Automated Transilluminator
User Manual

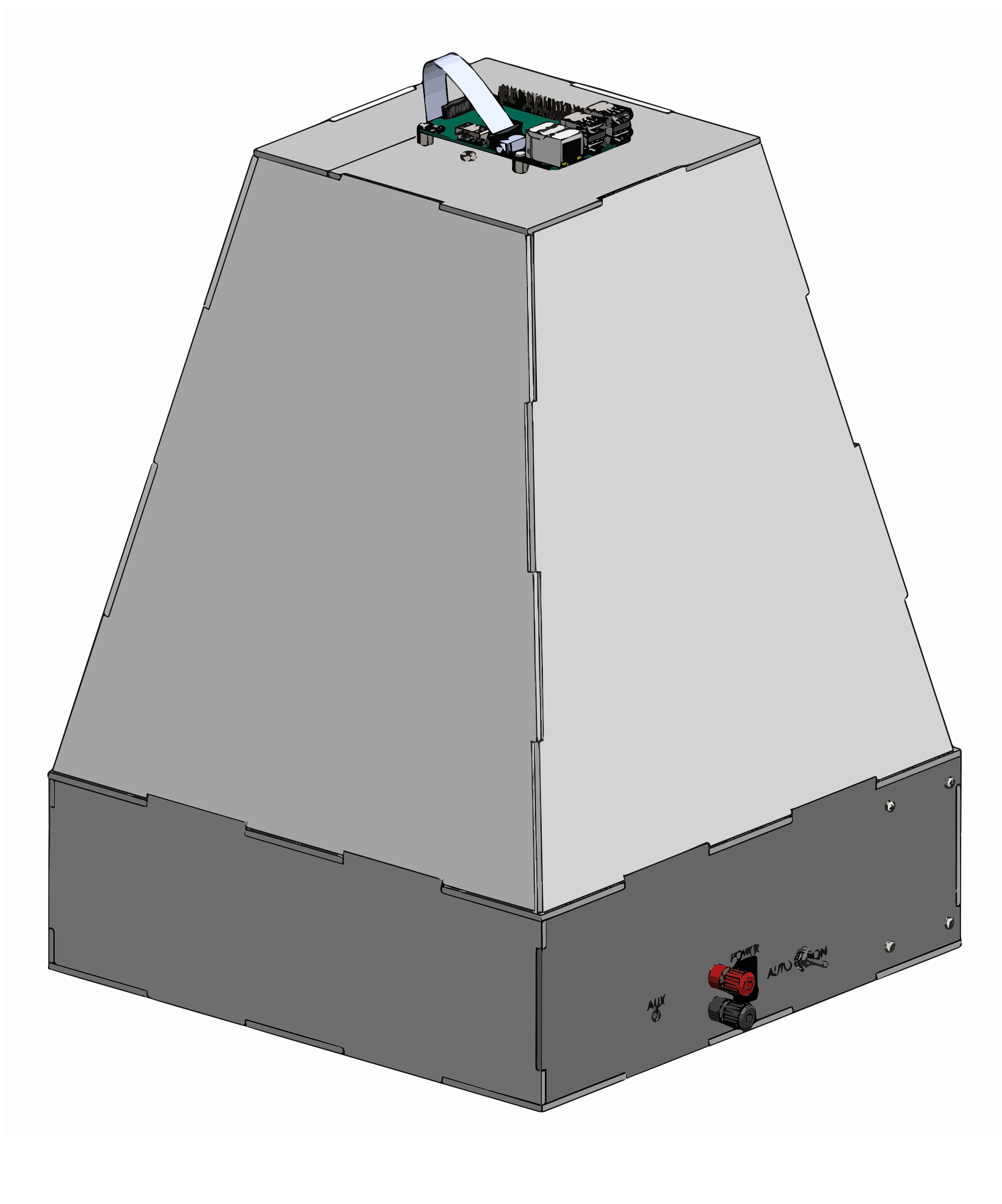


Table of Contents

AUTOMATING THE REMOTE IMAGE ACQUISITION SYSTEM USING THE RASPBERRY PI

The Raspberry Pi is a compact single-board computer that was developed to promote the teaching of basic computer science. Due to its affordability, flexibility, and ease of use, it has been adopted by the do-it-yourself (DIY) community and research laboratories alike. This Raspberry Pi-based automated image acquisition system de-couples experimental setup from data generation to eliminate the timing constraint of students being physically present in the lab during time-course fluorescence imaging. The system can also capture individual still images without automation.

Before getting started, please read the following guides on the official Raspberry Pi website to familiarize yourself with the basic principles of using the Raspberry Pi.

- What is a Raspberry Pi?
- Quick Start Guide

More information, including official documentation and frequently asked questions, can be found here.

To set up the Raspberry Pi operating system (OS) for our remote image acquisition system, we have created an SD card disk image file which is available for download. This image file (.img) can be copied onto an empty SD card to create a clone identical to the original OS that we use.

1. Cloning the SD Card Image onto an empty SD Card

You will need a blank SD card at least 8 GB in size. The image will provide a fully functional OS for the Raspberry Pi that is pre-loaded with all necessities to use it for automated image acquisition. Please download the SD card image (hosted on the University of Pennsylvania Scholarly Commons) onto your desktop or laptop computer.

For instructions on how to write an image onto an SD card for the Raspberry Pi, follow the below link depending on what operating system you use on your desktop or laptop computer.

- Linux
- Mac OS
- Windows

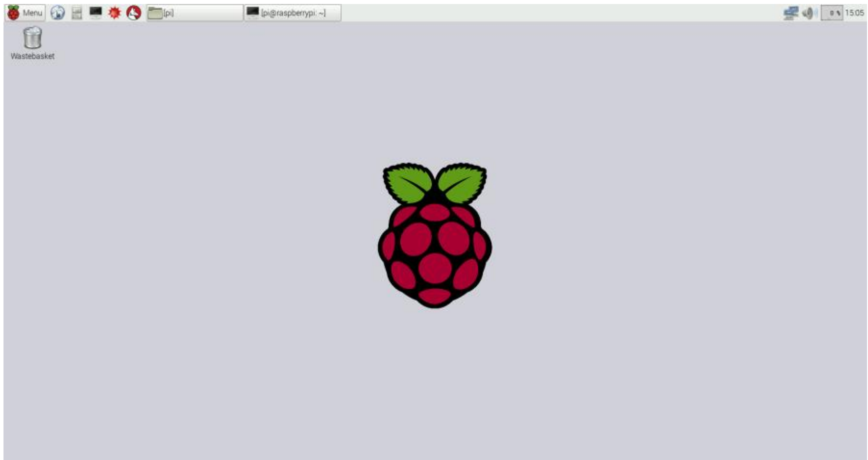
After cloning the SD card, proceed to the next section.

