

OpenSprinkler User Manual

Firmware 2.2.1(0) (Jul 22, 2024)

Contents

[Introduction](#)

[What's New in this Firmware?](#)

[Hardware Interface](#)

[Zone Wire Connection Diagrams](#)

[Installation](#)

[LCD and Button Functions](#)

[Firmware 2.2.0 User Manual](#)

[1. Overview](#)

[2. Accessing the Controller](#)

[3. Homepage](#)

[3.1 Device Status](#)

[3.2 Zone Attributes](#)

[4. Menu Buttons](#)

[4.1 Rain Delay](#)

[4.2 Pause Station Runs](#)

[4.3 Manually Stop a Zone from Running](#)

[4.4 Stop All Zones](#)

[5. Edit Options](#)

[6. Run-Once Program](#)

[7. Programs](#)

[7.1 Program Data](#)

[7.2 Program Preview](#)

[7.3 Zone's Sequential Group Attribute](#)

[8. Logging](#)

[9. Firmware Update, FAQ, Support, and Github](#)

[Advanced Topics](#)

[1. Installing Radio Frequency \(RF\) Transmitter](#)

[Specifications](#)

Introduction

OpenSprinkler is an open-source, web-based sprinkler/irrigation controller. It's a drop-in replacement for conventional sprinkler controllers that do not have web connectivity. OpenSprinkler's advantages include an intuitive user interface, remote access, and smart weather-based watering control. It is ideal for homeowners or business owners with applications in lawn/garden watering, plant irrigation, drip irrigation, hydroponics etc.

OpenSprinkler hardware has two different flavors: **OpenSprinkler v3** (with built-in WiFi, two independent sensor ports, and optionally wired Ethernet module) and **OpenSprinkler Pi** (or OSPi, driven by an RPi). OpenSprinkler v3 is fully assembled and comes with pre-flashed firmware. OSPi requires some assembly steps (e.g. plugging in your Raspberry Pi) and firmware installation. For all versions, the main controller can switch 8 zones; and additional zones are supported by adding zone expanders (each expander adds 16 zones). OpenSprinkler v3 supports **up to 72 zones**; OpenSprinkler Pi supports **up to 200 zones**. In addition, OpenSprinkler v3 has three variants: AC-powered, DC-powered, and LATCH:

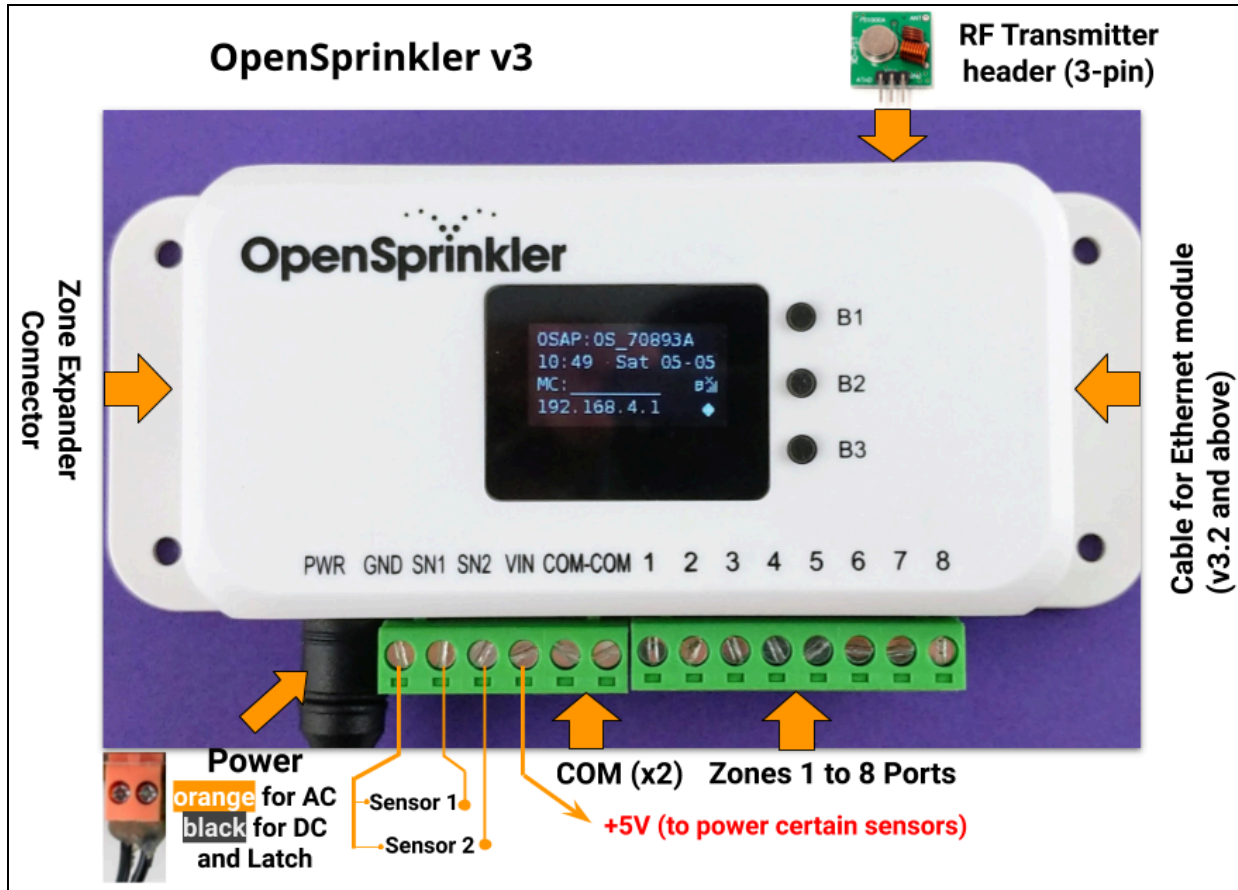
- **AC-powered** version comes with an **orange** terminal block. 24VAC power adapter is NOT included by default but can be purchased as an optional add-on (or use your own 24VAC power adapter).
- **DC-powered** version comes with a **black** power barrel, and a 7.5VDC adapter is included by default for North American users. You can also use your own DC adapter with output voltage between 7.5VDC to 12VDC, or use a 12VDC solar panel. Though it's powered by a DC adapter, it can operate 24VAC sprinkler valves.
- Similar to the DC-powered version, the **LATCH** version comes with a **black** power barrel, and a 7.5VDC adapter is included for North American users. It's designed specifically for **latching** solenoid valves **only**.

