Build OS-M Mk 1

Optical 3D Microscope Construction Guide

Imaad Syed
Aaban Syed
Dr. Lafe Spietz
Johns Hopkins University Applied Physics Laboratory

Dr. Aric Sanders
National Institute of Standards and Technology

This document is available on:
https://github.com/isyedjr/Open-Source-Microscope-Mk-1

May 2021

Table of Contents

List of Tables

List of Figures

1. Introduction

Standard industry-grade microscopes are very expensive, costing several thousand dollars just for one. This is the main limiting factor as to why third world country hospitals, interested students or even hobbyists can't just go out and buy one. In order to combat the cost problem, we developed the OS-M (Open Source Microscope) Mk 1 prototype. This costs at most $120 to replicate, and is done using classic DIY methods. In other words, you can put this together in your house without very sophisticated tools. All you need is a soldering iron, cardboard, plastic, a hot glue gun, hobby knife and some hobby electronics. While dealing with the cost problem, this also helps to address the growing problem of e-Waste, by repurposing old DVD drives for microscopy.

This document contains instructions on how to replicate the OS-M Mk 1 prototype. There is also a small section near the end of this document to assist with setting up the software and operating the microscope. Sample scans from the OS-M Mk 1 are in the Scans folder in the GitHub repository, and an image of the prototype can be seen below:

Fig. 1. An image of the OS-M Mk 1.

Keeping in the spirit of open-source projects, please feel free to make your own modifications or even your own prototype. This document is meant to provide a general direction, and will help you make the OS-M Mk 1.

2. Replication Instructions

The construction process is relatively simple, compared to standard robotics/electronics projects. This will be divided into 4 parts - imaging head, motion stage, control and electronics. In terms of materials, you'll need:

- Cardboard (recycled, assumed to have negligible cost)
- 1 Large Breadboard (comes with Arduino kit)
- 1 Arduino Uno (kit is $\sim$ \$35.00)
- 1 Raspberry Pi 3B+ ($\sim$ \$35.00)
- 1 PiCamera ($\sim$ \$30.00)
- 3 "old" DVD/CD Drives ($\sim$ \$5 online, further assumed to be \$0 as this is e-Waste)
- Jumper cables (comes with Arduino kit)
- Flashlight ($\sim$ \$2.00, can alternately use LEDs in Arduino kit)
- 3 MP6500 stepper motor drivers ($\sim$ \$18.00 total)
- Preferably 1 external monitor (for viewing the scans and/or live feed)

Additionally, make sure that you have a hobby knife, soldering iron, hot glue gun, needle-nose pliers, Flathead and Philips head screwdrivers and a well-ventilated area. Use your screwdrivers to deconstruct the 3 DVD drives to get 3 OPUs. The OPU (with separated lens) are shown below:

2.1 Imaging Head

In order to put together the imaging head, first connect the PiCamera to the provided ribbon cable, and connect to the Raspberry Pi. Extract the lens from the DVD OPU with a small flathead screwdriver, and remove one of the rubber rings from the corners of the OPU. Glue the lens so that it is fixed in the hole of the ring.

Take 2 pieces of duct tape, one face up, one face down, and have half of each strip overlap. Repeat this process to create a "sheet" of duct tape. This shouldn't be very long, but sufficient enough to maintain a small gap (roughly 3 mm) between the center of the lens and PiCamera. The flat side of the lens should be pointing outwards, towards you. Roll the sheet around the PiCamera, and mark a hole for the lens. Cut out this hole (sheet should be folded over around twice, and glue the ring-lens assembly to the hole. Place the duct

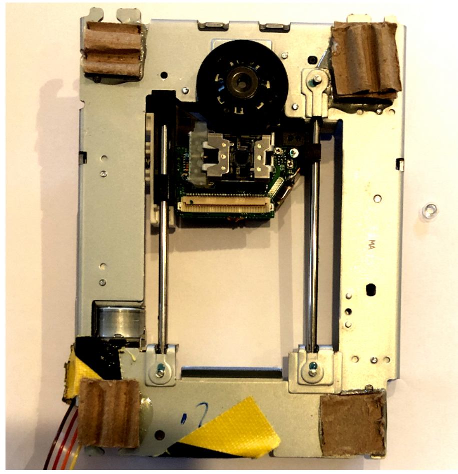

Fig. 2. An image of the DVD OPU with separated lens.