2. Booting up the Raspberry Pi

We will now use the cloned SD card to boot up the Raspberry Pi. Follow the below steps:

1. Insert the SD card into the slot on the Raspberry Pi.
2. Connect the Pi Camera by following the video tutorial: Connecting the Camera Module
3. Plug in an HDMI monitor, a USB mouse and a USB keyboard.
4. Connect the Raspberry Pi to the Internet via an Ethernet Cable.
5. Turn on the Raspberry Pi by plugging in the microUSB power supply. The device will boot up to a **Terminal** (black screen). Once the boot process has completed, a login prompt will appear: `raspberrypi login:`

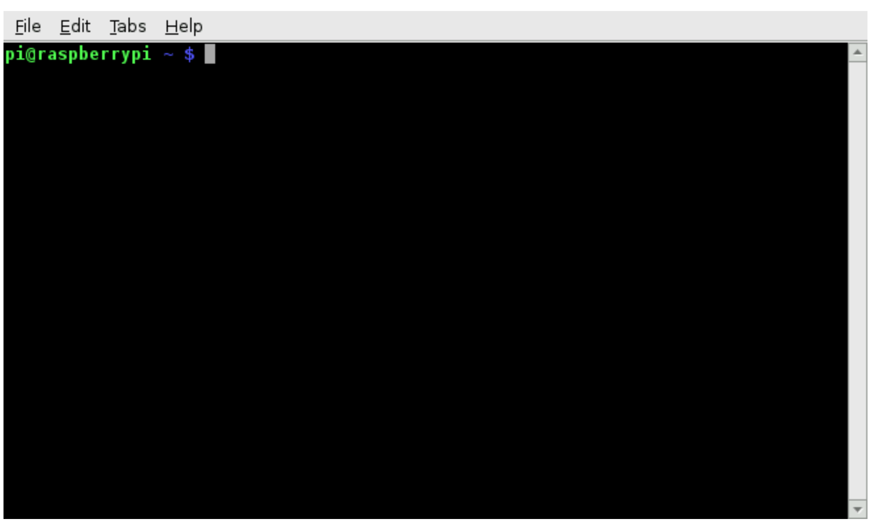
6. The default username is `pi`, and the password has been set to `synbiolab`. Log in by typing the username (`pi`) into the terminal. Press Enter, and then type the password (`synbiolab`). Note: you will not see any writing appear when you type in the password (Linux security feature).
7. After you have successfully logged in, you will see the command line prompt: `pi@raspberrypi ~ $`. You are now currently in the home directory for the Raspberry Pi user 'pi'.
8. Type `startx` and press Enter to load the Raspberry Pi's graphical user interface.
9. Once you see a screen like the image below, proceed to the next session.



3. Using the Raspberry Pi Terminal

The Raspberry Pi graphical user interface provides many of the same functions that you would find on Windows or Mac OS, such as a web browser and a file manager. To run the automation software, however, we need to access the command line. On the Raspberry Pi desktop, there will be the following icon:

Click on the icon to open **LXTerminal**, which will give you access to the command line while remaining in the Raspberry Pi GUI. You should see a black screen pop-up like the following:



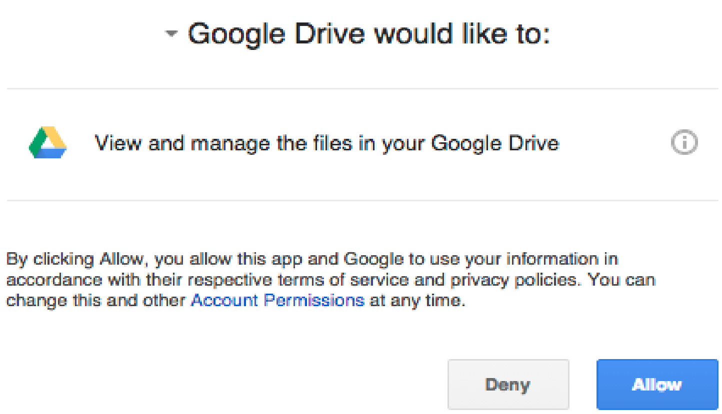
1. On the terminal, you can navigate to different directories by using the change directory command (cd). Navigate to the SynBio directory by typing `cd SynBio` into the terminal. The terminal should now show `pi@raspberrypi ~ /SynBio $`.
2. Type `ls` into the terminal to list the files in the current directory and verify that the `SynBioAutomation.py` file is present. This is the Python script responsible for automating the remote image acquisition system.

4. Initializing Google Drive

Now that you are inside the SynBio directory, you need to initialize the Google Drive client. This will allow you to push files or folders within the SynBio directory from the Raspberry Pi to the cloud.

1. While inside the SynBio directory, type `drive init` into the terminal.
2. Once the command is run, you should see the following line - `Visit this URL to get an authorization code` followed by a URL.
3. Go back to the Pi's graphical user interface and open the web browser by clicking on the icon shown below.

4. Copy the URL and paste it in your browser. You will be prompted to log into a Google account. Log onto the account whose Drive you want to sync with your Pi. When you login you will see the following request for permission. Click on "**Allow**".



5. You will be redirected to a page that provides an authorization code allowing you to sync the Pi with the specified Google Drive account. Copy the code and paste it in your Pi Terminal in the line, `Paste the authorization code:`
6. You are now able to push files saved in the `SynBio` directory to Google Drive of the specified account.

The above steps need to be repeated any time you want to switch the Google Drive account that you are using, but do not need to be repeated between experiments if the Google Drive account is remaining constant. We therefore recommend creating a central Google Drive account that you can use to push files to, then sharing the individual folders with students using the online Drive interface.

