenBeam 150mm aluminium extrusion
4	OpenBeam half shaft clamps
4	M3 hex-head 25mm screws
2	OpenBeam T-Brackets
14	M3 6mm buttonhead screws
18	M3 hex nuts
4	M3 washers

Step 4 – Attach feet

In this step the plastic OpenBeam feet are attached to the 300mm sections which will form the vertical sections of the structure. This step is optional – the plastic feet help prevent the aluminium from scratching the work surface, but you may wish to omit them or replace them with vibration-dampening rubber feet instead.

The feet attach to the 300mm Al section by tapping an M3 screw through to hole at the end of the section. This should ideally be done using an M3 tapping tool. Off the record, if you don't have one of these to hand, the aluminium is soft enough to screw a bolt directly into the metal, but be VERY careful not to damage the screw head or the screwdriver that you are using.

Once the holes have been tapped, screw the feet into place with a 6mm M3 screw.

In this step

4	OpenBeam 300mm aluminium extrusion
4	OpenBeam feet
4	M3 6mm buttonhead screws

Step 5 – Vertical columns and side pieces

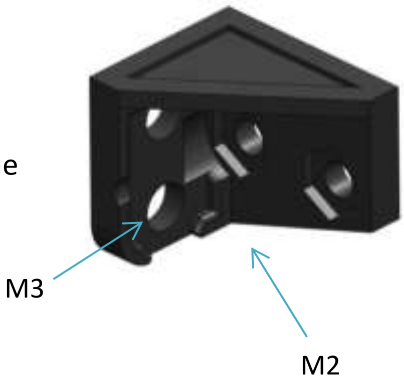
Attach the one of the 300mm sections with feet attached to each corner of your existing base structure. Ensure that the feet are on the opposite side to the motor mounting plate! The feet should protrude about 30mm below the bottom of the base frame to allow the motor to be connected underneath.

You can now complete the lower part of the assembly by connecting 150mm Al sections to the open sides of the base with L-brackets.

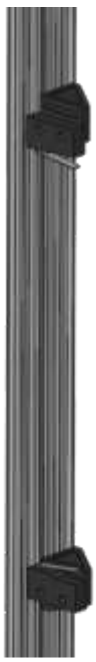
In this step

2	OpenBeam 150mm aluminium extrusion
4	OpenBeam L-brackets
20	M3 6mm buttonhead screws
20	M3 hex nuts

Step 6 – Endstops



The endstops are microswitches which prevent the stage from moving outside its maximum range. I would recommend adding the endstops at this stage rather than with the rest of the electronic components as it can be quite fiddly to fit the microswitches.



The mount has two 4mm holes for connecting to the OpenBeam, and two 3mm holes for the M2 microswitch screws. The interior is possibly a little over-complicated, but the hex-holes make it easier to fit the M2 nuts.

First solder a long piece of wire onto the common (C) and normally closed (NC) pins on each microswitch. Then screw the microswitches onto the mounts, one facing upwards, the other downwards.

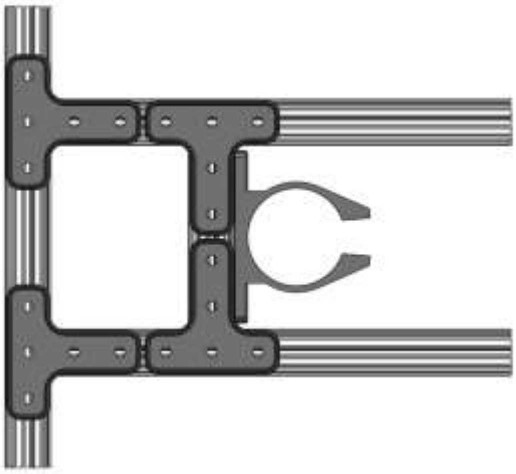
The enstops should then be attached to one of the back vertical OpenBeams in the usual fashion, with the switches protruding towards the front of the assembly. Position the endstops the required distance apart and tighten the M3 screws.

In this step

2	14mm lever microswitches V4
2	3D printed endstop mounts
2	M3 buttonhead screws 6mm
2	M3 hex nuts
2	M2 cheesehead screws 12mm
2	M2 hex nuts

Step 7 – Top frame and optics clamp

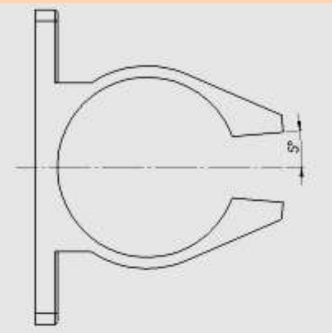
The top frame is effectively a duplicate of the base frame, with the motor plate omitted. Start by connecting the 60mm and 150mm at right angles as before. Add another 150mm section perpendicular to the first, again carefully measuring 30mm from the side of the first piece to the end of the second.



This time, instead of attaching the motor mount, secure the optics clamp using 4 6mm M3 screws. Proceed to attach the third 150mm section as before. For now it doesn't matter where exactly how far forward the 60mm section sits - we will adjust it later. Check that the optics clamp is centred and that the piece is nice and symmetrical, with 30mm gaps at either side.

Design Guide

The optics clamp was designed to hold a 25mm C-Mount tube The clamp needs to be both strong and flexible – a 2mm wall thickness was sufficient. Note that the tabs which the bolts fit into are angled back slightly (5° off centre). This means that in the clamped position the tabs are parallel so that the bolt sits flat. I added a hexagonal hole to the right hand tab which prevents the nut from turning in its socket and makes it easier to tighten the screw.



