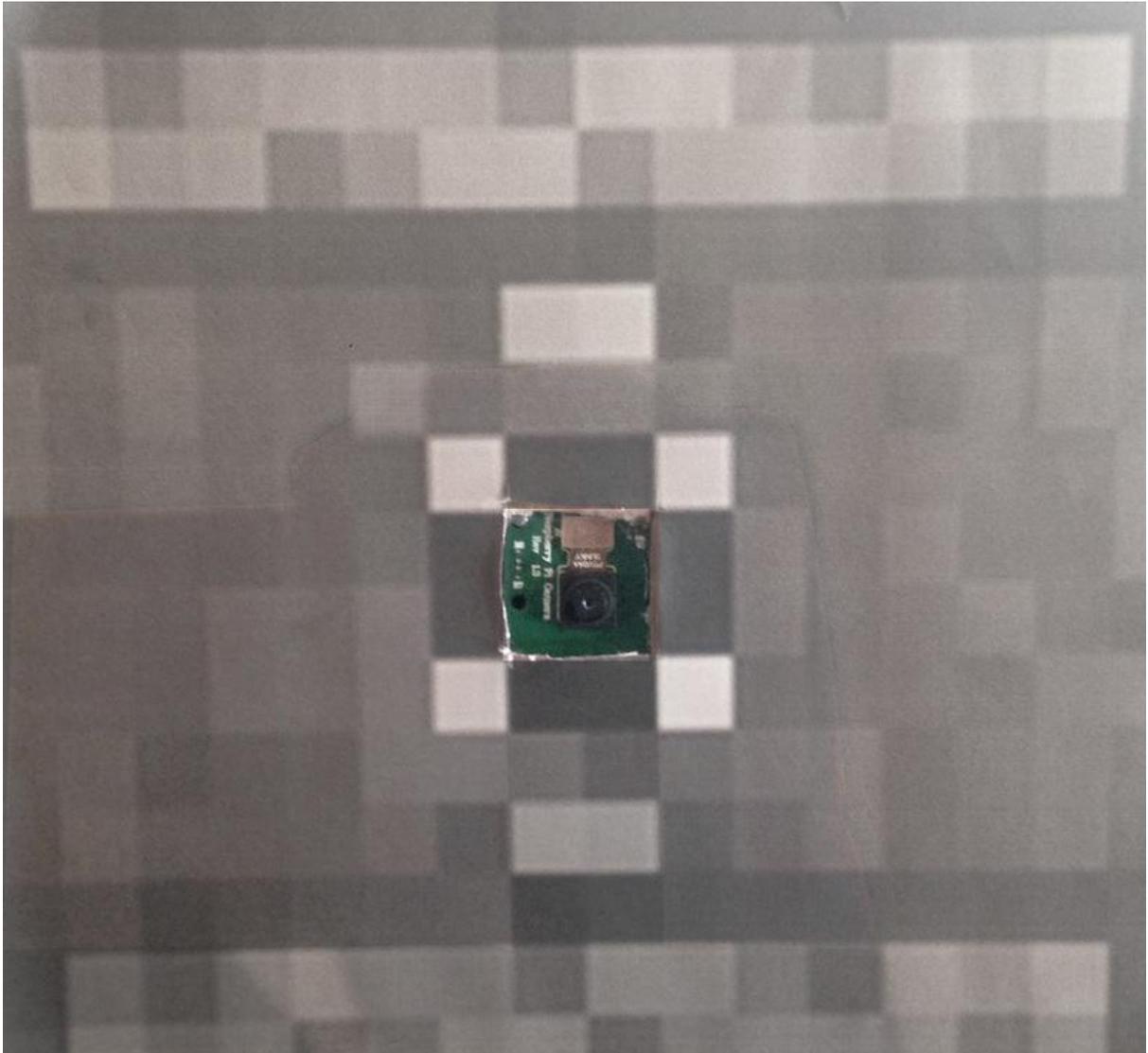




# Raspberry Pi SelfieBlock

Created by Thijs S



<https://learn.adafruit.com/selfieblock>

Last updated on 2022-12-01 03:10:20 PM EST

# Table of Contents

Introduction	3
Make the block	4
The Code	10
The Final Steps	18

---

# Introduction



We all know that Raspberry Pis are really cool computers, but did you know that you can use them to save you hundreds of dollars? In this guide, you will learn how to use a Raspberry Pi to make a selfie camera! This camera will take a picture at the press of a button, as well as upload the image to Dropbox. So let's get started! Here is a list of materials and supplies you will need:

- 
- |  |   |
|--|---|
| 1 x <a href="https://www.adafruit.com/product/3099">Raspberry Pi Camera Board v2 - 8 Megapixels</a><br>Snap, snap! The Camera v2 is the new official camera board released by the Raspberry Pi Foundation! | <a href="https://www.adafruit.com/product/3099">https://www.adafruit.com/product/3099</a> |
| 1 x <a href="https://www.adafruit.com/product/3055">Raspberry Pi 3 - Model B - ARMv8 with 1G RAM</a><br>Did you really think the Raspberry Pi would stop getting better...                                 | <a href="https://www.adafruit.com/product/3055">https://www.adafruit.com/product/3055</a> |
| 1 x <a href="https://www.adafruit.com/product/1119">Tactile Switch Button</a><br>Medium-sized clicky momentary switches are standard input "buttons" on electronic projects.                               | <a href="https://www.adafruit.com/product/1119">https://www.adafruit.com/product/1119</a> |
| 2 x <a href="https://www.adafruit.com/product/1919">Female-Female 2.54 to 2.0mm Jumper Wires</a><br>These jumper wires come with a twist!  | <a href="https://www.adafruit.com/product/1919">https://www.adafruit.com/product/1919</a> |
| 1 x <a href="https://www.adafruit.com/product/1566">USB Battery Pack for Raspberry Pi</a><br>Use this to make your selfie box portable!  | <a href="https://www.adafruit.com/product/1566">https://www.adafruit.com/product/1566</a> |
| 1 x <a href="https://www.adafruit.com/product/592">USB cable</a><br>Use this to connect your PI to your battery  | <a href="https://www.adafruit.com/product/592">https://www.adafruit.com/product/592</a>   |
-