5. Taking Pictures with the Raspberry Pi

1. Test to see if the camera is working by taking a picture from the terminal. Type in `raspistill -o image.jpg` and hit enter. A live image preview should appear. Afterwards the image will be saved as `image.jpg` in the current directory (`SynBio`).
   a. The above command can be used to capture still images without automation, e.g. endpoint analysis or live demonstration.
   b. More information about using the Pi Camera from the terminal can be found here: [Taking Pictures Using the Command Line]
2. The SynBioAutomation script makes use of the PiCamera Python Library, which is already included in the SD Card Image. More information about the PiCamera library can be found at the below links. The code is also extensively commented with explanations (Supporting File 1).
   a. [Taking Pictures using the PiCamera Python Library]
   b. [Documentation for the PiCamera library]

6. Running the Image Acquisition Script

You should now be all set up to run the automated image acquisition script**. In the terminal, make sure you are still in the SynBio directory (see 3. Using the Raspberry Pi Terminal). The below commands only work as long as you are in the same directory as the `SynBioAutomation.py` file.**

Initializing the Image Acquisition Script

The first step is to initialize the image acquisition script. This step creates the folder in which the images for this experiment will be stored, and initialization should be repeated for every new group, trial, etc. In addition, this step locks the camera settings that will be used for the duration of the experiment, so images can be easily compared across time points.

1. In the terminal, type in the following command:
   ```
   sudo python SynBioAutomation.py Initialize FolderName
   ```
   where `FolderName` is the user-defined name of a new folder that will be created for image storage.
2. The status of the script will be printed into the terminal. Wait until `Initialized` is printed in the terminal before proceeding.
3. In the browser, log into the Google Drive account that you chose to sync with the Raspberry Pi. In the main MyDrive folder, there will be a new folder with the name you specified above. In that folder there will be an image titled Initial.png. This is the photo taken during initialization.

4. All subsequent images will be uploaded to this folder, until a new initialization script is run or the automation is ended.

Setting up the Image Acquisition Time Course

With the camera successfully initialized, you can now begin the automated time course.

1. In the terminal, type in the following command
   `sudo python SynBioAutomation.py TimeCourse`
2. The status of the script will be printed into the terminal. When `Time Course Active` is printed into the terminal, the script has concluded. An automated time course has now been set up that will take images every hour (from the time started) and will push these images to the Google Drive folder created above.

Taking Photos outside of the Time Course

1. At any point during the Time Course, an additional image can be captured in real time by using the following command in the command line (while still in the SynBio directory)
   `sudo python SynBioAutomation.py`
   This will take an image in real time without disrupting the current TimeCourse.
2. If you want to reset the time course, simply run the command
   `sudo python SynBioAutomation.py TimeCourse`
   and the automation will be adjusted to take photos hourly from the current time.

Ending the Automation Script

1. At the end of the experiment, simply run the following command in the terminal to end the time course and make sure no further pictures are acquired:
   `sudo python SynBioAutomation.py End`

7. Accessing the Raspberry Pi Remotely

1. Please see the Assembly Instructions (page 13) for step-by-step instructions on attaching the Raspberry Pi and the Pi Camera Module to the custom-build transilluminator.
2. Once the Raspberry Pi is attached to the transilluminator, it is easiest and most economical to access it remotely rather than connecting a local external monitor, mouse, and keyboard to it. Setting up the Raspberry Pi for remote access helps with scalability because multiple Pi's can be accessed by a single computer. More information for Remote Access can be found here: Remote Access
3. We recommend using SSH (Secure Shell) to remotely access the Raspberry Pi. SSH allows remote access to the command line through each Raspberry Pi's unique IP address. From an SSH terminal, all of the steps in **Sections 3-6** above can be followed, with the only difference being that

the commands are issued on a remote computer rather than a keyboard/monitor connected to the Pi.

4. Follow the Instructions below depending on your operating system to set up SSH into the Raspberry Pi
    a. Linux & Mac OS
    b. Windows
5. We recommend utilizing your organization's tech support to set up a static IP address on each Raspberry Pi and help with the remote access. Once each device is assigned a static IP address, you can keep track of each Pi through its IP without worrying that it will change. This will help with keeping track of experimental groups.

Parts List

Quantity	Unit Price		Vendor	Part Number	Part Name	Assembly Guide Annotation
				Raspberry Pi		
1	$	35.00	MCM Electronics	83-16530	Raspberry Pi™ 2 Model B 1GB Project Board	**a**
1	$	8.99	MCM Electronics	83-17022	16Gb Sandisk Ultra MicroSDHC Class 10 Card	
1	$	0.99	MCM Electronics	83-16447	microSD Card Adapter to SD	
1	$	25.99	MCM Electronics	28-17733	Raspberry PI 5MP Camera Board Module	**b**
1	$	5.99	MCM Electronics	28-19335	5VDC 2A Regulated AC Power Adapter - 48" Cord Micro USB Plug	
SUBTOTAL	**$**	**76.96**				
				Optics		
1	$	29.10	Thorlabs	LA1908-A	N-BK7 Plano-Convex Lens, Ø1", f = 500.0 mm, AR Coating: 350-700 nm	**c**
1	$	15.23	Thorlabs	LMR1	Lens Mount with Retaining Ring for Ø1" Optics, 8-32 Tap	**d**
1	$	6.80	Thorlabs	MS2R	Mini Series Mounting Posts, Ø6 mm, L = 2"	**e**
1	$	6.08	Thorlabs	MS1R	Mini Series Mounting Posts, Ø6 mm, L = 1"	**f**
1	$	16.10	Thorlabs	MSRA90	Mini-Post Right-Angle Post Clamp, Fixed 90° Adapter	**g**
1	$	1.90	Thorlabs	AP8E4E	Adapter with External 8-32 Threads and External 4-40 Threads	**h**
SUBTOTAL	**$**	**75.21**				
				Circuit		
1	$	6.72	Digikey	V2025-ND	2" x 3" Perforated Protoboard**	**i**
1	$	0.10	Jameco	690742	Resistor Carbon Film 330 Ohm 1/4 Watt 5%	**aa**
1	$	0.10	Jameco	691340	Resistor Carbon Film 100k Ohm 1/4 Watt 5%	**bb**
1	$	1.75	Jameco	2210626	IRLB8721 HEXFET® Power N-Channel MOSFET	**cc**
1	$	0.75	SparkFun	PRT-10571	Screw Terminal 2.5mm Pitch (2 Pin)	**dd**
2	$	0.19	Jameco	233437	Connector Wire to Board Header 2 Position 2.54mm Solder Right Angle Thru-Hole	**ee**
2	$	0.15	Jameco	234798	Straight Connector Housing Receptacle 2 Position 2.54mm	**ff**
1	$	0.25	Jameco	233445	Connector Wire to Board Header 3 Position 2.54mm Solder Right Angle Thru-Hole	**gg**
1	$	0.25	Jameco	234801	Connector Housing Receptacle 3 Position 2.54mm Straight	**hh**
7	$	0.06	Jameco	234923	Connector Contact Female 1 Position Crimp Straight Cable Mount Bag	
1	$	14.95	MPJA	29902PS	3-12V, 2A Selectable Output Supply	
SUBTOTAL	**$**	**25.97**				