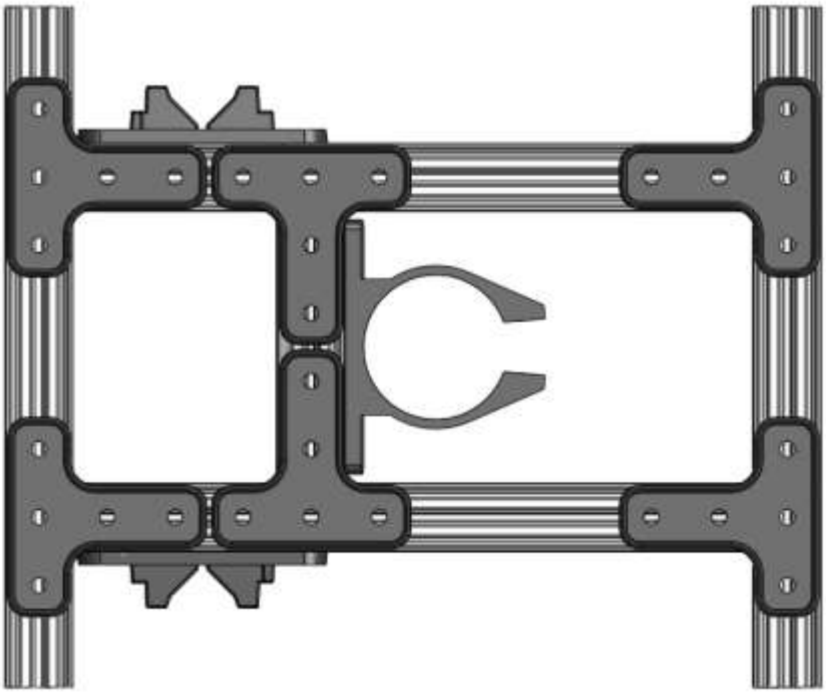
In this step

3	OpenBeam 150mm aluminium extrusion
1	OpenBeam 60mm aluminium extrusion
4	OpenBeam T-brackets
1	3D printed optics clamp
24	M3 6mm buttonhead screws
24	M3 hex nuts

Step 8 – Upper shaft clamps

Repeat the instructions from step 3, sliding one shaft clamp piece onto each side, so that it is touching the aluminium at the back of the frame. For this step we will leave the other half of the clamp off, so we can add the bearings to the shafts.

Close off the top frame with a 150mm section, measuring 30mm from either end.



You can now attach the top frame to the top of the vertical columns on the base structure using L-brackets.

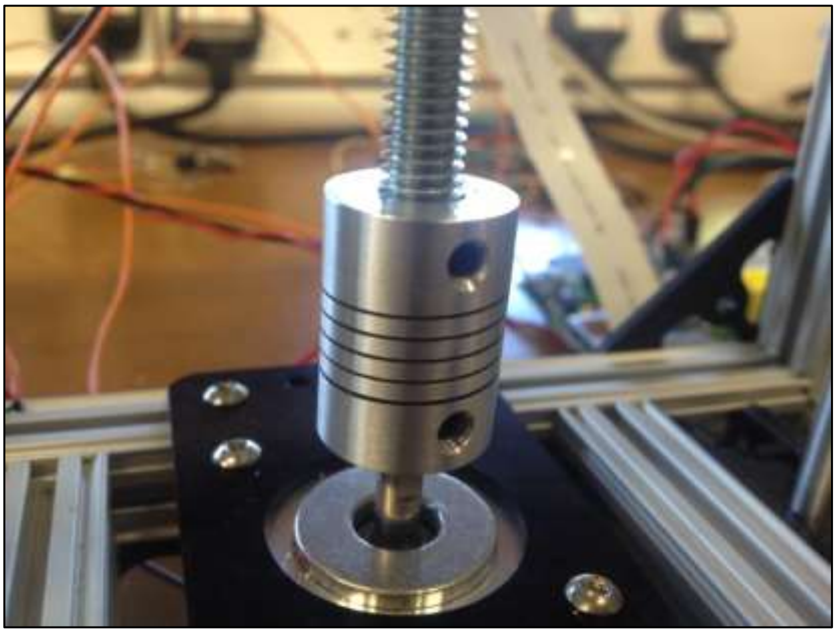
In this step

1	OpenBeam 150mm aluminium extrusion
2	OpenBeam half shaft clamps
4	M3 hex-head 25mm screws
2	OpenBeam T-Brackets
14	M3 6mm buttonhead screws
14	M3 hex nuts

Step 9 – Stepper motor, shaft coupler and lead-screw

Now that the support structure is almost complete, the mechanical drive system can be put in place. Before attaching the stepper, you may wish to extend the leads from the motor by soldering on additional lengths of wire.

Screw the motor into the underside of the stepper motor mount using the standard M3 6mm screws.



Insert the M8 threaded rod into the 8mm bore hole in the flex-shaft coupler, and tighten the grub screws using a small 2mm allen key. The 5mm bore end can then be inserted onto the shaft of the stepper motor and tightened.

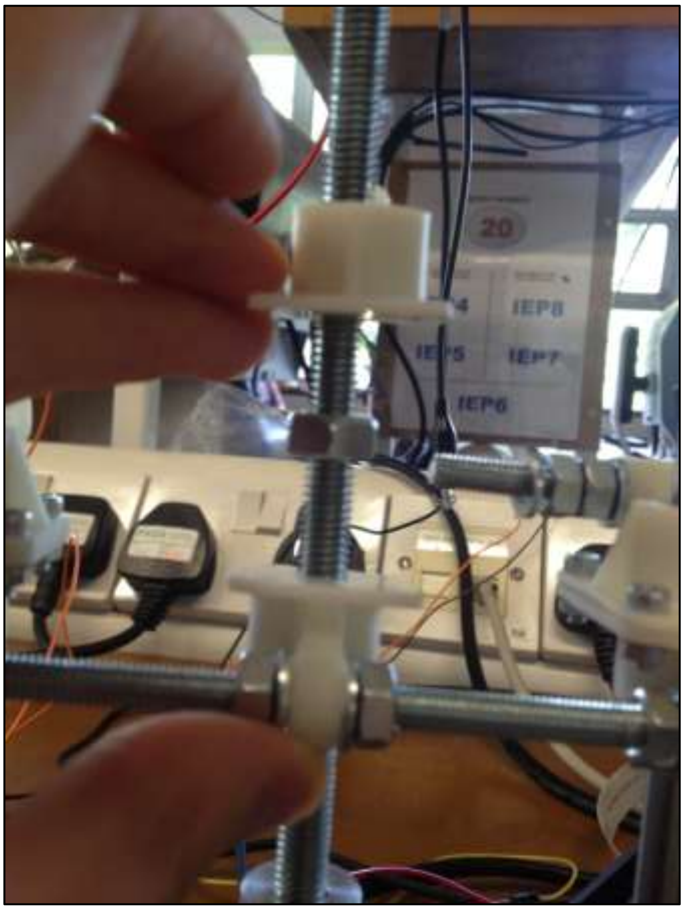
Design Guide
Note that the threaded rod is unconstrained at the top and will have quite a lot of freedom to move around relative to the motor shaft. This is intentional – the leadscrew will be kept approximately vertical by the leadnut. Adding a constraint on horizontal movement at the top would over-constrain the system and introduce unnecessary forces in the screw.