Other Items:

1. Monitor for connecting to the Pi. VNC Viewer is also usable.
  2. Cardboard + Cutter
  3. Glue or Double Sided Sticky Tape :)
  4. Hot Glue Gun
- 

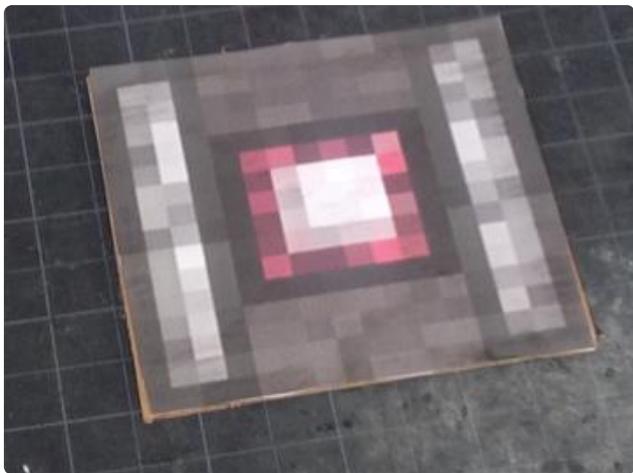
## Make the block

Now let's make the block! This block will hold the Pi and camera to take pictures!



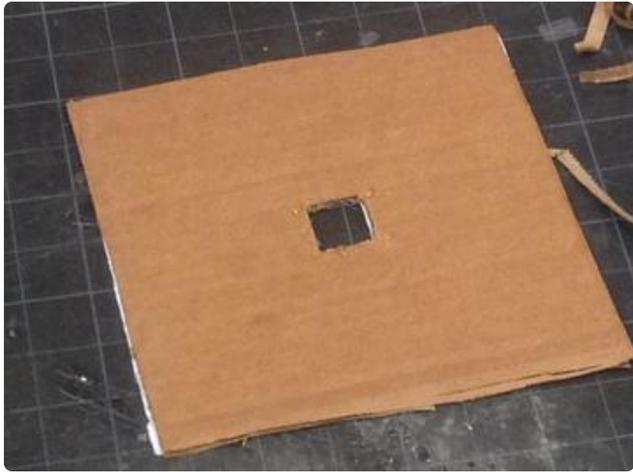
Cut 6 squares, measuring 7 x 7 inches each. These will be the sides.



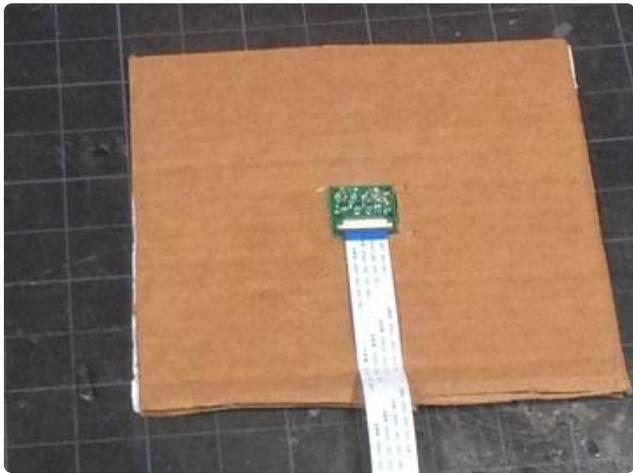


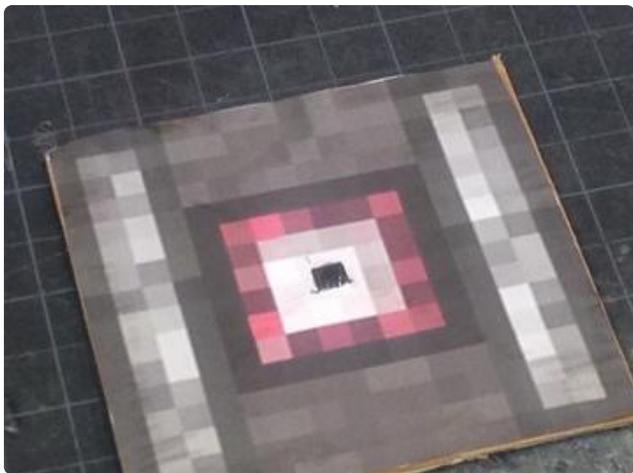
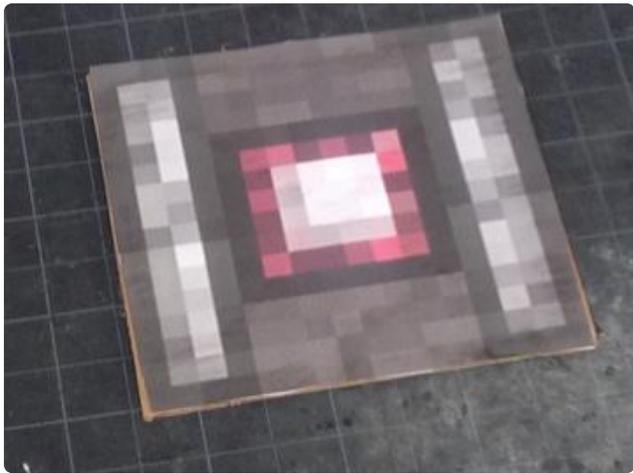
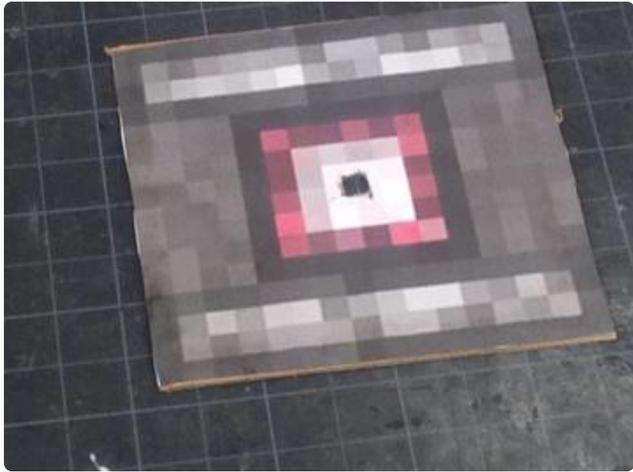
Download the observer side files and print them at their original size. Take the image with the red dot in the center (The observer side) and one of the cardboard squares, and cut out the center like shown on the left. Glue it on to the cardboard so there is a hole in the center for the camera. Glue the rest of the sides onto the other pieces of cardboard.