Fig. 3. An image of OS-M Mk 1's imaging head.

tape assembly around the PiCamera, so the lens is over the PiCamera lens, and hot glue the duct tape to itself and green PiCamera board (away from electronics) to secure the lens assembly. See figure 4 for a visual.

Green represents the duct tape, and the blue box is the PiCamera. The lens and OPU ring are shown, positioned on the lens of the PiCamera. This assembly is what makes the OS-M Mk 1 a microscope, as this imaging head is what allows the OS-M Mk 1 to take pictures, and obtain over 150x magnification.

2.2 Motion Stage

The motion stage is very similar to the mechanical components of a 3D printer. 2 OPUs are used to position the imaging head at the appropriate X and Y, and a third OPU is used to raise/lower the slides to get a clear image of the sample.

Fig. 4. A more detailed diagram of the imaging head.

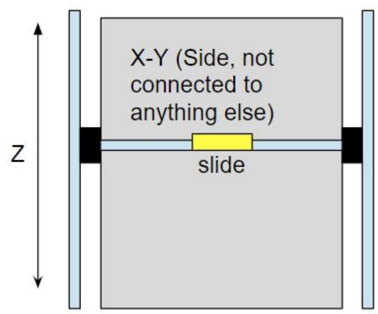

Fig. 5. A simple illustration of the motion stage z-axis.

In order to construct these, take 3 OPUs, glue cardboard to the DVD objective lens assembly, and solder 4 wires to each stepper motor. To assemble the X-Y portion of the stage, glue a large piece of metal (usually easiest to use exterior sheet of DVD drive) to an OPU, with the DVD objective head pointed up. A counterweight (rubber band some of the DVD drive PCBs to the other end of the metal sheet) may be needed. Glue a second OPU with the DVD head pointed down, with the motion directed in a perpendicular direction to the first OPU. Glue the imaging head to the bottom of the second OPU, and this is the X-Y portion. For a visual, refer to Fig. 6.

In order to assemble the Z axis portion of the stage, take the final OPU and stand it on its side, so that the objective head moves up and down. Glue a long cardboard piece and cut a hole out in the middle of the cardboard to fit a slide. This completes the z-axis. Now, position the motion stage components into a similar fashion to Fig. 7-9.

Now, you have the hardware for the microscope complete. All that's needed now is input and electronic control.

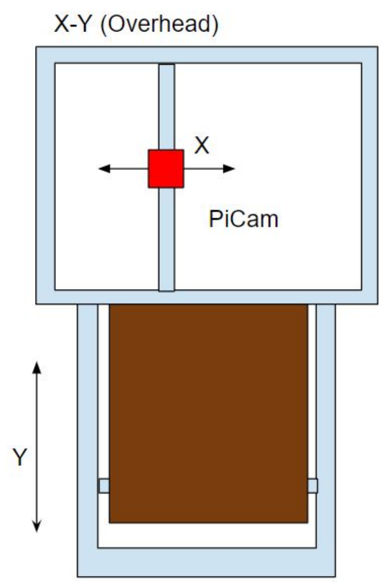

Fig. 6. A simple illustration of the motion stage x and y axes.

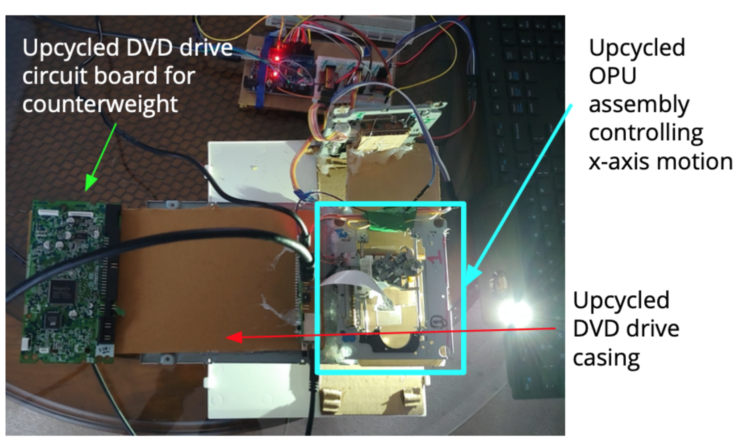

Fig. 7. An image of prototype with the X-Y axes highlighted.