In this step

4	M3 6mm buttonhead screws
1	Adafruit 12V Stepper Motor 200 steps/rev. (NEMA 17)
1	Aluminium Flex shaft coupler
1	M8 300mm threaded rod

Step 10 – Lead Nut

The basic principle of a leadscrew is a rotating shaft with a stationary screw. The rotation of the screw causes the nut to move up and down the shaft. In the OpenLabTools stage, the rotation of the nut is constrained by a custom 3D mount with a hexagonal hole inside it.

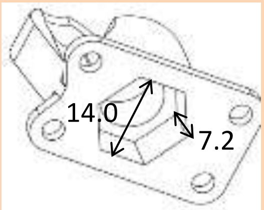


Put the bottom half of the 3D printed lead-nut mount onto the shaft first. The bottom is distinguished from the top by the round hole. Then screw an M8 nut onto the threaded rod. Add the top half, line up the holes and secure with M3 bolts.

Design Guide

The hex hole to fit the nut has the following dimensions

Width across flats	14mm
Width across corners	16.17mm
Thickness	7.2mm

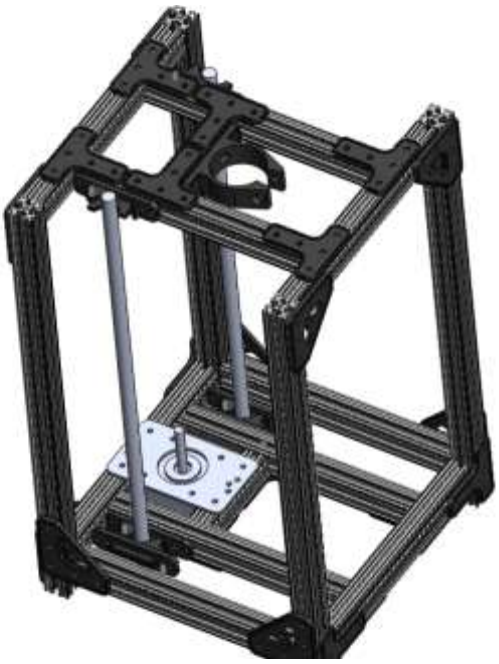


In this step

1	3D printed lead nut mount top
1	3D printed lead nut mount bottom
1	M8 hex nut
4	M3 8mm caphead screws
4	M3 hex nuts
4	M3 washers

Step 11 – Steel rods

The next components of the mechanical design are the smooth steel rods. These prevent the leadnut from rotating relative to the stage. The shafts used in the OpenLabTools microscope are 8mm hardened steel which allow the linear bearings to smoothly run up and down them.



Insert the steel rods into the bottom shaft clamps and tighten. Leave the top clamps open for now so that the bearings can be added.

In this step

2	8mm Silver steel rod 330mm

Step 12 – Bearings

The linear stage uses 2 LM8UU linear bearings, which fit inside custom 3D printed mounts. The main function of the bearings is to prevent the lead-nut from turning and to keep the stage level. The ‘8’ corresponds to the diameter of the shaft, so LM6UU would fit on a 6mm diameter shaft.

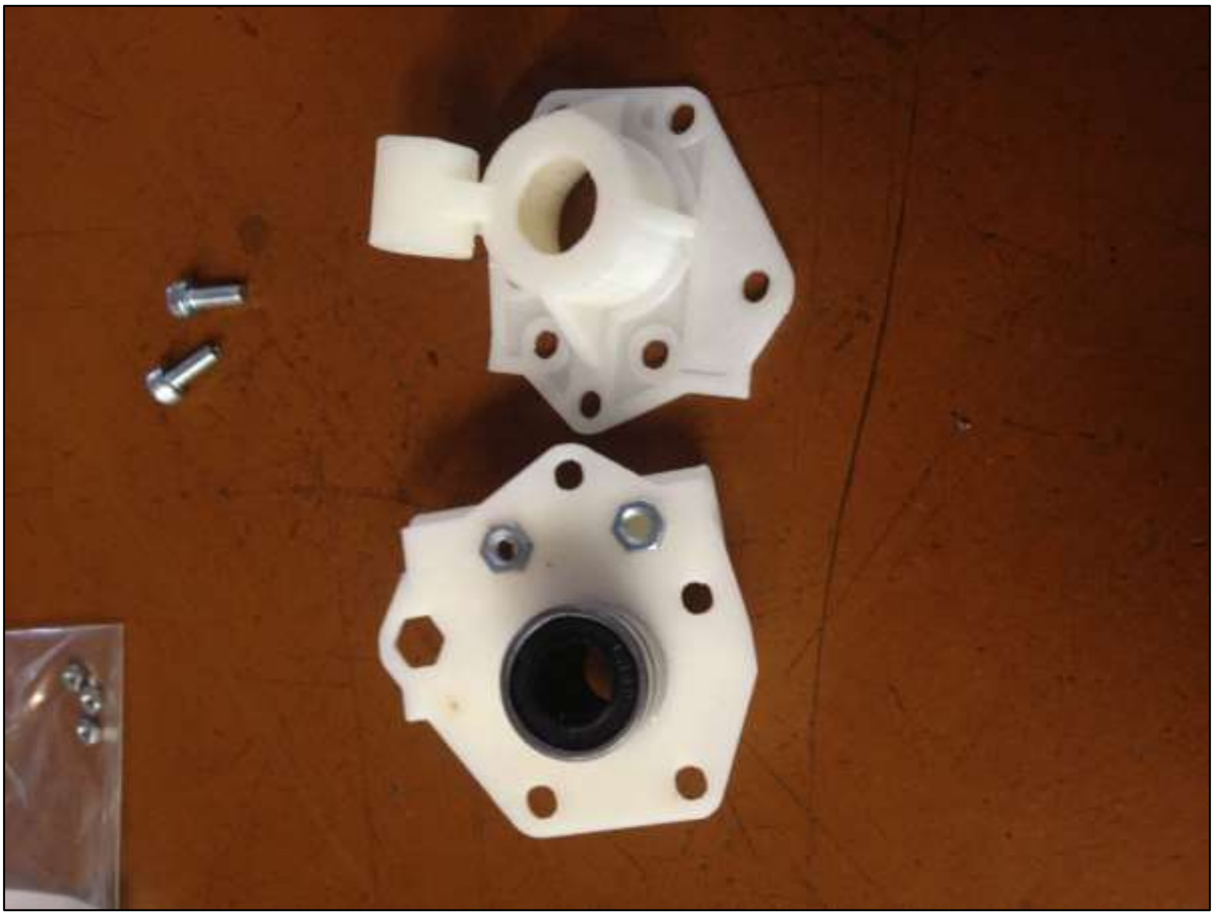
Design Guide

- The dimensions of the LM8UU linear bearings are 15mm x 8mm x 24mm (Outer diameter x inner diameter x length)
- The printed enclosure has dimensions