[Download Sides \(\)](#)



On the side that you just cut a hole for, put hot glue as shown in the picture on the left. Then, place the camera in the space.





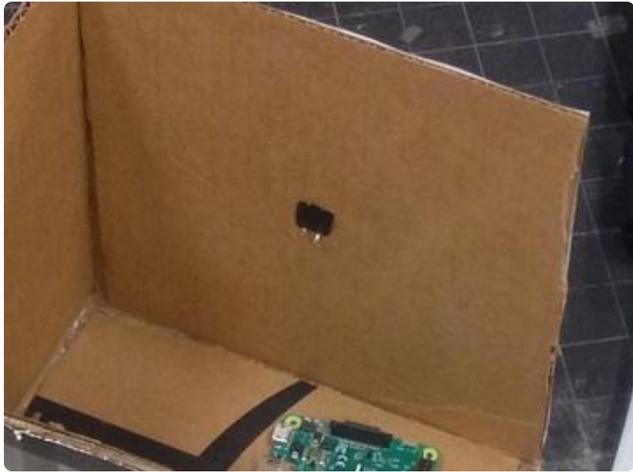
On the side is shown on the left, cut out a hole just the right size for the button. Then, glue the button to the hole. Make sure it's leads are facing out.



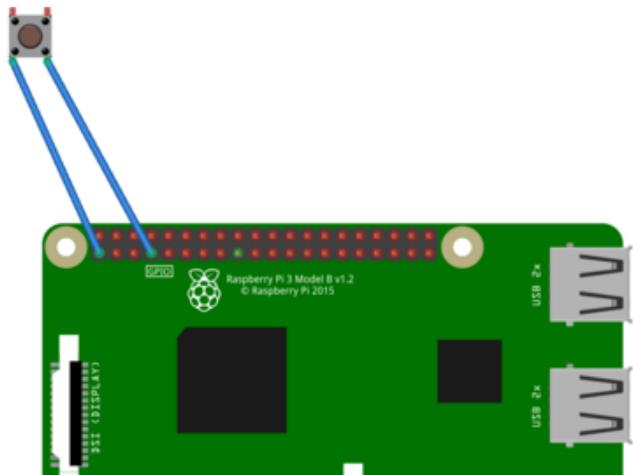
Glue the sides together to form a part of a cube. Connect the camera to the Pi.

1. On the top-right corner of the Pi, there is a black port. Pull the nubs on the sides of the port up.
2. Insert the ribbon cable with the leads facing down, as shown in the picture. Push it in until you feel it hit the bottom





Connect 2 female-female cables from 2 leads of the button, as shown in the picture.



Attach the female cables of the button to the Pi as shown on the right. Be sure your Pi is facing the same way as the one in the picture.

SelfieBlock.fzz

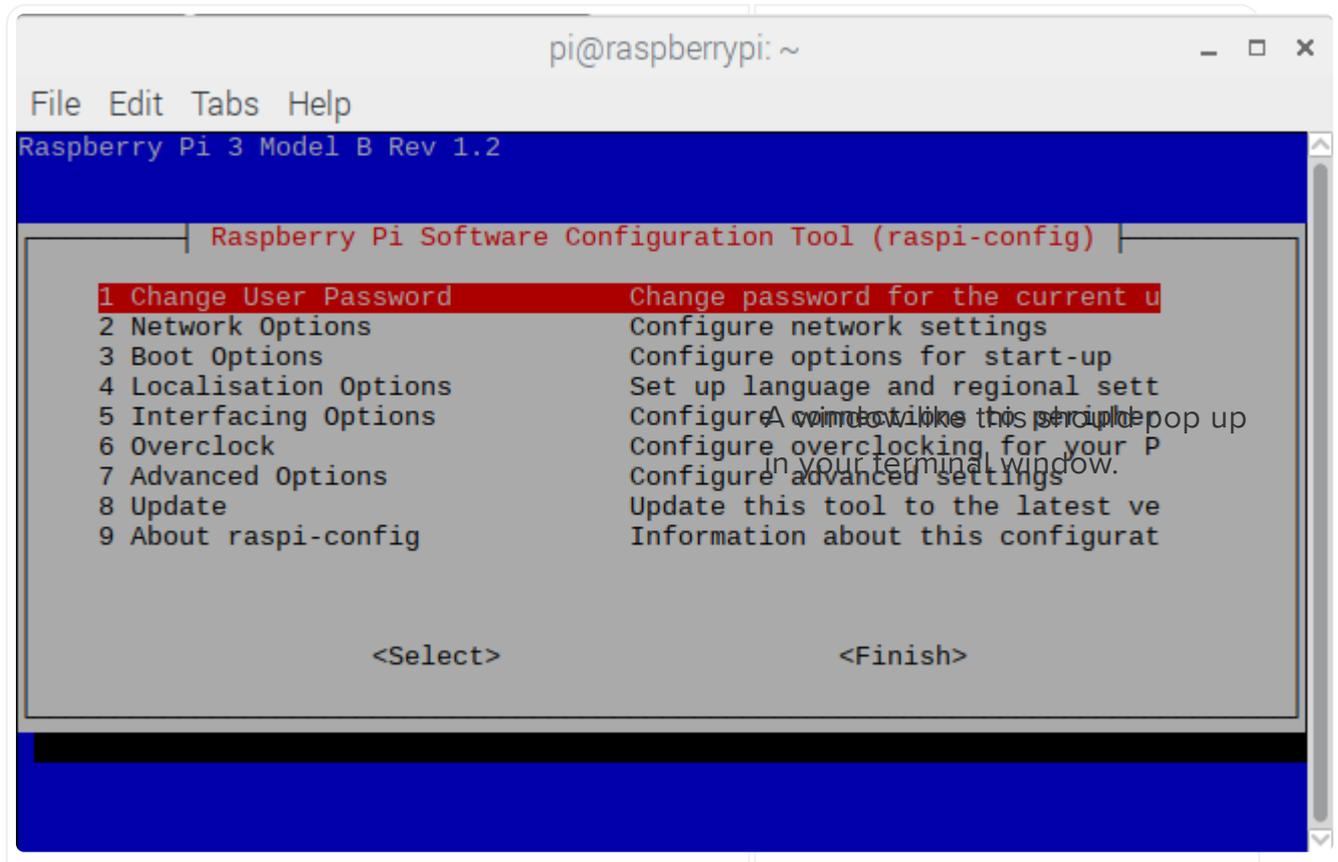
Great Job! Now let's work on the code!