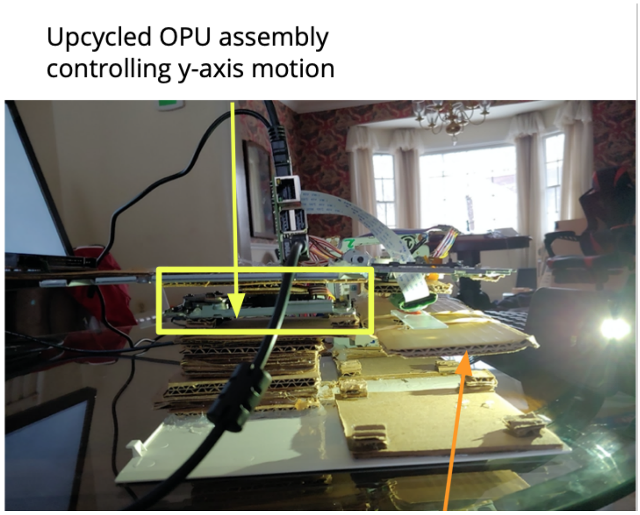

Fig. 8. Another image of prototype with the X-Y axes highlighted.

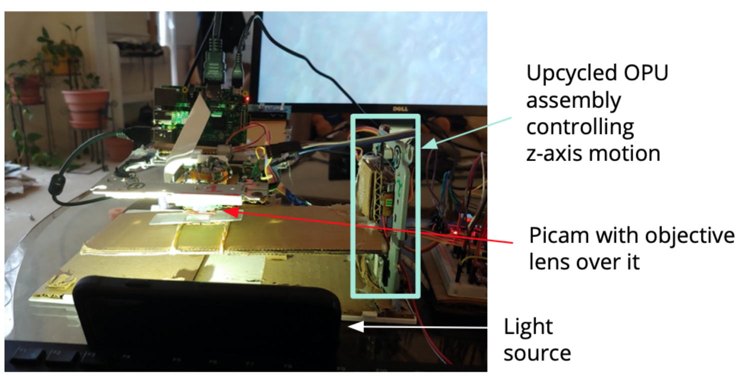

Fig. 9. An image of prototype with the Z axis highlighted.

2.3 Control

The microscope needs input to control the motion stage, as only the position of the imaging head should change to view the sample. Thus, a joystick and two buttons are used. The joystick controls X-Y motion, while the two buttons control Z axis motion.

2.4 Electronics

The electronics need to read input and put out the appropriate output. The only wiring to microscope components is to be done to the Arduino, as microcontrollers are best for controlling motors and other electronic devices. Raspberry Pis are excellent for computing, but making them work with analog inputs is challenging.

Fig. 10 covers the control electronics, while Fig. 11 refers to outputs. These are just schematics, and should be used as general wiring guides, as specifics will vary with the use of different types of breadboards. Additionally, all stepper motors are not the same. This will likely affect how you connect the motors to the MP6500 controllers.

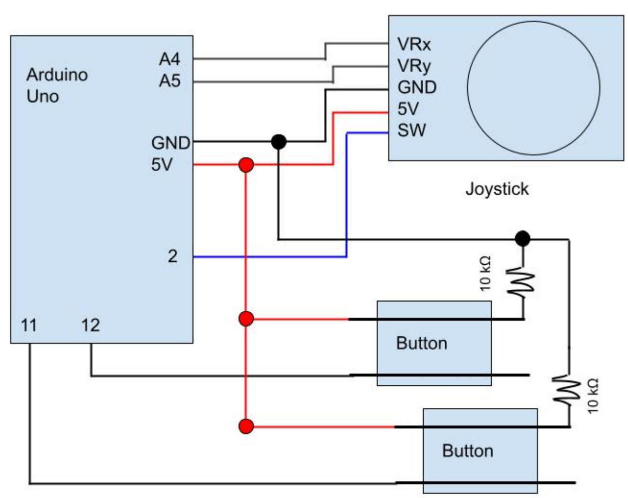

Fig. 10. Electronics schematic/diagram for control equipment. Circles represent junction points.