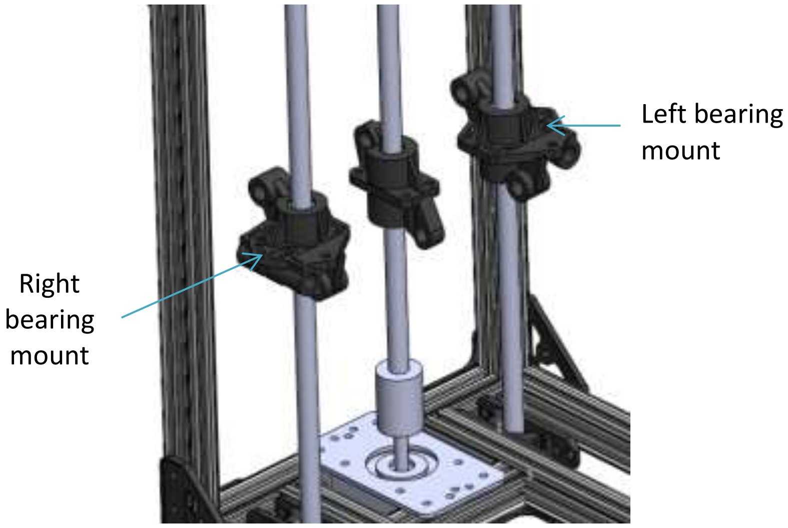
Inner diameter	15.5mm
Length	26mm

This gives a tight push-fit and allows no movement within the enclosure.

Assemble the bearing mounts by pressing the LM8UUs into top and bottom of the mounts, and secure with M3 screws and nuts.



Insert the bearings onto the steel rods. This must be done very carefully as you can easily pop the ball bearings out of the LM8UUs. ‘Left’ and ‘right’ here is looking from the back of the assembly, so the long horizontal tubes should be on the outside.



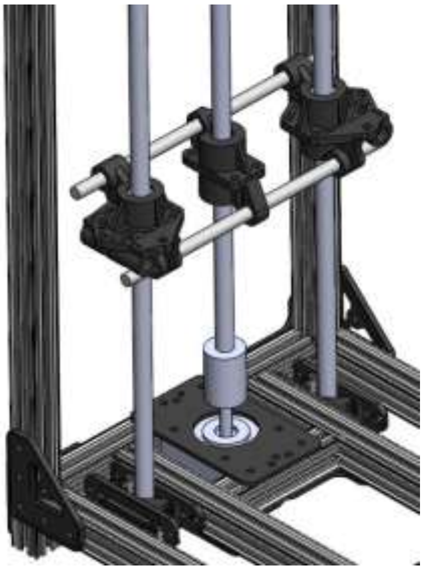
You can now clamp down the top shaft clamps with nuts and washers. At this point check that the shafts are perfectly parallel. Ideally use a plumbline to ensure they are vertical.

Finish off the structure by closing off the two open sides of the top frame with 150mm OpenBeam sections and L-plates.

In this step

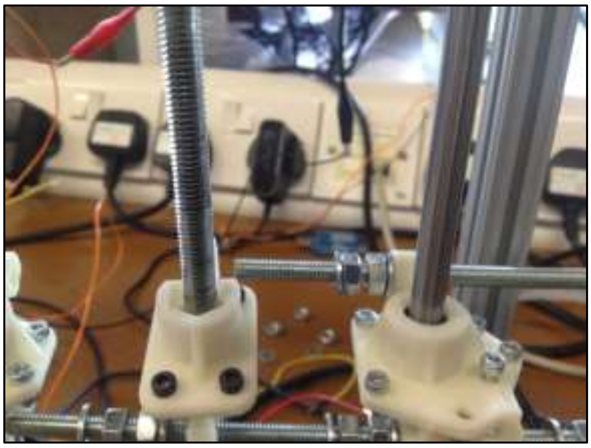
2	LM8UU linear bearings
1	3D printed bearing mount left - top
1	3D printed bearing mount left - bottom
1	3D printed bearing mount right - top
1	3D printed bearing mount right - bottom
4	M3 8mm caphead screws
2	OpenBeam half shaft clamps
2	OpenBeam 150mm aluminium extrusion
4	OpenBeam L-Brackets
20	M3 6mm buttonhead screws
32	M3 hex nuts
12	M3 washers

Step 13 – Stage frame

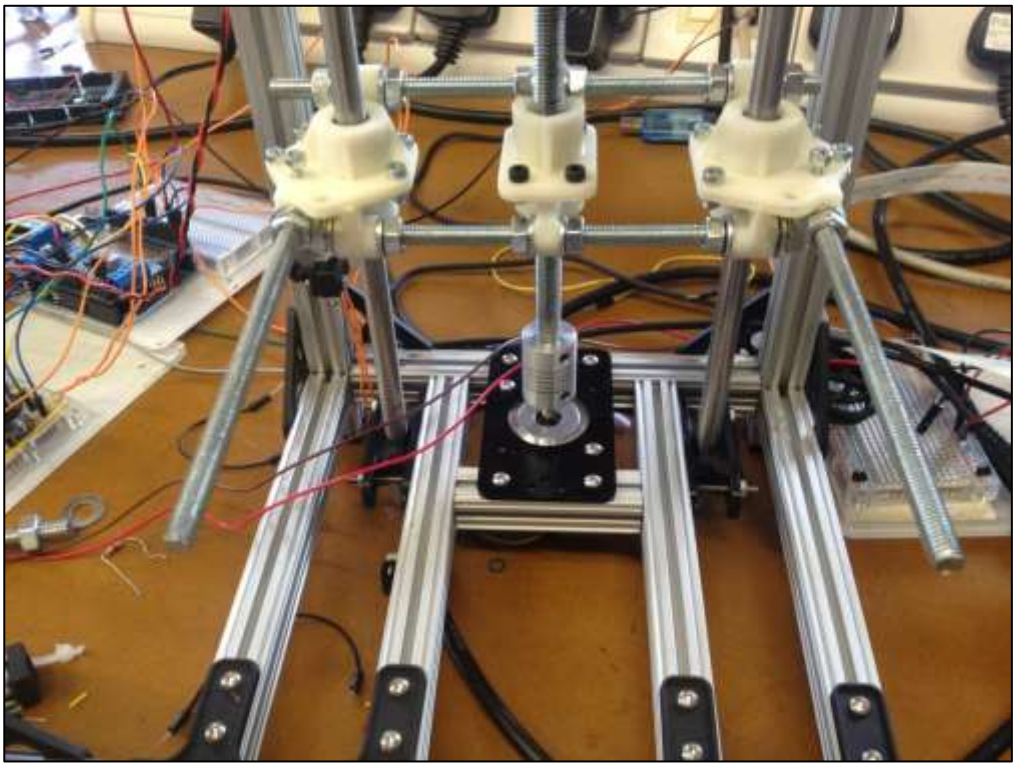


As the stage will only supported from the back of the assembly, it needs a rigid frame to support the platform and any lighting or other attachments.

