

# OpenGarage (Firmware v1.1.0) User Manual

This is the user manual for OpenGarage (Firmware v1.1.0). OpenGarage is a fully open-source product. Hardware and software details are all published at the [OpenGarage Github repository](#). For additional details, video tutorials, technical support, and user forum, visit <http://opengarage.io>

## What's New in Firmware 1.1.0?

Support for several types of temperature/humidity sensors (AM2320, DHT11, DHT22, DS18B20); improved built-in web interface; and a new setting for distance sensor reading interval.

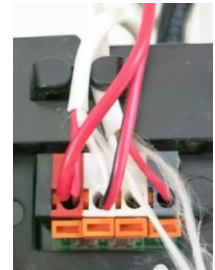
## Software Setup

- OpenGarage supports a built-in web interface for local access, and remote access through the **Blynk** app. We also provide the cross-platform **OpenGarage mobile app**. Before proceeding, please install the **Blynk app**, create an account, and scan [the QR code image here](#) to create the OpenGarage project. Blynk gives you 2000 energy points for every new account. The QR code requires less than 2000 points, so you don't need to pay any extra.
- You can log into **the same Blynk account** on multiple mobile phones, so that you and your family members can all share access to the same device.
- If you need to set up multiple OpenGarage devices, just scan the same QR code as many times as you need. Each time you scan, the project will be replicated. You can give each project a unique name, and each comes with its unique authorization token. You **will** need to purchase additional Blynk energy points for settings up multiple projects.
- After the project is scanned, go to project settings and click **Email All (or Copy All) to obtain the 32-digit authorization token**, which you will need in the following steps.
- **Power on** the device by plugging in a USB cable. The first time it powers on (or after WiFi/factory reset), the device will create an open WiFi AP named **OG\_** followed the last 6 digits of its MAC address (example: OG\_67FG8A). This is the so-called **AP mode**. Use your phone or computer to **connect to this WiFi**.
- Your phone or computer should prompt you to Sign In to the WiFi network. If you don't see the prompt, just open a browser and type: <http://192.168.4.1>. Follow the instructions there to select or input your WiFi router's SSID, and your WiFi password. If you already have a Blynk token, paste it to the **Cloud Token** box; if not, just leave that box empty and you can always set it in later.
- Once the device successfully connects to your WiFi, you will **hear a short tune** from the buzzer on OpenGarage, and the device will reboot in 10 seconds. At this point it has remembered your router's SSID and password, and the next time it's powered on it will automatically enter client mode, and obtain an IP from your router. To access the device using a browser, type in **the client mode IP** (note: this is **NOT** the 192.168.4.1 IP as you did in AP mode step above). To perform certain actions, such as trigger a button click, change options, you will need a device key. **The default key is: opendoor**
- If you don't know your OpenGarage's IP address, there are several ways to obtain it: you can find it on your WiFi router's configuration page; alternatively, there is an audible-way to obtain the IP address by hearing sound from the buzzer, as explained below; in addition, the device creates a broadcast DNS name, which is the AP mode name followed by .local/ which you can type into browser URL directly to access it (using the AP name example above, it would be OG\_67FG8A.local/).

