

How to Make a DIY HomeKit Smart Garage Door Opener with a Raspberry Pi

Open Your Garage Door with Your iPhone

2018.03.07

Auto-install: HomeKit/Siri Smart Garage Door Opener with a Raspberry Pi and H...



How to Make a DIY HomeKit Smart Garage Door Opener with a Raspberry Pi
Open Your Garage Door with Your iPhone!

Introduction

This video shows step-by-step how to make a DIY HomeKit Smart Garage Door Opener with a Raspberry Pi. It provides a central location for a comprehensive list of the minimum steps necessary to get up and running with a tested, working configuration.

Although a Mac computer is used in these steps, the equivalent Windows and Linux utilities can be used but are not documented here.

Part 1: Parts list

Part 2: Download the operating system

Part 3: Download and install Etcher

Part 4: Configure the camera

Part 5: Prepare the Raspberry Pi MicroSD card

Part 6: Boot the Raspberry Pi

Part 7: Remotely log into the Raspberry Pi

Part 8: Run the Geek Slant Smart Garage installation script

Part 9: LEGO case overview

Part 10: Connect the wires

Part 11: Install in the garage

Part 12: Add to Apple Home

Conclusion

Part 1: Parts list

Minimally, a Raspberry Pi 3, a HiKam S6 camera, and a magnetic switch are needed to complete all the steps as documented in this video.

For the complete parts list with links, see <http://smartgarage.geekslant.com> (<http://smartgarage.geekslant.com>).

Part 2: Download the operating system

- On your computer, open your web browser and go to <https://www.raspberrypi.org/> (<https://www.raspberrypi.org/>).
- Click on **DOWNLOADS**.
- Click on **RASPBIAN**.
- Click on **Download ZIP** under *RASPBIAN STRETCH LITE*. (Time: ~18 minutes. Even though I have gigabit fiber, this took a while because the download server was slow.)

While you wait for Raspbian to download, continue to step 3 to download and install Etcher.

Part 3: Download and install Etcher

- Go to <https://etcher.io/> (<https://etcher.io/>).

- Click on **Download for macOS**. (This was quick for me, taking only a few seconds.)
- Open Finder and go to the **Downloads** folder.
- Double click the downloaded Etcher file.
- Drag the Etcher icon to the Applications folder icon.

Part 4: Configure the camera

- Plug in the Micro USB cable into the camera to boot it up.
- On your iPhone, open the App Store and install the HiKam camera app.
- Open the HiKam app, and register a new account.
- Log in with your new account.
- With the camera booted up and beeping, reboot the camera into AP Mode using the supplied reset pin. Hold down the reset button until you hear a chime and a loud beep, and wait for the camera to reboot.
- Connect your iPhone's WiFi directly to the camera. The prefix will be "Hi."
- Open the HiKam app. Because you're directly connected to the camera via WiFi, wait for the app to autodetect the camera. It might take a few seconds.
- Tap on **Discover 1 New Camera**.
- Tap your camera.
- Enter a name for your camera, and enter the password printed on the back of your camera.
- You'll then be prompted to enter your own camera password.
- Tap **Settings**, and then **Network Settings** to configure your camera for your WiFi network.
- After you enter your WiFi password, the camera will reboot and connect to your WiFi network.
- Once the camera is back online, tap **Settings** and then **Camera Update** to update the firmware.
- Once the firmware update is complete, the camera will reboot again.
- Once the camera is back online, tap **Settings** and then **Camera Management**, and turn on **RTSP authentication** to password protect your network stream.
- Go back, and tap **Date and Time**.
- Set the Timezone and then Date and Time.
- Go back and tap the **Camera Info** button at the top, and write down the IP Address.

Part 5: Prepare the Raspberry Pi MicroSD card

Wait for the Raspbian download to complete before proceeding.

Flash the MicroSD card

- Open Etcher.
- Click on **Select Image**.
- Click on the downloaded Raspbian file and then **Open**.

- With the MicroSD card plugged in, click on **Flash!**. (Time: ~4 minutes.)

Enable SSH

- Created an empty SSH file:
 - Open Terminal.
 - Enter:

```
touch ssh
```

- Flashing the MicroSD card will automatically eject the card and remove it from Finder. Re-insert the card, which will show up as “boot,” into the Mac to add it back to Finder.
- Copy the ssh file to the MicroSD card.
- Eject the MicroSD card from the Mac and insert it into the Raspberry Pi.

Part 6: Boot the Raspberry Pi

- Connect the Raspberry Pi to the network with an Ethernet cable. (The Raspberry Pi has a built in WiFi adapter, but using that is not documented here.)
- Plug the Micro USB power adapter into the Raspberry Pi.
- Wait about 1 minute for the boot process to finish. If you wish to view the boot process, plug an HDMI cable into the Raspberry Pi and attach to a monitor or TV, but this is not necessary as this will be a headless computer.