The first stage is to connect the leadnut mount and bearing mounts together – this is what prevents the leadnut from rotating. Pass two 150mm threaded rods through the upper and lower holes in the mounts, and secure in place with M6 washers and nuts. This can be very fiddly: I would start by passing the threaded rod through two of the holes, add two nuts and washers, and then screw these along; passing the threaded rod through the third hole and out of the first hole so you have enough space to add the two nuts on the opposite side. Then slide the threaded rod back into the centre add the outer nuts. Repeat for the second threaded rod.



Insert M6 threaded rods through the tubes in the bottom halves of the bearing mounts and secure with nuts and washers. Make sure that the rods are parallel.



Design Guide

When designing the bearing mounts, I had in mind the idea of adding X-Y movement to the translation stage as well as just vertical Z-movement. The tubes in the bottom half of the mounts were therefore designed to take a 6mm smooth rod as well as a threaded rod. If you want to do this you can add two M3 hex nuts to the hexagonal holes in the top surface of the bottom piece. This will allow screws to be screwed in from the top to secure the smooth rod.

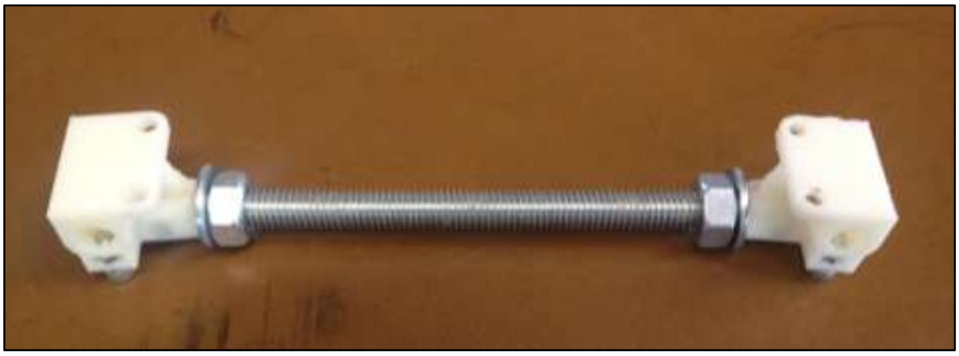
In this step

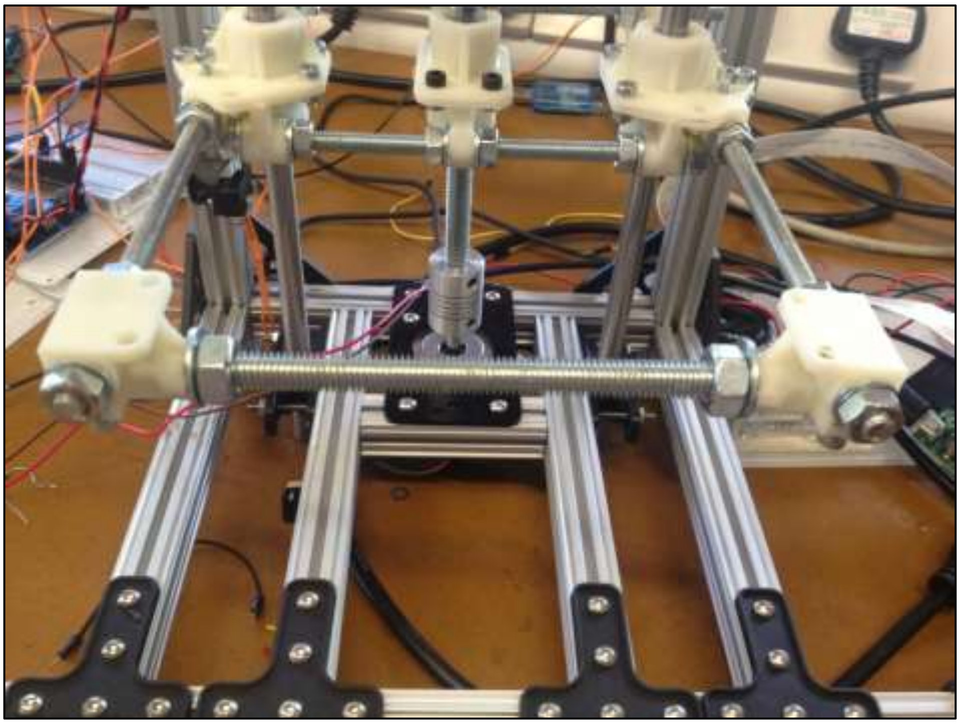
4	M6 Threaded rod 150mm
10	M6 hex nuts
10	M6 washers

Step 14 – Stage end pieces

First add M6 nuts and washers onto the protruding threaded rods.

Design Guide
When the stage end pieces were designed, we had a few pieces of M8 threaded rod left over from the leadscrew, so I designed them to fit both a 120mm M8 rod and a 150mm M6 one. You can use either, although I haven't had the chance to try them out with the M6 rods.





If using an M8 rod, screw two M8 nuts onto the rod and add washers to the outside. Insert into the 9mm holes in the stage mounts and then slide the stage mounts onto the existing M6 threaded rods.