** May also use custom PCB provided with CAD files (Supporting Files 1)

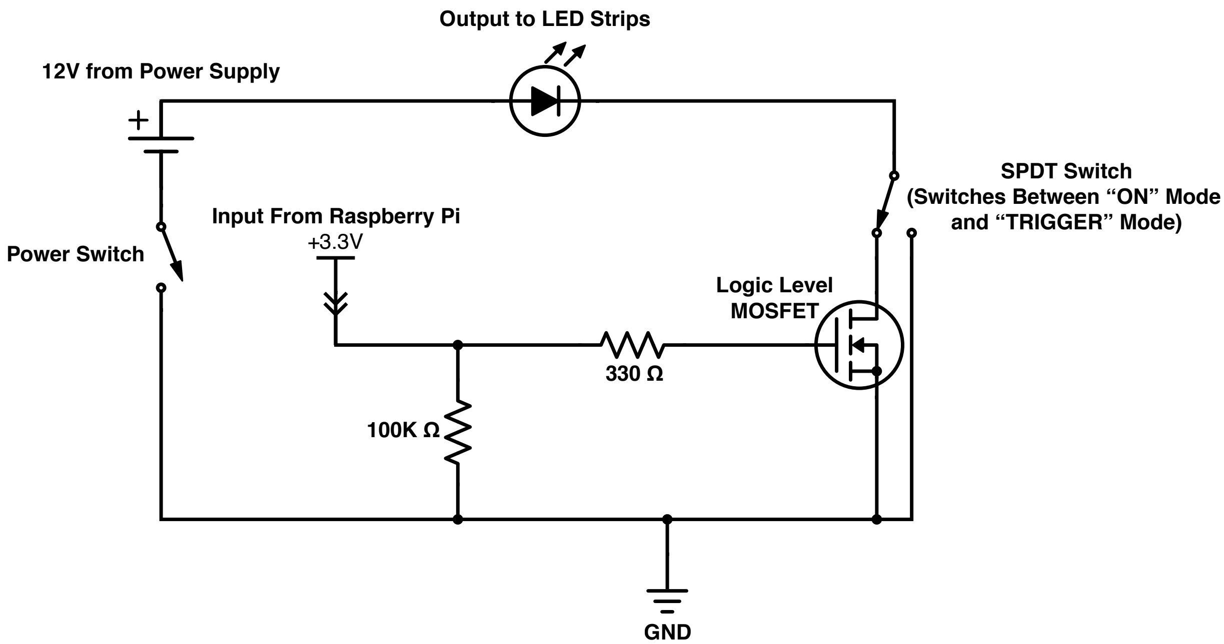
Quantity	Unit Price		Vendor	Part Number	Part Name	Assembly Guide Annotation
				Chassis		
1	$	5.00	Local Lumber Supplier		1/8" Medium Density Fiberboard, 1000 sq in. total	
0.083333333	$	7.84	McMaster-Carr	85635K421	Optically Fluorescent Cast Acrylic, 1/8" Thick, 12" x 12", Amber (Single Sheet enough for 12)	
0.25	$	8.63	McMaster-Carr	8560K239	Optically Clear Cast Acrylic Sheet, 1/8" Thick, 12" x 12" (Single Sheet enough for 4)	
1	$	1.95	Jameco	71240	Binding Post Breadboard Screw Type Chassis Mount 1 Black 1 Red	**j**
1	$	0.99	Jameco	316014	Rocker Switch R13-66 Amp-B-02 Single Pole Single Throw 10 Amp On Off Black	**k**
1	$	1.49	Jameco	317236	Toggle Switch (On-On) Single Pole Double Throw Solder Lug 5 Amp 250 Volt AC 28 Volt Panel Mount	**m**
2	$	1.49	Jameco	1766180	3.5mm Right Angle Non-Threaded Single Stereo Audio Jack	**n**
1	$	1.45	Jameco	228494	Stereo 3.5mm Male Plug To 3.5mm Male Plug 6 Foot Cable	
8	$	0.47	Digikey	36-8714-ND	4-40 Aluminum Hex Standoff, 1/4" Length	**o**
8	$	0.06	Digikey	36-9300-ND	4-40 Pan Slotted Head Machine Screw, 1/4" Length	**p**
10	$	0.08	Digikey	36-4694-ND	4-40 Hex Nut	**q**
3	$	0.06	Digikey	36-9301-ND	4-40 Pan Slotted Head Machine Screw, 3/8" Length	**r**
2	$	1.51	Digikey	36-8412-ND	4-40 Aluminum Hex Standoff, 2" Length	**s**
4	$	0.16	Digikey	36-9556-ND	2-56 Flat Slotted Machine Screw , 1/2" Length	**t**
4	$	0.04	Digikey	H212-ND	2-56 Hex Nut	**u**
4	$	0.16	Digikey	492-1074-ND	Round Spacer, 1/4" Length	**v**
2	$	19.99	Oznium	Super Thin Ribbon LED Strips	Super Thin Ribbon LED Strips	**w**
SUBTOTAL	**$**	**66.31**				
TOTAL	**$**	**244.45**				