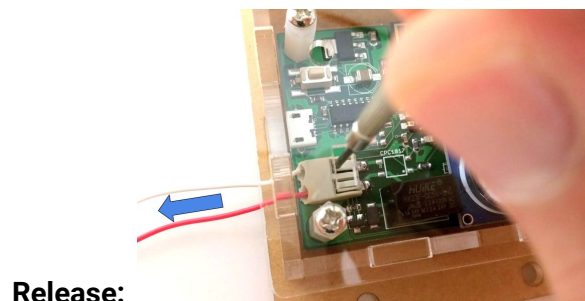
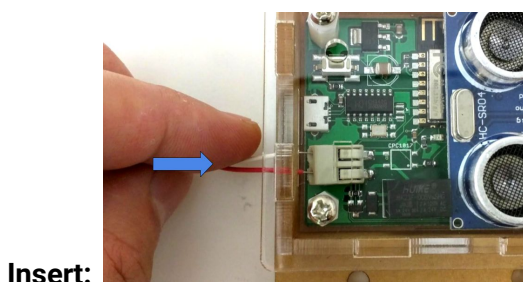
- To re-configure WiFi, you can **Reset it to AP mode**, either through the web UI, or using the pushbutton on OpenGarage. Specifically, press and hold the button for **5 seconds, until the LED changes status** (i.e. from on to off, or vice versa), but no more than 10 seconds, then release the button. Resetting the device to AP mode will NOT erase settings or log data.
- To perform **Factory Reset**: hold the push-button on the device for **10 seconds or more**, during which the LED will turn on and then turn off. Then release the button.
- If you briefly **click the pushbutton**, it triggers the relay. If you press and hold the button for more than 2 seconds but less than 5 seconds (i.e. before LED turns on), then release the button, that triggers the **Report IP** feature, which reports the IP using buzzer tones. Each digit of the IP is indicated by counting notes, starting from C4 as 1. For example, if the IP is 192.168.1.10, you will hear C4 (the leading 1), followed by a pause; then C4-C#4-D4... continuously until G#5, indicating 9, followed by a pause; then C4-C#4-pause, indicating 2; then a high-pitch tone, indicating a **dot**. Then C4-pause again, indicating the leading 1 in 168, and so on. By counting the number of continuously increasing notes, you can obtain each digit of the IP.

## Hardware Setup

- **NOTE**: OpenGarage is **NOT waterproof**. If you plan to use it outdoors, you must place it inside a waterproof box.
- **Locate Door-button Terminals**: OpenGarage uses a **built-in relay** to simulate button click. First, on your garage door system, **locate the terminals that connect to your door-button** (a.k.a. wall-button or console). Most garage door systems have four terminal ports: two connect to the door-button, and two connect to sensors. Refer to your garage door system user manual if you have trouble locate the terminals.



- **Wiring**. Take out the two-wire cable from the package, strip the wires to appropriate length. Next:
  - o If your device comes with an **orange** or **blue** terminal block, insert the stripped wires into the terminal block and tighten the screws on the terminal block.
  - o If your device comes with a **white** surface mount terminal block (as shown below), slowly and firmly push the stripped wires into the terminal holes. The terminal block has internal hooks to prevent the wires from coming out on their own. To release the wires, use a small screwdriver to press on the tabs of the terminal block, then pull the wires out.



- o Insert the other end of the stripped wires to the door button terminal you located above. Because OpenGarage uses a relay to simulate button click, **the two wires have no polarity**.

- **Mounting.** The typical way is to mount OpenGarage to the ceiling, with the distance sensor facing down. Generally you position it such that **when the garage door is fully open, the distance sensor can 'see' the door.** After you've decided on the location, use screws or double sided tape to securely mount the device to the ceiling.



- Next, **plug in the provided microUSB cable to OpenGarage**, and power OpenGarage through a USB adapter (any USB adapter with at least 1 amp output current is sufficient).
- **Security+ 2.0 System.** If your garage door system has a **Yellow antenna** and **Yellow learn button**, it's one of the newest Security+ 2.0 system. For these systems, you need a Security+ 2.0 adapter, which you can purchase as an add-on to OpenGarage. If your garage door systems has different colored antenna (other than yellow), you don't need this adapter.