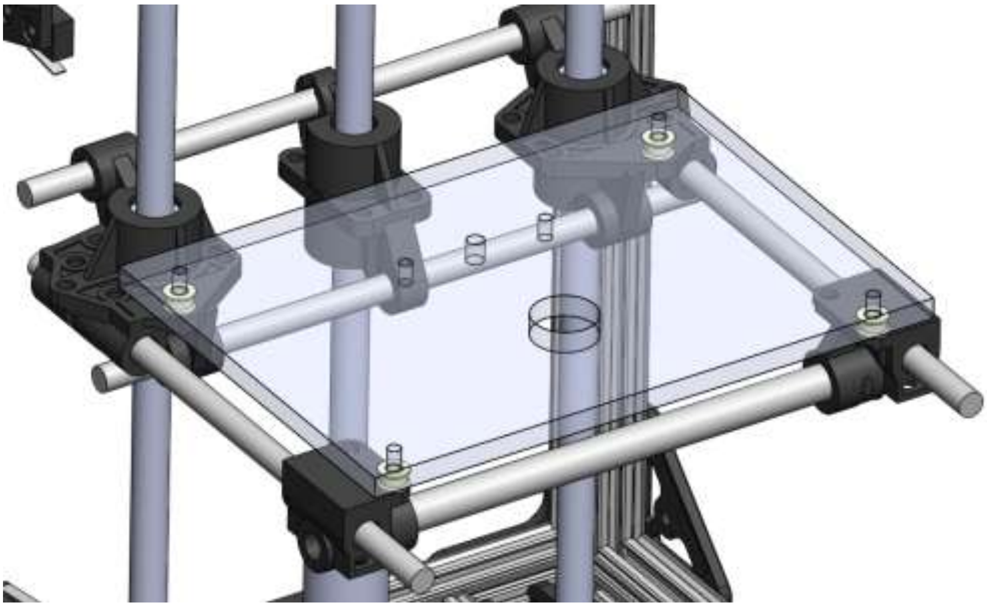
If using the M6 threaded rods, insert an M6 nut into the hex holes on each mount. Thread the rod through the first mount until the end of the rod is at about the end of the mount, then add two nuts, with washers to the other end of the rod. Thread the other mount onto the threaded rod, and adjust both mounts until they line up with the threaded rods attached to the stage.

In this step

2	3D printed stage end pieces
EITHER	
1	M8 threaded rod 120mm
2	M8 hex nuts
2	M8 washers
4	M6 hex nuts
4	M6 washers
OR	
1	M6 threaded rod 150mm
8	M6 hex nuts
6	M6 washers

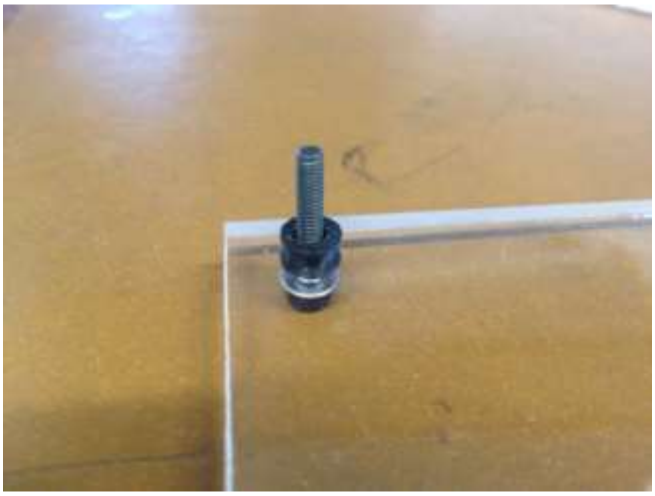
Step 15 – Mount stage

Line up the M3 mounting holes on the stage with the corresponding holes on the bearing mounts and front stage mounts. This may require you to adjust some hex-nuts.



Pass a 20mm M3 screw through each hole in the stage (I use a washer to protect the top) then through the small 3D printed spacers, and finally through the bearing and front stage mounts. Secure with M3 nuts – these fit in hex-holes which make it easier to tighten. If you haven't done so already tighten the remaining M6 nuts to secure the stage mounts.

Design Guide
You can very easily build a mechanism for levelling and adjusting the height of the stage by adding springs instead of the spacers. The hex-holes in the bearing and stage mounts should mean that you can then very easily adjust each corner with an allen key. A great and cheap source of suitable springs is ballpoint pens.



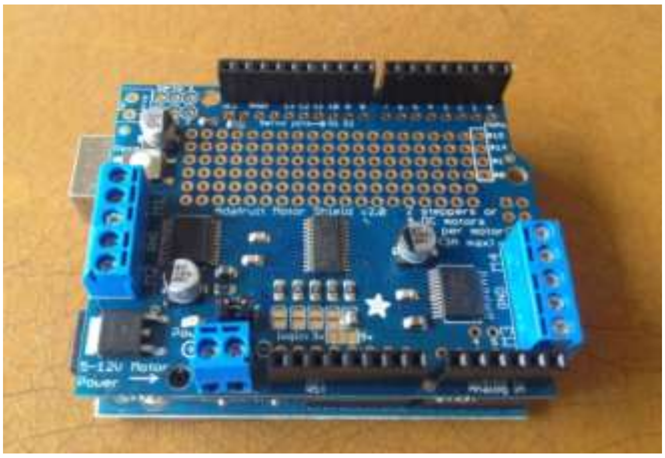
Design Guide
We used Perspex for our stage, as it is very stiff and gives a very flat surface. The holes were drilled with a CNC drilling machine, but hand drilling is fine. You could use a whole range of other materials such as metal sheet, or even 3D print one, but be warned that this will use a lot of material!

In this step

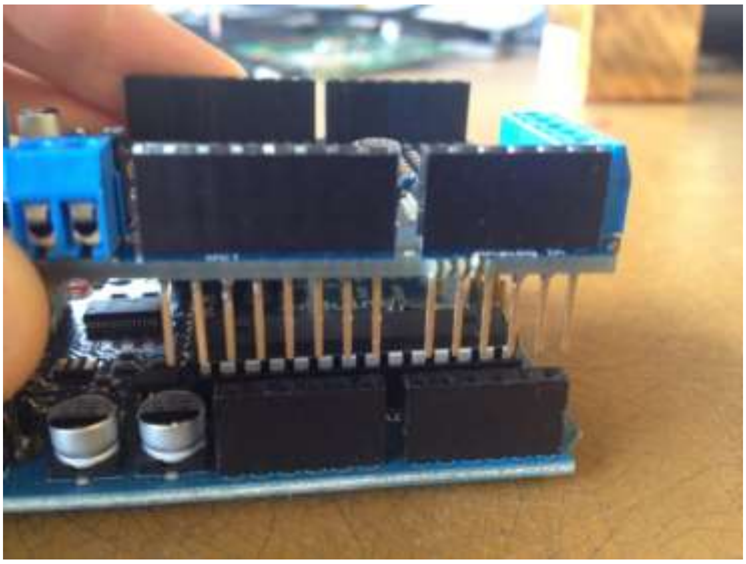
1	Perspex stage
4	3D printed spacers
4	M3 20mm caphead screws
4	M3 hex nuts
4	M3 washers