Misc/Extras		
Thorlabs	BK5	Black Rubberized Fabric, 5' x 9' (1.5 m x 2.7 m) x 0.005" (0.12 mm) Thick
Thorlabs	T743-1.0	High-Performance Black Masking Tape, 1" x 180' (25 mm x 55 m) Roll
		22 AWG Wire
		Banana Plugs and Cables
		Flat Black Spray Paint
		USB Mouse for Raspberry Pi
		USB Keyboard for Rasperry Pi
		HDMI Cable to Connect Pi to Monitor
		HDMI-capable monitor

Tools		
Thorlabs	SPW606	SM1 Spanner Wrench, Length = 1"
		Wire Cutters/Stripper
		Molex Crimp Tool (Optional)
		Soldering Iron and Solder
		Hot Glue Gun
		Digital Multimeter
		Precision Screw Driver
		Scissors/Razer Blade

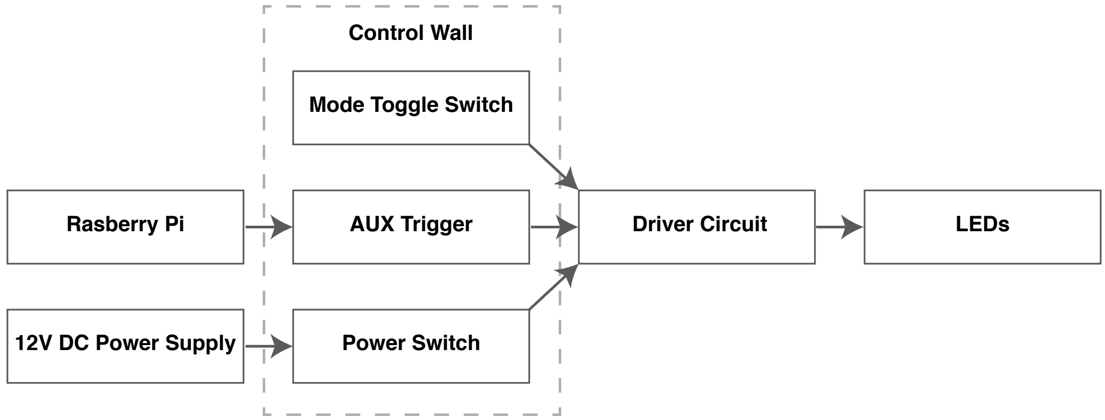
Circuit Diagram

Output from Raspberry Pi GPIO pin is placed on the gate of a Logic-Level n-channel MOSFET in enhancement mode such that a signal from the Pi connects the LEDs to ground and turns on the device. An SPDT switch enables bypassing of the MOSFET in cases where the LEDs must be turned on manually. Resistor values were chosen to limit current to the Raspberry Pi in case of accidentally setting the GPIO pin as an input.