Note that the same Arduino Uno board is used in each schematic. There are 2 separate diagrams to make replication easier. Additionally, ”MS1 2” means that both MS1 and MS2 are connected to 5V. The diagram for the MP6500 that was used is shown in Fig. 12.

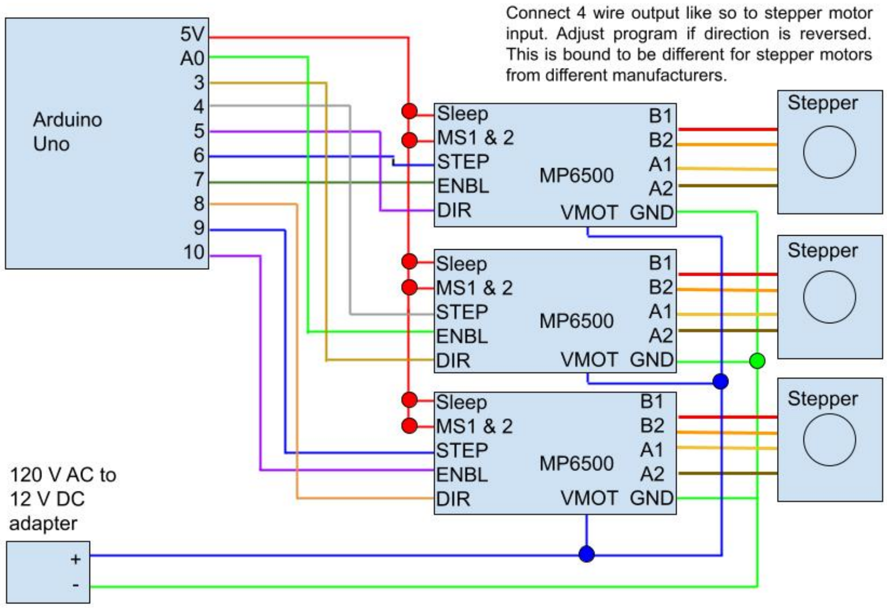

Fig. 11. Electronics schematic/diagram for stepper motor outputs. Circles represent junction points.

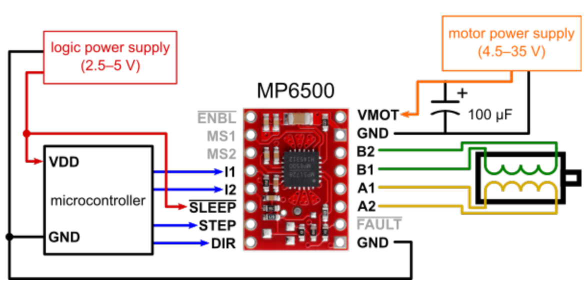

Fig. 12. MP6500 diagram from https://www.robotgear.com.au/Product.aspx/Details/5672-MP6500-Stepper-Motor-Driver-Carrier-Digital-Current-Control.

3. Software Setup

3.1 Arduino

In order to setup the Arduino (and motion stage), download the file ”motion.ino”. Double click it to open, and if prompted to make a folder, proceed to do so. Connect the Arduino to the computer with ”motion.ino” open, and press upload. You may need to check if the right port is selected.

The Arduino program simply translates user input into motion on the motion stage. The logic is best summarized in Fig. 13.

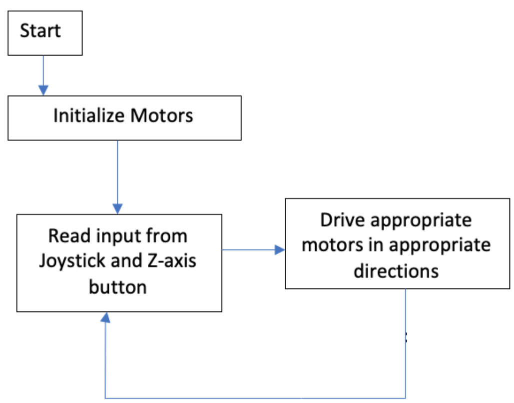

Fig. 13. Arduino logic diagram.