Step 16 – Connecting the stepper motor

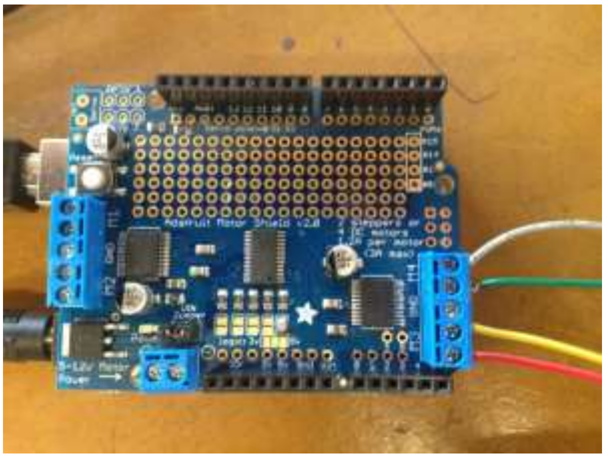
The stepper motor in this project was controlled by an Arduino microcontroller connected to an Adafruit V2 Motorshield. The Arduino in our project was actually a Duemilanove, but we would recommend using the newer version: the Arduino Uno R3. The motorshield is a stackable board which sits on top of your Arduino and contains all the circuitry required to run the motor. It is able to control up to two stepper motors or 4 DC motors per board, and as many as 32 shields can be stacked together (in theory!). I would definitely recommend using the Adafruit shield as they provide a very easy to use library for the Arduino. It also keeps most of the I/O pins available.

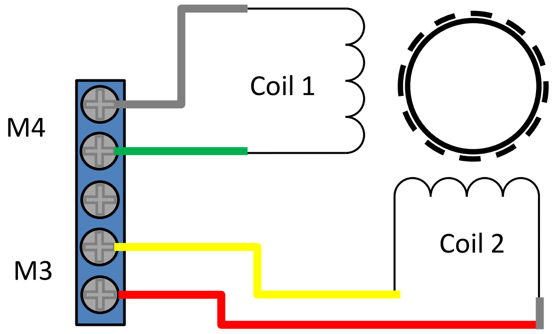


First you will need to assemble the shield by attaching stacking headers. Solder female headers to the input/output pins on the Arduino, and solder the male headers to the underside of the motor shield.



To connect the stepper motor to the motor shield, you must first work out how the stepper is wired. Bipolar stepper motors have four wires. These consist of two pairs – each pair is connected to one of two coils within the stepper motor. The two wires in the pair should be connected to one of the 4 ports on the shield; these are labelled M1 to M4. If using a unipolar stepper motor, you will have a 5th wire – this should be connected to the central terminal between M1 and M2 or M3 and M4. If you are using the same Adafruit 12V bipolar stepper motor that we used, then the red and yellow wires correspond to one coil, and the green and grey wires correspond to the other.





Note that while the Arduino should just about be able to run the stepper motors using just the USB power supply, you may need to provide an additional 5V power supply if you wish to add additional hardware later.

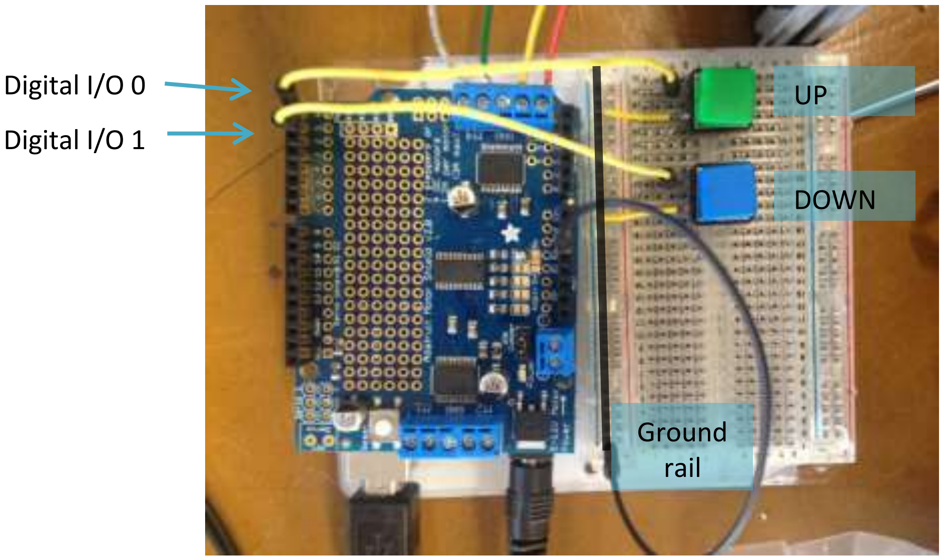
In this step

1	Arduino Uno Microcontroller
1	Adafruit Motor Shield for Arduino V2.0 Kit
1	Set of stacking headers

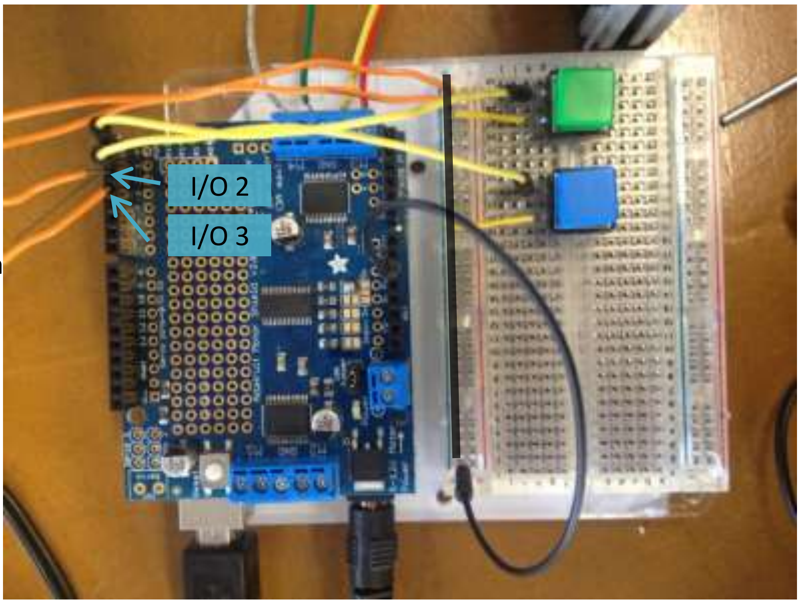
Step 17 – Sensors and input

The next two steps cover a very basic introduction to controlling the stage using switches and software. I haven't gone into much detail as these topics are covered in much greater depth in the OpenLabTools tutorials on hardware and software – see the OpenLabTools website for details.