Part 7: Remotely log into the Raspberry Pi

- Open Terminal on the Mac.
- Enter:

```
ssh pi@raspberrypi.local
```

- When you connect with ssh for the first time, you will be prompted to accept the signature. Enter:

```
yes
```

- For the password, enter:

```
raspberry
```

- When you log in for the first time, you will be advised to change the password. Change the password by entering:

```
passwd
```

Part 8: Run the Geek Slant Smart

Garage installation script

The Geek Slant Smart Garage software repository automates the rest of the software installation and configuration. It updates the operating system, installs the required libraries, installs Homebridge, installs the garage door opener plugin for Homebridge, installs the camera plugin for Homebridge, and configures everything to be up and running ready to be added to the Apple Home app.

Install git onto the Raspberry Pi

- Enter:

```
sudo apt-get install -y git
```

Clone the Geek Slant Smart Garage git repository:

- Enter:

```
git clone https://github.com/geekslant/smartgarage
```

Execute the install-smartgarage.sh script

- Enter:

```
sudo ./smartgarage/install-smartgarage.sh
```

- Configure your garage door opener and camera.
- Enter your camera's password and IP address so that the installation script can automatically create the configuration file for you.

The script took 37 minutes to run on my Pi, so expect to wait a while.

Where to edit the configuration

- To edit your configuration, edit the config.json in the /var/lib/homebridge directory:

```
sudo nano /var/lib/homebridge/config.json
```

- Set the "doorOpensInSeconds" to match your garage door opener.
- If your camera password or IP address changes, or if you want to add a new camera, scroll down to the "cameras" section.

Part 9: LEGO case overview

A custom LEGO case was made for this project using old pieces saved from childhood. It's relatively easy to make a custom LEGO case for a Raspberry Pi's dimensions. Many Raspberry Pi

cases for sale exist as well.

My case has access to power, HDMI, and audio, as well as Ethernet and USB ports. There's holes in the side for wires, and an access door for the MicroSD card. There are pieces that stick out for mounting to a wall, and there's an access door on top for the GPIO pins for the Raspberry Pi.

Part 10: Connect the wires

(Please see the YouTube video for all of the diagrams.)

- Connect two wires to the relay using Normally Open 1 (NO1) and Common 1 (COM1). These two wires will go to the garage door opener.
- Use three wires to connect the relay to the Raspberry Pi:
 - Connect VCC to DC Power 5V (Pin 04)
 - Connect GND to Ground (Pin 06)
 - Connect IN1 to GPIO 05 (Pin 29)
- Connect two more wires to the Raspberry Pi, one to GPIO 20 (Pin 38), and the other to DC Power 3.3V (Pin 17). These two wires will go to the magnetic switch using the Common (COM) and Normally Open (NO) terminals.

Part 11: Install in the garage

- I mounted the case to the wall near the ceiling in my garage.
- The two wires from the relay connect to the garage door opener using the screw terminals used by the existing garage door opener wall button.
- The two wires from the Raspberry Pi go to the magnetic switch, with half of the switch mounted to the center channel, and the other half mounted to the carriage slide, with the two sides lined up with the garage door closed.
- I mounted my camera to the ceiling using the supplied plastic anchors and screws:
 - First screw on the camera base plate.
 - Twist the camera onto the base plate.

Part 12: Add to Apple Home

- In the Apple Home app, tap **Add Accessory**, and then **Don't Have a Code or Can't Scan?**.
- Tap **Geek Slant Smart Garage**.
- Use the code, 12345678.
- Tap **Room** to create a new room called Garage. Feel free to change the wallpaper.
- Change the Garage Door Room to Garage.
- Tap the + button at the top right to add the camera.
- Tap **Add Accessory**, and then **Don't Have a Code or Can't Scan?**.
- Tap **Garage Camera**.
- Again, use the code, 12345678.

- Change the Room to Garage.
- After a few seconds, you should see a snapshot for the camera.
- Tap the **Garage Door** button to open and close the garage.
- You can also use Siri to open your garage door. Say, “open the garage door.”

Conclusion

I hope you found this installation script, video, and documentation helpful. If this had been available for me, it would have saved me hours of troubleshooting and frustration!

← **PREVIOUS POST ([HTTPS://GEEKSLANT.COM/LEGO-FERRARI-F40-10248-SPEED-BUILD/](https://geekslant.com/lego-ferrari-f40-10248-speed-build/))**



(<https://youtube.com/geekslant>)



(<https://facebook.com/geekslant>)



(<https://twitter.com/geekslant>)



(<https://instagram.com/geekslant>)

© 2018 Geek Slant™ (<https://geekslant.com>)