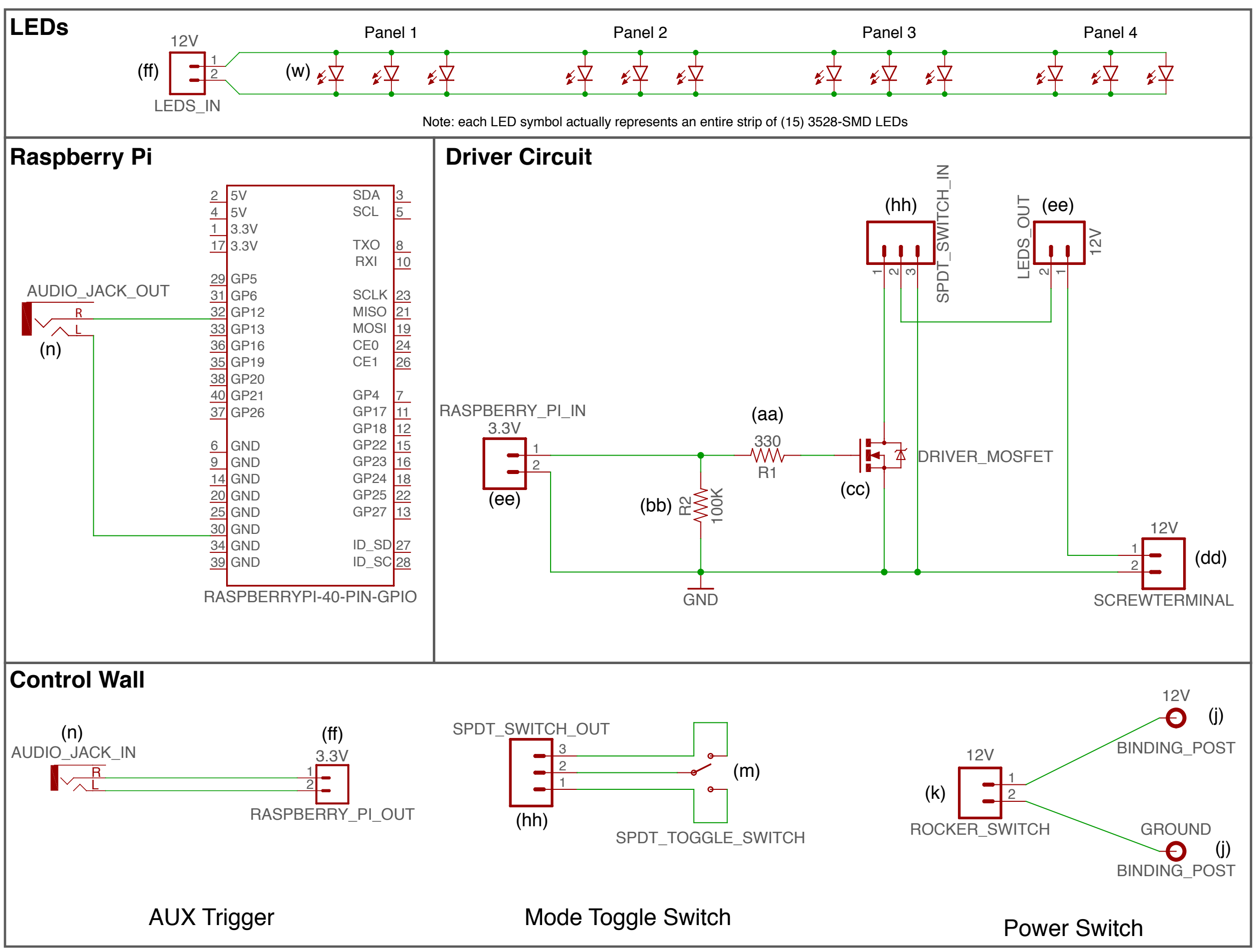


WIRING SCHEMATIC

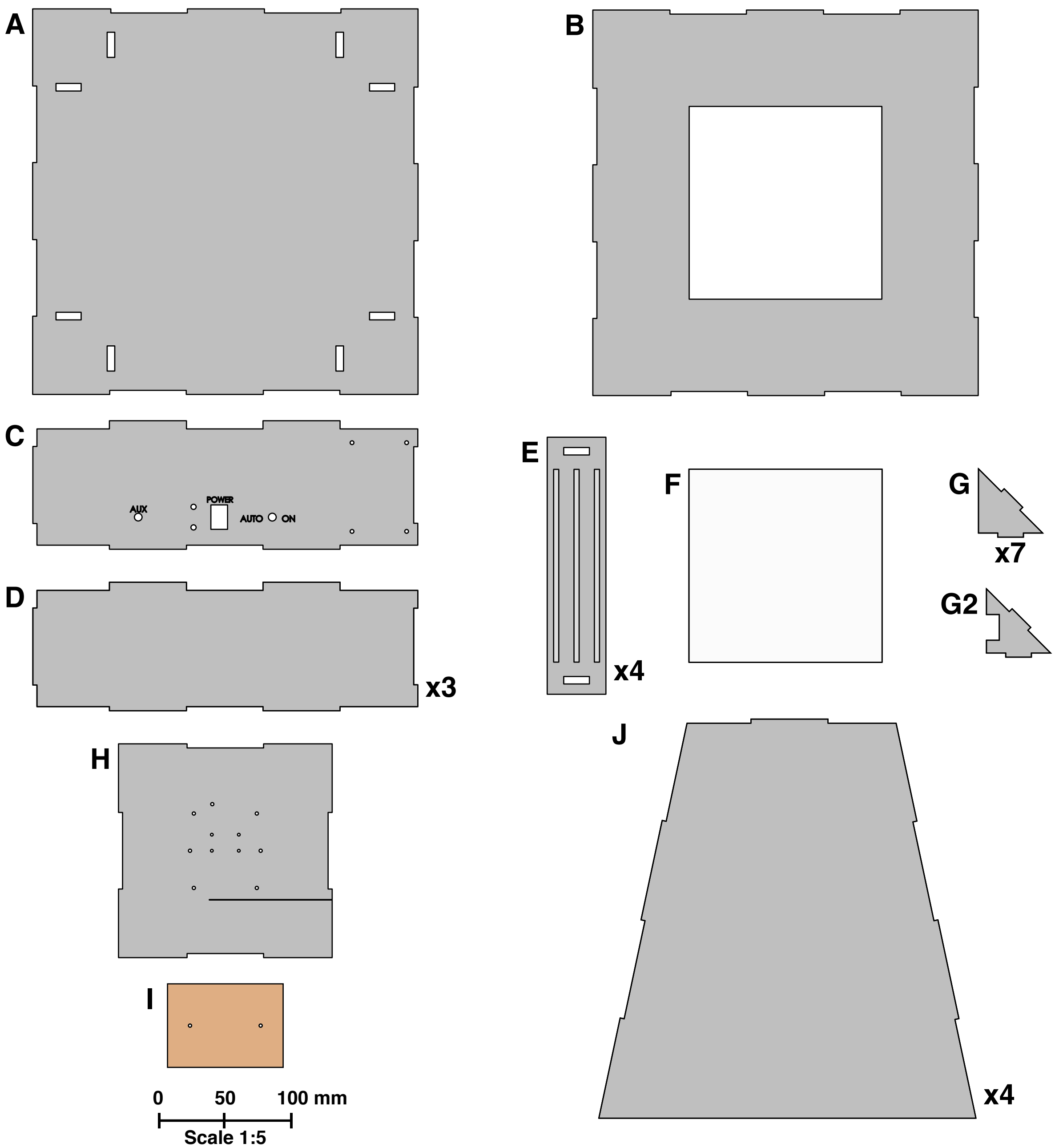
The below block diagram shows the logical electrical connections used within the automated imaging acquisition system. Both the Raspberry Pi and the 12V DC Power supply are connected to the Driver Circuit through the components mounted onto the Control Wall (See Assembly Instructions, page 13).



The below diagram shows the wiring connections within each module that is depiced in the above block diagram. Letters in parentheses indicate part numbers as labeled on the Parts List (pages 7-9)



Laser Cut Parts

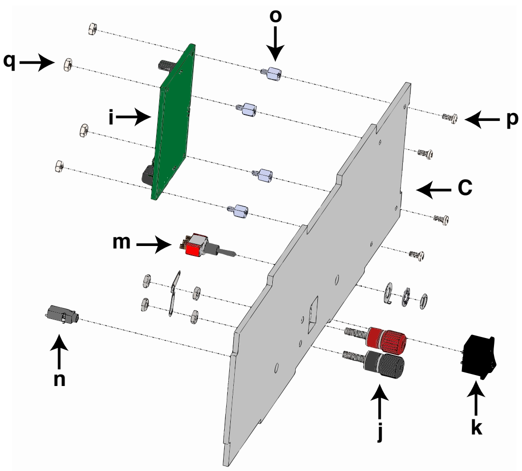


Letter	Description	File Name
A	Bottom of the Transilluminator Base	Base_Bottom
B	Top of the Transilluminator Base	Base_Top
C	Control Wall with Panel Mounted Electronics	Base_ControlWall
D	Wall of Transilluminator Base	Base_Wall
E	LED Panels	Base_LEDPanel
F	Clear Acrylic Window	Base_ClearWindow
G	90° Mount for Light Panel	Base_PanelMount
G2	90° Mount for Light Panel, used on Control Wall	Base_PanelMountControls
H	Top of Imaging Hood	Hood_Top
I	Amber Acrylic Filter	Hood_AmberFilter
J	Wall of Imaging Hood	Hood_Wall