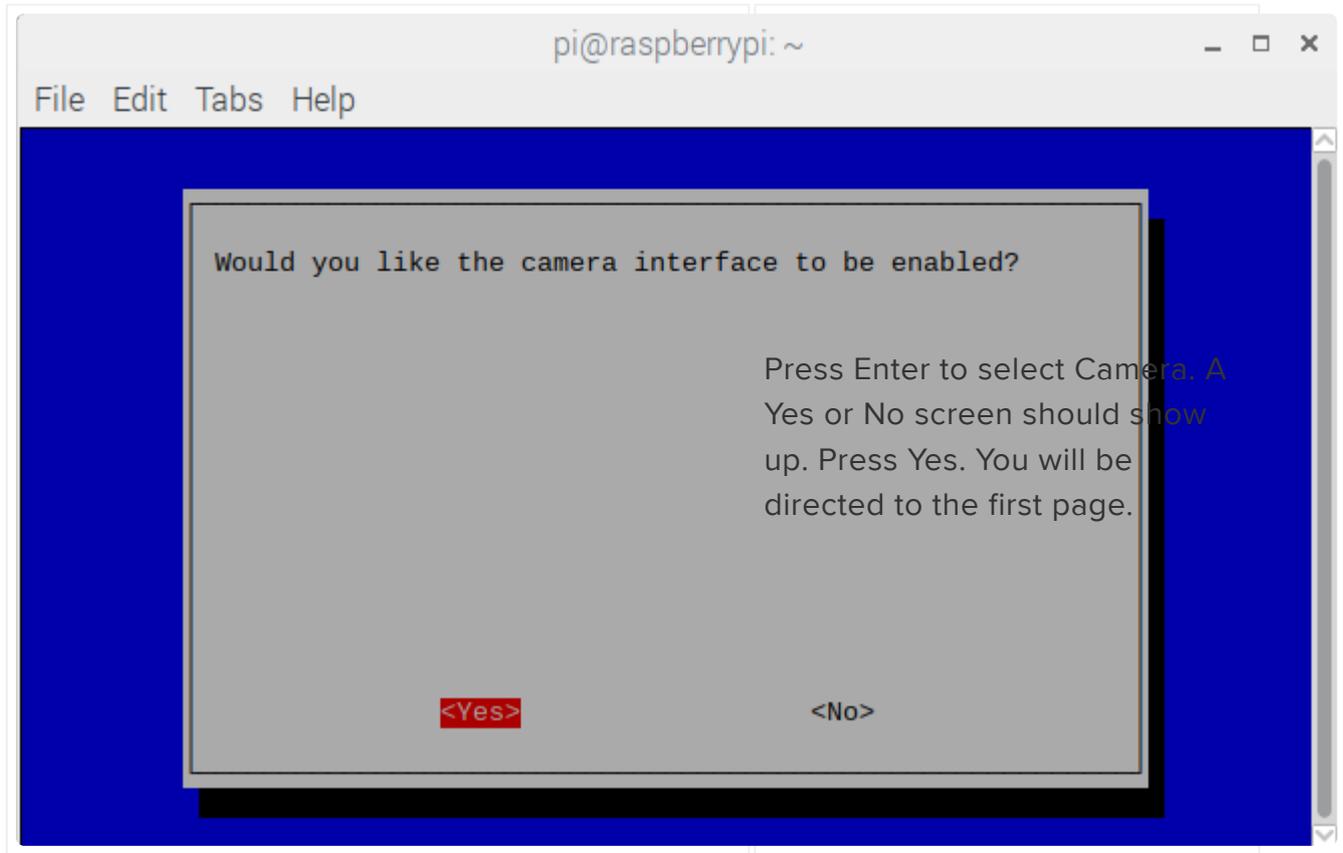
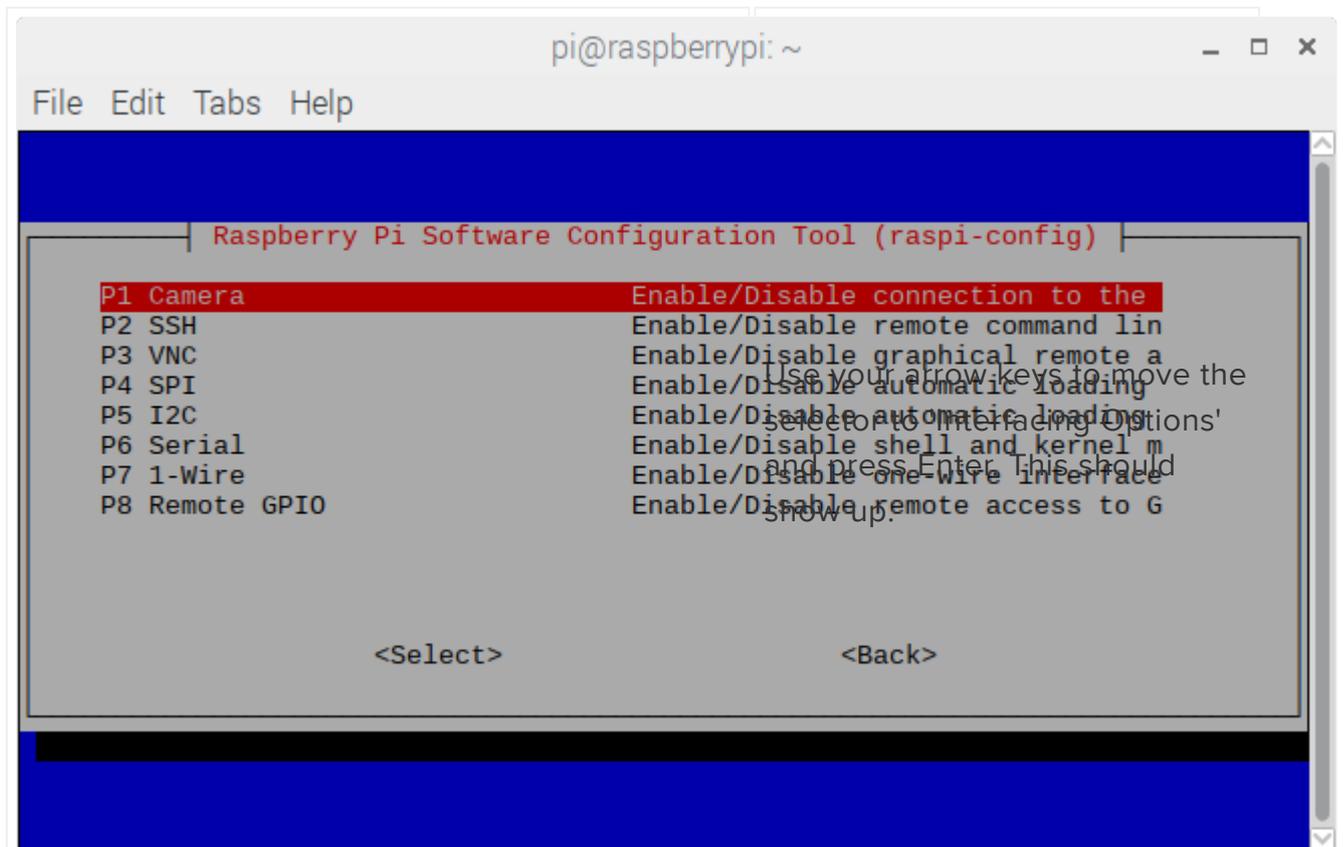
---