## Options

The device key is required to change any option below. The default device key is **opendoor**. Many options require a reboot to take effect. On the **Basic** tab, you have:

- **Device Name:** custom device name (this name will show up on the OpenGarage homepage).
- **Door Sensor:** select the type of door sensor and mounting options as below:
  - **Ceiling Mount:** this is the default option, which uses the built-in ultrasonic distance sensor and the controller is mounted on the ceiling facing down.
  - **Side Mount:** for **rollup** type garage door systems where ceiling mount is not feasible; in this case, it's easier to set up OpenGarage on the side of the door, facing the outside.
  - **Normally Closed or Open Switch:** you can also choose to use external sensor (e.g. magnetic switch), connected between pin G04 and GND. Using these options will bypass the built-in distance sensor and use external sensor instead.
- **Door Threshold:** the distance at which the door is regarded as open. This threshold should be slightly larger than the distance from the ceiling to the door when the door is fully open, and smaller than the vehicle threshold below. The unit is centimeter. For example, if the distance from the sensor to the door when the door is fully open is 30cm, you can set this to be 50cm to allow some margin.
- **Car Threshold:** the distance at which car is detected. This threshold should be slightly larger than the distance from the ceiling to the top of your vehicle parked in the garage. The unit is centimeter. For example, if the distance from the sensor to the top of your car is 80cm, you can set this to 100cm to allow some margin.
- **Read Interval:** time between every controller status update. The unit is second.
- **Click Time:** time that the relay holds when it clicks. Default is 1000ms (1 second).
- **Distance Read Interval:** time between every distance sensor reading. Default is 500ms. *[effective after reboot]*.
- **Sensor Timeout:** if the distance sensor encounters a timeout, should the reading be ignored or capped at maximum value (450cm). Default is 'Ignore', which can help reduce sensing noise.
- **Sound Alarm:** time that the alarm will sound to alert the user before each door action.
- **Log size:** defines how many log records you'd like to keep. If you change the log size, please go to the Homepage and click Clear Log for the change to take effect.

- **T/H sensor:** choose the type of temperature/humidity sensor attached to the controller (requires additional sensor and soldering). Once configured and sensors are connected correctly, the temperature and humidity values will be shown on the homepage. *[effective after reboot]*.

On the **Integration** tab, you have:

- **Blynk token:** Blynk authorization token *[effective after reboot]*.
- **Blynk Domain and Port:** if you use custom Blynk server or port, configure them here.
- **IFTTT Key:** IFTTT webhook service key. To use this feature, go to [ifttt.com](https://ifttt.com), create an account, search 'webhook' service and create a webhook key. Copy and paste the key here. You can then **create IFTTT Applets** that use 'webhook' as trigger (this), **opengarage** as event name, and SMS, email, or push notification as action (that). When the device sends notifications, you will receive the message through SMS, email, or push notifications. The content of the message is passed via parameter **value1** in IFTTT Applet. Without IFTTT, you can still receive push notifications through the Blynk app.
- **MQTT Server:** MQTT service IP. The subscription topic is the device name.
- **Choose Notifications:** choose what notification messages you'd like to receive (this applies to both Blynk notifications within the Blynk app, and also IFTTT and MQTT notifications).
- **Automation:** if the door is left open for more than the specified amount of time, you can choose for the device to **notify** you and / or **auto-close** the door. When auto-close is selected, a minimum 5-second sound alarm will be enforced to alert the user before the door is closed.

Similarly, if the door is left open till a specified UTC time, you can choose for the device to notify you an / or auto-close the door. Note that this is UTC time (as the controller does not know your local time zone). For example, if you want it to trigger at 6pm (18:00) and your local time zone is UTC-4, then set the time in the box to 22 (as the UTC time would be 22 when it's 6pm your local time).

On the **Advanced** tab, you have:

- **HTTP port:** custom HTTP port (default is 80) *[effective after reboot]*.
- **Use Static IP:** configure the device to use custom IP *[effective after reboot]*.
- You can also change the device key on this tab.

## Firmware Upgrade

As new firmwares become available, you can download the new firmware (call **og\_x.x.x.bin**) and upgrade firmware by clicking on the **Update** button next to the Firmware Version at the bottom of the webpage. Then follow the instructions on the Update page.

## Links and Resources

- [OpenGarage Homepage](#) (including product page, technical support, and user forum)
- [OpenGarage Github Repository](#)
- [OpenGarage Firmware Files](#)
- [OpenGarage Documentation](#) (including API)
- [OpenGarage Blog Post](#)