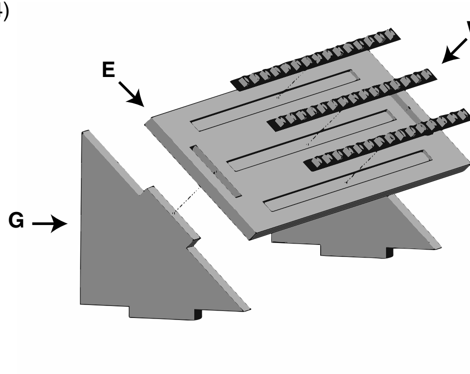
All Components are laser cut from 1/8” MDF except for G and G2, which are laser cut from 1/4” MDF

Assembly Instructions

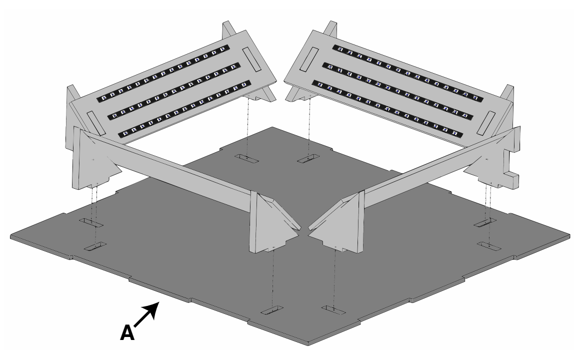
1 Assemble the Control Panel (**C**) with the electrical components (**m**, **n**, **o**, **p**, **q**, **j**, **k**) and driver circuit board (**i**, see Wiring Schematic on page 11).



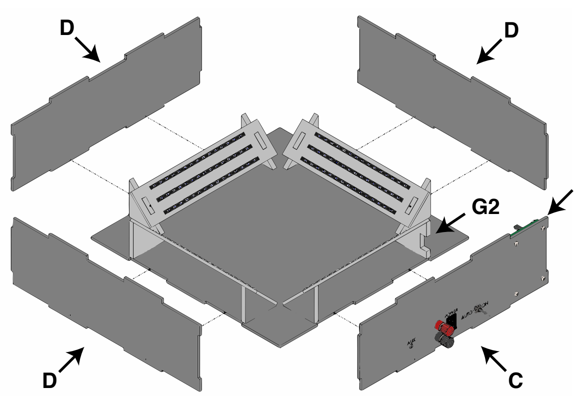
2 Attach the LED strip segments (**w**) onto the grooved panels (**E**) using the adhesive backing. Press-fit the panel onto the triangular stands (**G**). Repeat this for all 4 panels, using part **G2** as the right stand for one of the panels. (See Step 4)



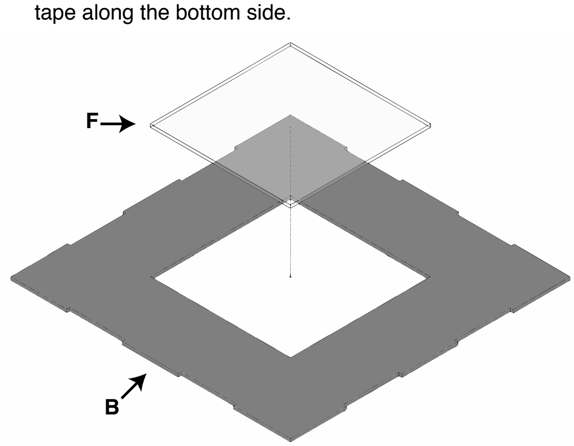
3 Press-fit the 4 parts made in Step 2 into the bottom of the base (**A**).



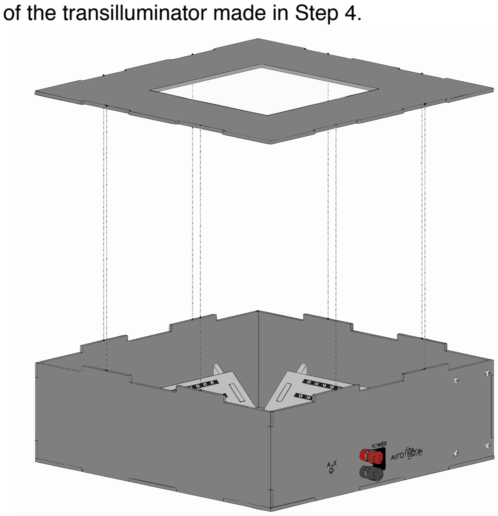
4 Press-fit the walls of the transilluminator (**C**,**D**) onto the base made in Step 3. The notched triangular stand (**G2**) is located adjacent to the driver circuit board (**i**). Connect and secure the wiring as shown in the Wiring Schematic (page 11) by passing wires through the notch of **G2**.



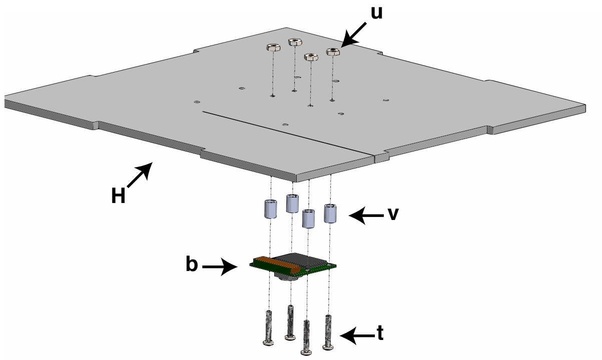
5 Press-fit the clear piece of acrylic (**F**) into the top piece of the base (**B**). Reinforce the seal with either epoxy, glue, or tape along the bottom side.