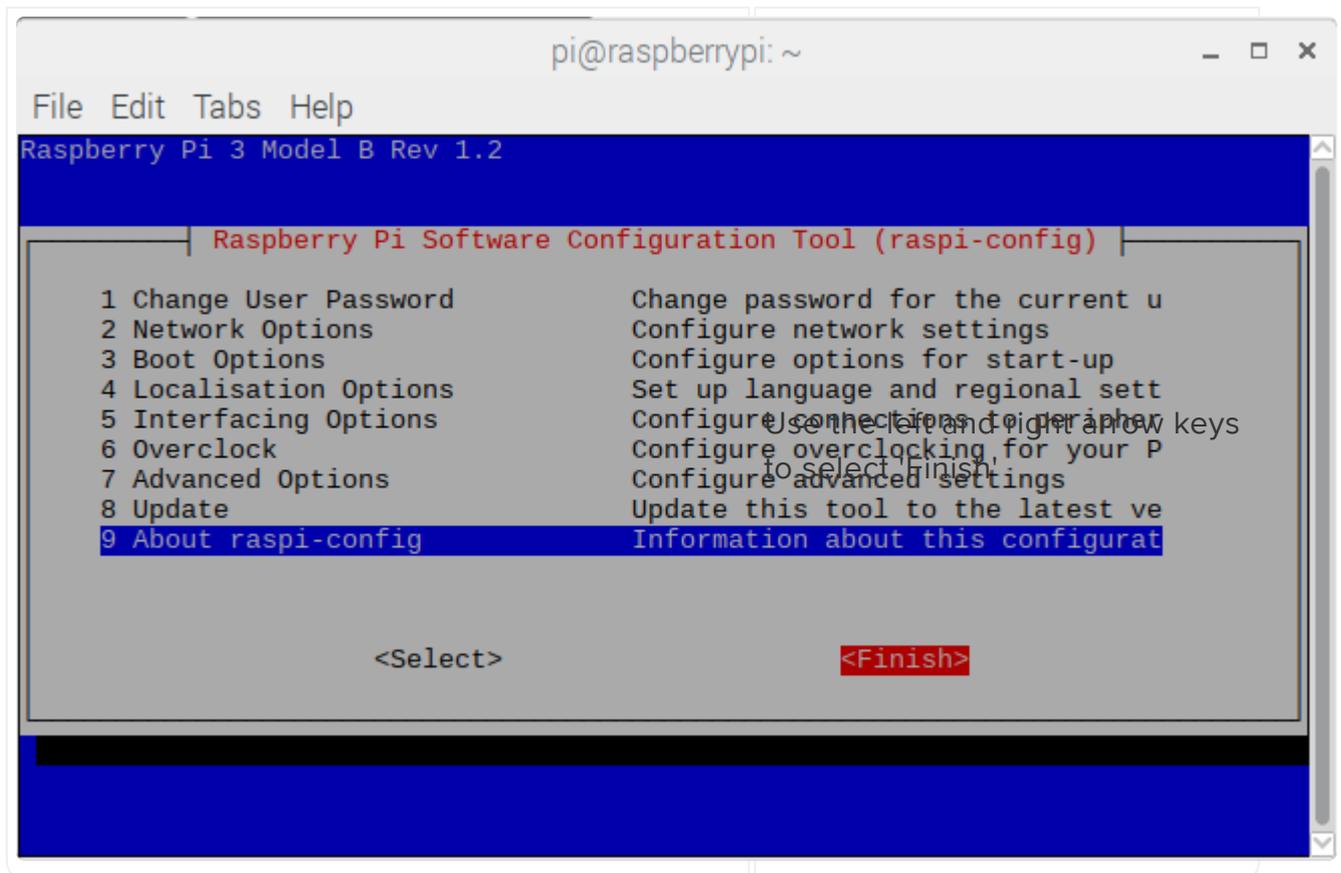
# The Code

Plug in and connect your pi to a display. Once it has started up, open terminal and run this command:

```
sudo raspi-config
```