3.2 Raspberry Pi

The Raspberry Pi is needed to manage the PiCamera and display the feed to an external monitor. To set this up, start up your Raspberry Pi, and download ”viewer.py” from the GitHub repository. Make sure that you set up the PiCamera before running ”viewer.py”.

Running the program can be done in 2 different ways:

- Running directly from terminal with ”python viewer.py”
- Opening up the program in a Python IDE and running it

The first method is much simpler, as there is no IDE intermediate. Just be sure you are in the right directory before typing the command in the terminal on the Pi. Also make sure that the Arduino is powered, in order for the motion stage to move, while the PiCamera is on.

The logic is best summarized in Fig. 14.

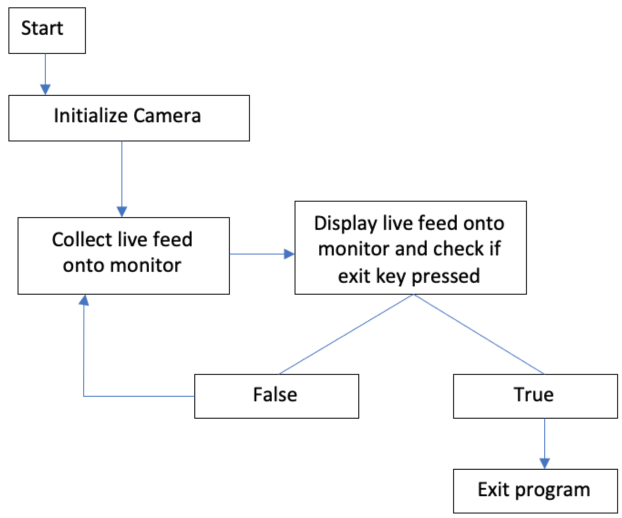

Fig. 14. Raspberry Pi logic diagram.

4. Operation Instructions

This operates similar to a regular optical microscope, except that imaging is computerized, and the microscope moves based on input to a joystick and 2 buttons. There are a few things you should know before operating OS-M Mk 1.

4.1 Making & Mounting Slides

Like regular microscopes, you should try to flatten your samples before imaging. The microscope otherwise only focuses on certain distances, due to the fixed focal length, which results in a partially blurry image. If you seek to only view a particular region, then this step is not neccessary.

Slides should be placed onto the Z-axis arm and raised up for viewing. You might need to rubber band the slide to this arm, in order to keep the slide from moving.

4.2 Imaging

In ”viewer.py”, there are several commands programmed in for imaging. They are listed in the table below:

Table 1. Microscope Keyboard Functions

Key	Function
x	Quit
q	Take Picture
i	Invert Color
r	Toggle Record (press once to start recording, once to stop)

The videos are saved in the ”.h264” format, and the following should be used to view the video:

omxplayer [videoname.h264]

Videos are saved as ”video[x].h264”, where x is some number starting from 1, increasing by 1 each time a video is recorded.

Don’t forget about lighting the sample! Use the flashlight to shine light on the sample accordingly, in order to yield the best image.

5. Improvements

First, let's summarize the strengths and weaknesses of this prototype in the following table. Ratings will be given from 0 to 10, with 10 being very successful, and 0 being a failure. Note that this is a self-assessment.

Table 2. Quality Analysis

Criteria	Score
Inexpensive?	8
Easy to Replicate?	5
Impact on e-Waste?	7
Functionality	9
Use of Available Technology	5
Educational Value	9

OS-M Mk 1 is sufficient for the intended purposes of education and inexpensive biomedical imaging, but it can definitely be improved on. The cost can likely be reduced, replication can be made easier, more e-Waste can be used instead of hobby electronics, and there is quite a bit of the DVD OPU which is not used. Thus, these are the top areas for improvements.

We will be working on addressing these criteria, and encourage you to do the same, after getting comfortable with the ideas/concepts behind open source microscopy. Remember to feel free to clone, experiment and share!