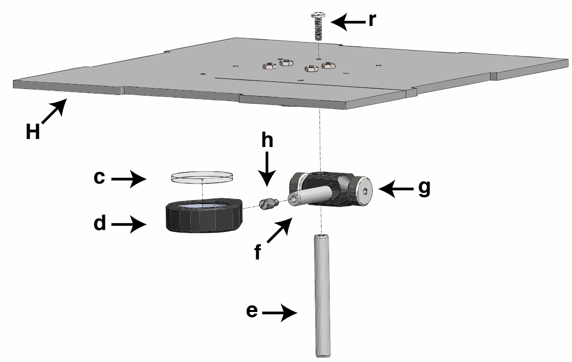
6 Press-fit the top of the base made in Step 5 onto the body of the transilluminator made in Step 4.



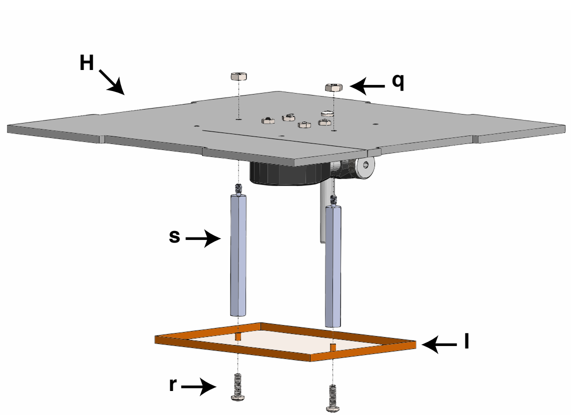
7 Screw the Raspberry Pi Camera Module (**b**) into the top panel (**H**) using 2-56 screws (**t**) and hex nuts (**u**). Use small spacers (**v**) between the camera and the panel.



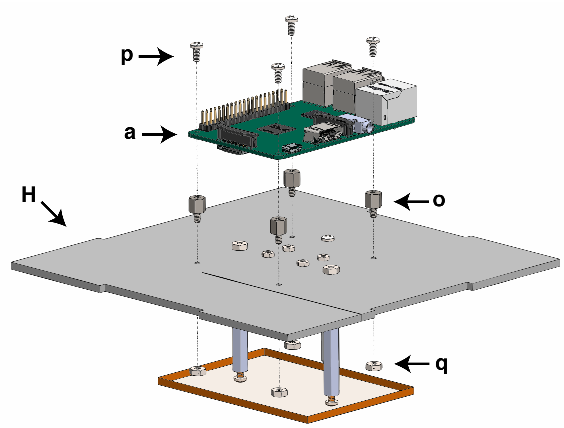
8 Set up the optics to enable focal adjustment of the camera. Screw the lens (**c**) into the lens mount (**d**) and secure using a spanner wrench and retaining ring (included with the lens mount). Use the thread adapter (**h**) to connect the lens mount to the 1” mounting post (**f**). Use the right-angle post clamp (**g**) to connect the lens to the panel (**H**) using the 2” mounting post (**e**) and screw (**r**).



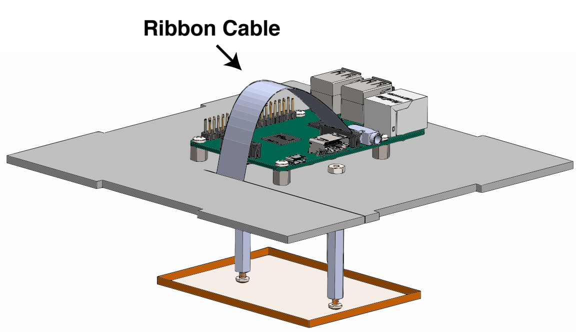
9 Connect the amber acrylic filter (**l**) to the panel (**H**) using 4-40 standoffs (**s**), screws (**r**), and hex nuts (**q**).



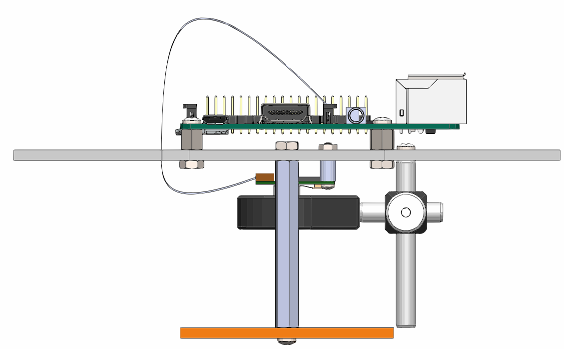
10 Mount the Raspberry Pi (**a**) to the top panel (**H**) using 4-40 standoffs (**b**), screws (**p**), and hex nuts (**q**).



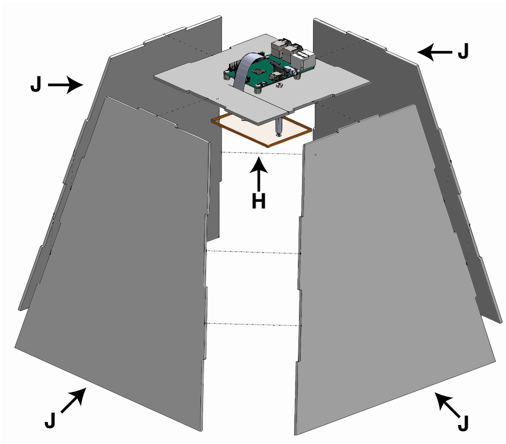
11 Feed the Camera Ribbon Cable (included with camera) through the relief slot and attach it to the Camera.



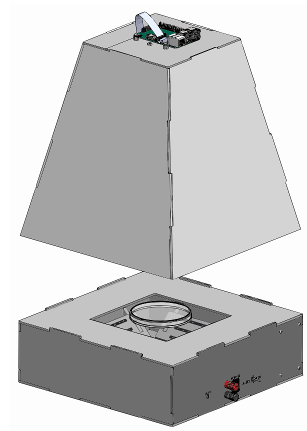
Side View of Completed Top Panel



12 Assemble the imaging hood by press fitting the trapezoidal sides (**J**) to the completed top panel (**H**). Stabilize the hood with an adhesive such as epoxy, hot glue, or tape.



13 Put plate onto the clear acrylic. Ensure that both transilluminator and Raspberry Pi are powered on and the Pi is connected to the internet. Close the lid and you are ready to run the experiment.



Fully Asseambled Transilluminator