Great! you have now enabled the camera function! Now let's setup the Dropbox uploader so we can receive our files. In terminal, run the following commands:

```
sudo apt-get update
sudo apt-get install -y curl
cd ~
git clone https://github.com/andreafrabrizi/Dropbox-Uploader.git
cd Dropbox-Uploader
chmod a+x dropbox_uploader.sh
sudo cp dropbox_uploader.sh /usr/local/bin/dropbox_uploader
```

Once you have that all installed, you should be able to run the script. Try it out:

```
dropbox_uploader
```

After you run that command, you should see the following:

```
Dropbox Uploader v1.0
Andrea Fabrizi - andrea.fabrizi@gmail.com

Usage: /usr/local/bin/dropbox_uploader [PARAMETERS] COMMAND...

Commands:
  upload <LOCAL_FILE/DIR ...> <REMOTE_FILE/DIR>;
  download <REMOTE_FILE/DIR> [LOCAL_FILE/DIR]
  delete <REMOTE_FILE/DIR>;
  move <REMOTE_FILE/DIR> <REMOTE_FILE/DIR>;
  copy <REMOTE_FILE/DIR> <REMOTE_FILE/DIR>;
```

```
mkdir    &lt;REMOTE_DIR&gt;
list     [REMOTE_DIR]
monitor  [REMOTE_DIR] [TIMEOUT]
share    &lt;REMOTE_FILE&gt;
saveurl  &lt;URL&gt; &lt;REMOTE_DIR&gt;
search   &lt;QUERY&gt;
info
space
unlink
```

Optional parameters:

```
-f &lt;FILENAME&gt; Load the configuration file from a specific file
-s           Skip already existing files when download/upload. Default:
```

Overwrite

```
-d           Enable DEBUG mode
-q           Quiet mode. Don't show messages
-h           Show file sizes in human readable format
-p           Show cURL progress meter
-k           Doesn't check for SSL certificates (insecure)
-x           Ignores/excludes directories or files from syncing. -x filename -
x directoryname. example: -x .git
```

For more info and examples, please see the README file.

If you don't get this message, please check that you have followed the steps before carefully.

Now that you have the command set up, it's time to connect your dropbox account. Run the following command to set it up.

```
dropbox_uploader upload
```

It will realize that there isn't an account associated with it, and it will start the setup. You should see something like this:

This is the first time you run this script.

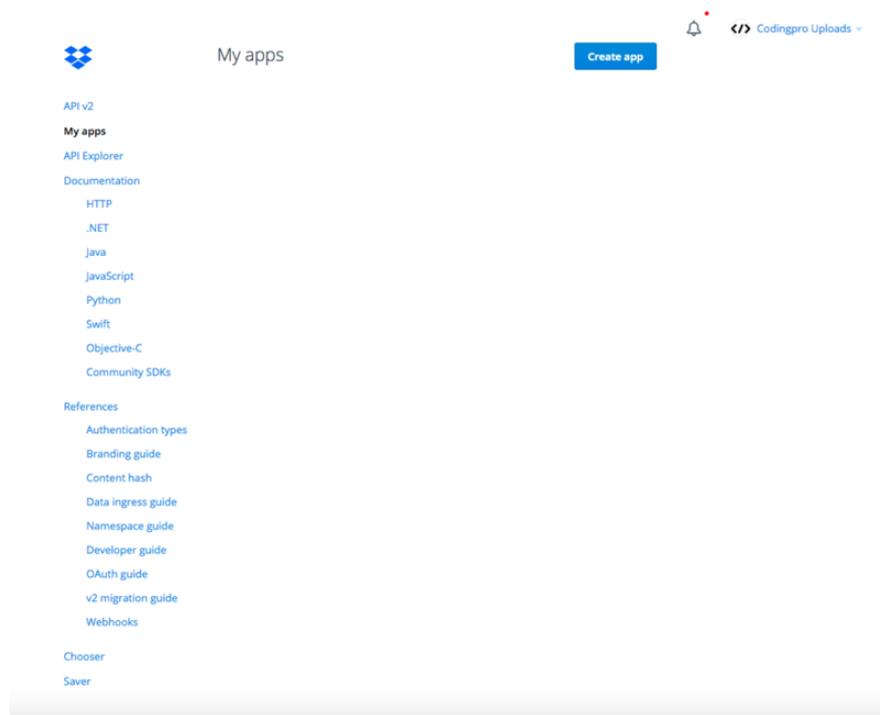
- 1) Open the following URL in your Browser, and log in using your account: <https://www.dropbox.com/developers/apps>
- 2) Click on "Create App", then select "Dropbox API app"
- 3) Now go on with the configuration, choosing the app permissions and access restrictions to your DropBox folder
- 4) Enter the "App Name" that you prefer (e.g. MyUploader6293443213380)

Now, click on the "Create App" button.

when your new App is successfully created, please type the App Key, App Secret and the Permission type shown in the confirmation page:

```
# App key:
```

Follow the instructions and open to the [https://www.dropbox.com/developers/apps \(\)](https://www.dropbox.com/developers/apps) URL. To make it easier, open the URL on your Pi. Sign in if you haven't already. The page should look like this:



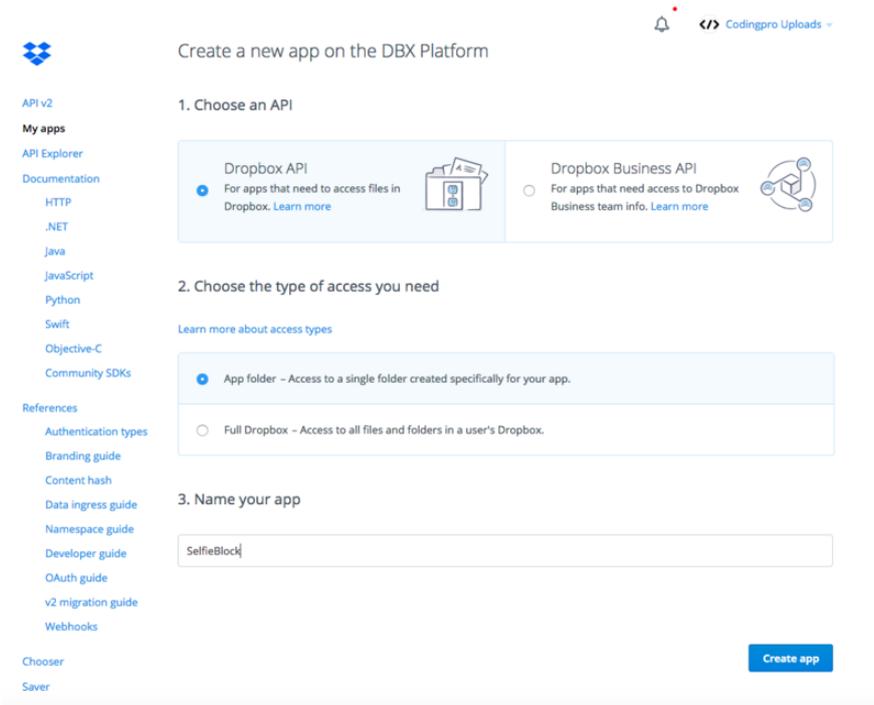
Click 'Create app' and choose the following options:

Choose API: Dropbox API

Choose the type of access you need: App folder

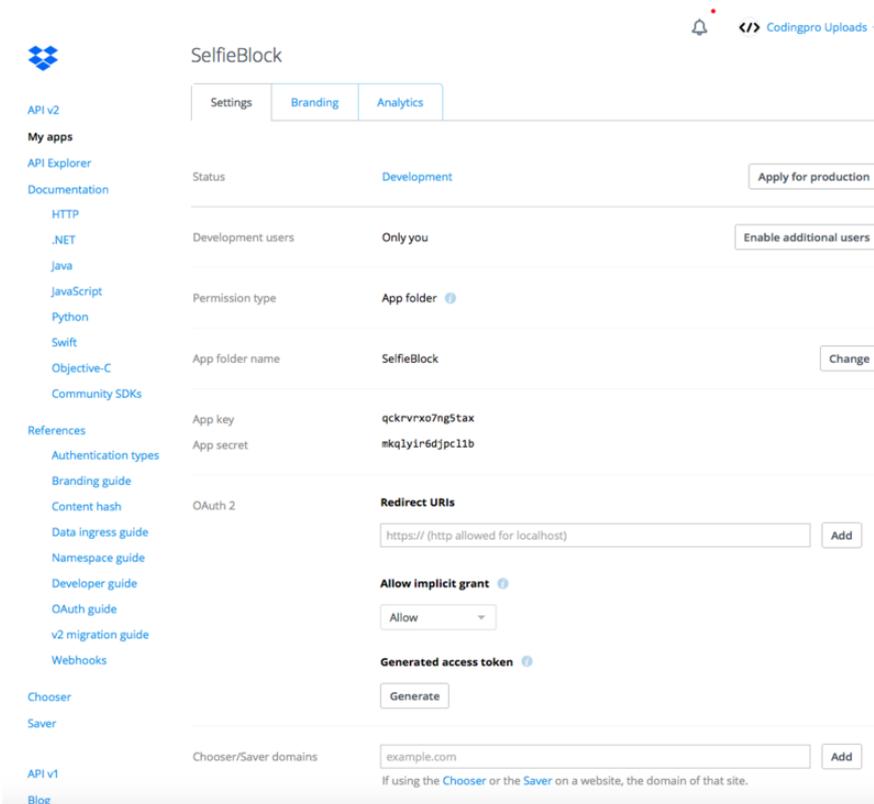
Name your app: Anything you want!

Press 'Create app' again.



You should be directed to a page like this:

Please Note: These Keys are fake.



On this page, you will need the App Key and the App Secret. First, copy the app key and paste it into the terminal running the setup. Press Enter. Then, click show on the