As a basic introduction we will include four sensors: the two microswitches which you have already installed on the frame assembly; and two tactile switches for moving up and down. Connect the ground rail on your breadboard to the ground pin on the motor shield. Place the two tactile buttons onto the breadboard, and connect one pin from each to ground.



Next decide which you want to be your ‘up’ and ‘down’ button. Connect the remaining pin of the ‘up’ button to digital I/O pin 0 on the motor shield, and ‘down’ to I/O pin 1.

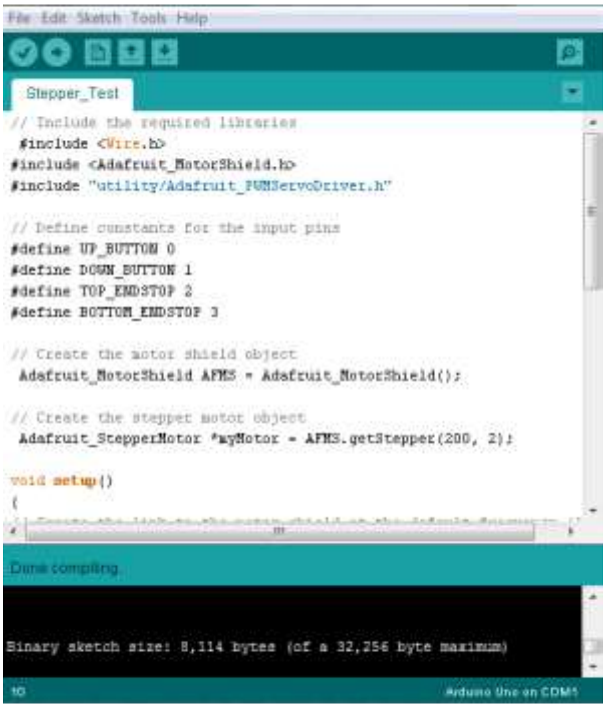


Similarly, connect one wire from each of your endstops to ground. Connect the other wire from the top endstop to digital I/O pin 2, and the bottom endstop to pin 3. If you haven’t done so already, connect the Arduino to your computer or laptop using a USB cable.

Step 18 - Software

Before you can write programs to control the stage, you will need to download the Arduino Integrated Developer Environment which can be found here. Follow the instructions on installing the software and connecting to your Arduino.

In order to get access to the stepper motor, you will also need a library to control it. If you are using the Adafruit Motor Shield you can download their motor shield library here. The linked page is part of a great tutorial on using the motor shield and talks about connecting to servos and DC motors in addition to steppers.



Once everything has been installed, open up the Arduino IDE. An Arduino program, or ‘sketch’ is programmed in C: for this tutorial I assume a very basic understanding of C programming. A sketch must contain at least two functions: a ‘setup’ function and a ‘loop’ function. The setup function is called once at the start of the program. The loop function is then called repeatedly.

Begin by running this simple test program, which sets up the motor shield, and sets the stepper motor to run forwards and backwards repeatedly. Copy-paste into your Arduino sketch or download from our Github here. Click the tick in the top-left corner of the IDE to check that the code compiles, then the arrow to download it to the Arduino. The program should begin running as soon as it has been downloaded. You should find that the stage moves up and down by about 3cm.

```
// Include the required libraries
#include <Wire.h>
#include <Adafruit_MotorShield.h>
#include "utility/Adafruit_PWMServoDriver.h"

// Create the motor shield object
 Adafruit_MotorShield AFMS = Adafruit_MotorShield();

// Create the stepper motor object
// The first param is the steps/rev (200 for the Adafruit stepper)
// The other param is the port: use port 1 for M1 and M2 or port 2
// for M3 and M4
 Adafruit_StepperMotor *myMotor = AFMS.getStepper(200, 2);


void setup()
{
// Create the link to the motor shield at the default frequency (1Hz)
AFMS.begin();

// Set the speed of the motor in rpm
myMotor->setSpeed(100);
}


void loop()
{
// Run the stepper motor. Parameters are:
//    #steps - number of steps
//    direction - options are FORWARD or BACKWARD
//    steptype - options are SINGLE, DOUBLE, INTERLEAVE or MICROSTEP
myMotor->step(500, FORWARD, DOUBLE);

// Wait for half a second
delay(500);

// Run the motor in the opposite direction
myMotor->step(500, FORWARD, DOUBLE);

// Wait for half a second
delay(500);
}
```

Note that some of the parameters in this code may be different if you have connected the steppers differently etc. Experiment to find out which work best. Once you are confident that the test program works, we can begin to introduce some input controls.

Add the following code to the top of the program.

```
// Define constants for the input pins
#define UP_BUTTON 0
#define DOWN_BUTTON 1
#define TOP_ENDSTOP 2
#define BOTTOM_ENDSTOP 3
```

Add the following to the setup function

```
// Setup inputs
pinMode(UP_BUTTON, INPUT);
pinMode(DOWN_BUTTON, INPUT);
pinMode(TOP_ENDSTOP, INPUT);
pinMode(BOTTOM_ENDSTOP, INPUT);
```

Finally, replace the 'loop' function with the following code

```
void loop()
{
  // Get the current states of the buttons
  bool up_pressed = digitalRead(UP_BUTTON);
  bool down_pressed = digitalRead(DOWN_BUTTON);

  // If both buttons are pressed, do nothing
  if(up_pressed && down_pressed)
    return;

  // If the up button is pressed and the stage hasn't
  // reached the endstop, go up
  if(up_pressed && !digitalRead(TOP_ENDSTOP))
    myMotor->step(1, FORWARD, DOUBLE);

  // If the down button is pressed and the stage hasn't
  // reached the endstop, go up
  if(down_pressed && !digitalRead(TOP_ENDSTOP))
    myMotor->step(1, BACKWARD, DOUBLE);
}
```

The full code can be found in the OpenLabTools github repository.

Final words

Thank you very much for using this tutorial, I hope it has been useful! This tutorial has mainly focussed on the mechanical aspects of a linear translation stage. Please check out OpenLabTools.org for more tutorials on automation, Arduinos, 3D printing as well as information relevant to our main microscope project.