App Secret and copy that code. Paste it in and press enter again. This is what you should see:

```
pi@raspberrypi ~/Dropbox-Uploader $ dropbox_uploader upload
This is the first time you run this script.

1) Open the following URL in your Browser, and log in using your account: https://
www.dropbox.com/developers/apps

2) Click on "Create App", then select "Dropbox API app"

3) Now go on with the configuration, choosing the app permissions and access
restrictions

to your DropBox folder
4) Enter the "App Name" that you prefer (e.g. MyUploader6293443213380)

Now, click on the "Create App" button.

when your new App is successfully created, please type the
App Key, App Secret and the Permission type shown in the confirmation page:

# App key: qckrvrx07ng5tax
# App secret: mkqlyir6djpclib

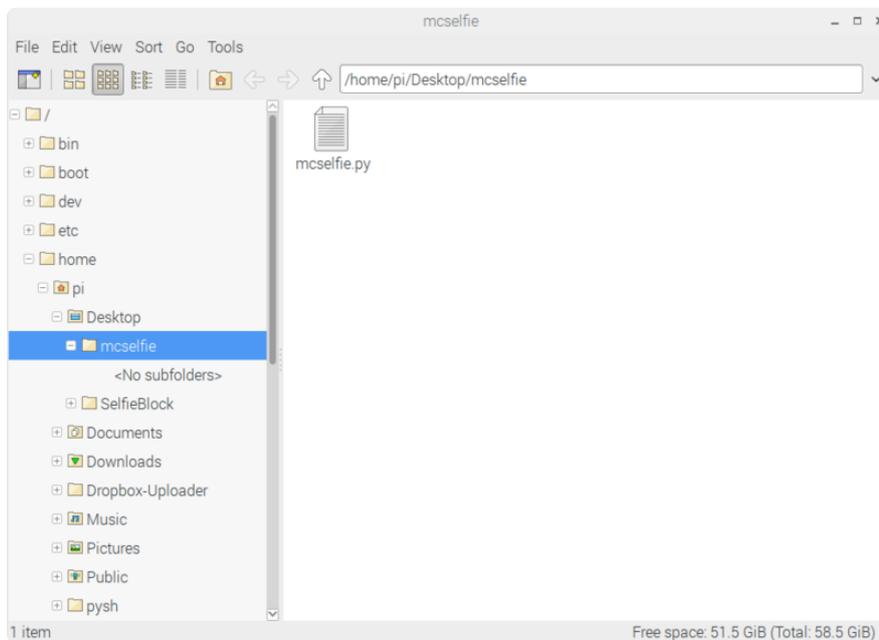
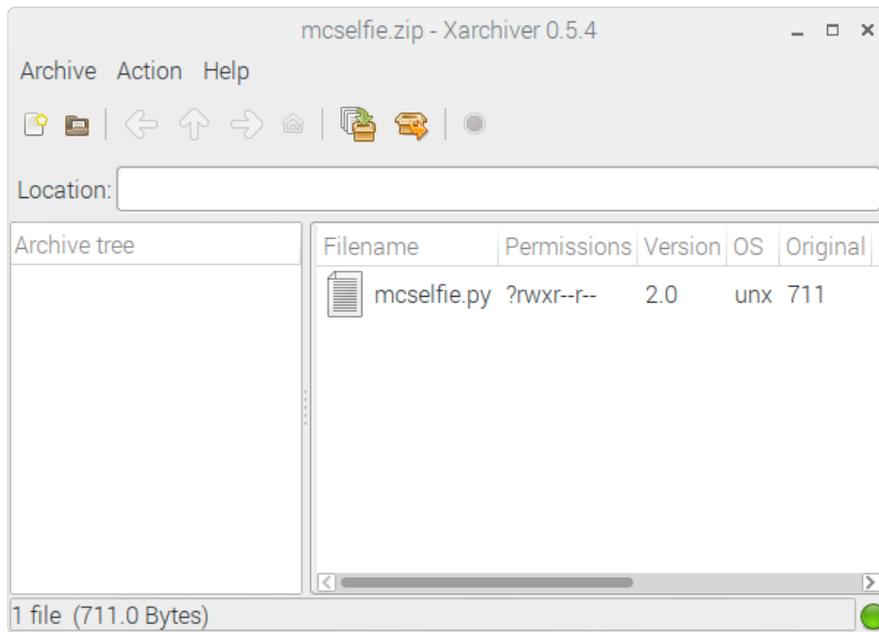
Permission type:
App folder [a]: If you choose that the app only needs access to files it creates
Full Dropbox [f]: If you choose that the app needs access to files already on
Dropbox

# Permission type [a/f]: f

> App key is qckrvrx07ng5tax, App secret is mkqlyir6djpclib and Access level is
Full Dro
box. Looks ok? [y/n]: y
```

If the key looks ok, press y and then press enter. It will ask you to go to a URL. Copy the URL and paste them into your browser. It may ask you which dropbox to connect to if you have multiple, and then say Success! Press enter in terminal and you have now connected your account!

Now let's get the python code running! On the Pi, make a folder on the desktop called mcselfie. It must be the exact name so that the code can run. Open a new tab in your Pi's web browser and go to this shortlink: [adafru.it/selfiecode](https://adafru.it/selfiecode). ()Extract the .zip file and drag the .py file into the mcselfie folder on your desktop.



Open up the python file in Thonny. You should see the following code:

```

from picamera import PiCamera
from gpiozero import Button
from time import sleep
import time, os
button = Button(4, pull_up=False,)
while True:
    button.wait_for_press()
    name = (time.strftime("%y-%b-%d_%H:%M:%S"))
    fname = "/home/pi/Desktop/mcselfie/" + "Selfie " + name + ".jpg"
    cmd = "/home/pi/Dropbox-Uploader/dropbox_uploader.sh upload /home/pi/Desktop/
mcselfie " + "/Selfie_" + name
    command = "/home/pi/Dropbox-Uploader/dropbox_uploader.sh upload" + fname + "/"
mcselfie"

    with PiCamera() as camera:
        camera.resolution = (1500,1080)
        camera.start_preview()

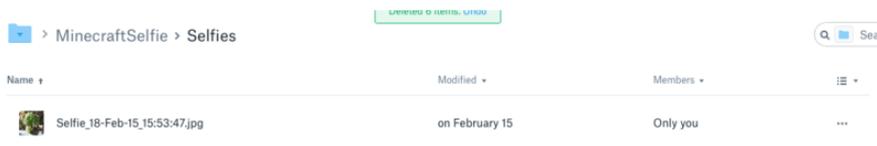
```

```
sleep(3)
camera.stop_preview()
camera.capture(fname)
sleep(1)
os.system(cmd)
```

Press the green 'Play' button to run the code. You should see the following in the console:

```
&gt;&gt;&gt; %Run mcselfie.py
```

Press the button to take a picture. The red light on the camera should turn on, and then turn off once it has finished uploading. You have 3 seconds to pose for the picture. Go to your dropbox, click apps, and click the app name. Inside the app files should be a folder called Selfies. Open it up and your selfies should show up.



If the camera light doesn't turn on, check that the button is connected correctly. If you get an error in the console when you press the button, make sure that the camera is connected, and restart the Pi and run the file again. If you still get the error, ping me in the discord server.

To make the camera portable, unplug the power and connect a portable battery to the Pi. Note that every time to restart the Pi, you have to open and run the mcselfie.py file again.

---

## The Final Steps

Glue the side of the Observer on with more Hot Glue. To give easy access to the inside, tape the top onto the box instead of glue (Shown below).



And you are done! Invite some friends over and fascinate them with your own selfie machine!



---

I get a Camera Not Enabled error but I enabled the camera!

Make sure you enabled the camera and that it is plugged in. If the problem persists, restart your Pi.

---

## Nothing happens when I press the button!

Make sure that the button is plugged in correctly, and that the script is running.

---

## I have another problem not listed here!

Ping me on the Adafruit discord server.