What's New in this Firmware?

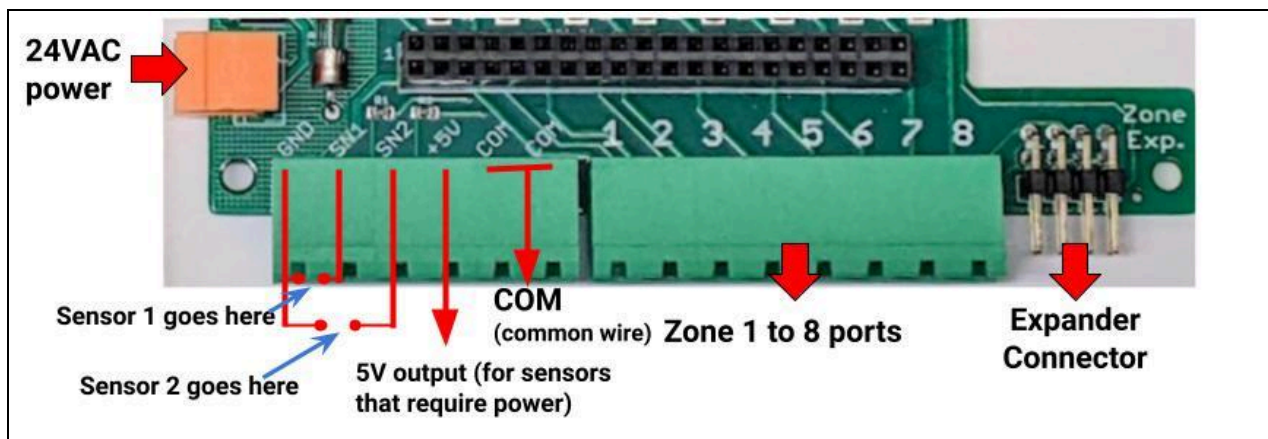
Firmware 2.2.1 is a major revision. The following are the main new changes since firmware 2.2.0:

- **For OS 3.2 and 3.3:** a new option to keep the controller in wired Ethernet mode and not fall back to WiFi.
- **For OSPi:** support for GPIO controls on Raspbian Bookworm and RPi 5.
- **For all systems:**
 - **Remote access via OTC** is now fully supported on all of OS v3, OSPi, and Linux-based systems. OTC allows remote access via the OpenThings cloud server and eliminates the need to set up port forwarding. We've also updated the OTC library and server to allow fragmentation, with the benefit of significantly lower memory / RAM usage. The smaller memory footprint subsequently enables the following two new features.
 - **HTTPs station and Remote OTC station:** two special station types are added. 1) **HTTPs station** is similar to the previous HTTP station but can connect to secure servers; 2) **Remote station (OTC)** is similar to the previous remote station specified by IP address, but allows a master OpenSprinkler to command remote controllers on a **different** network via their OTC tokens.
 - **Email notification** is now supported. It allows OpenSprinkler to send emails upon notification events by using a user-configured SMTP server. This feature is implemented to address the issue that IFTTT now requires a paid plan to use their Webhook service. The Email notification feature is free and does not require a paid plan.
 - **MQTT** has been extended to support subscribing, as well as customizing the publish and subscribe topics. Subscribing allows OpenSprinkler to receive commands from a MQTT client to trigger certain actions such as starting a zone, starting a program, rebooting etc. In addition, the character limits on MQTT settings (e.g. username, password) have been increased; and the event when a master zone is turned on has been added to MQTT and other types of notifications.

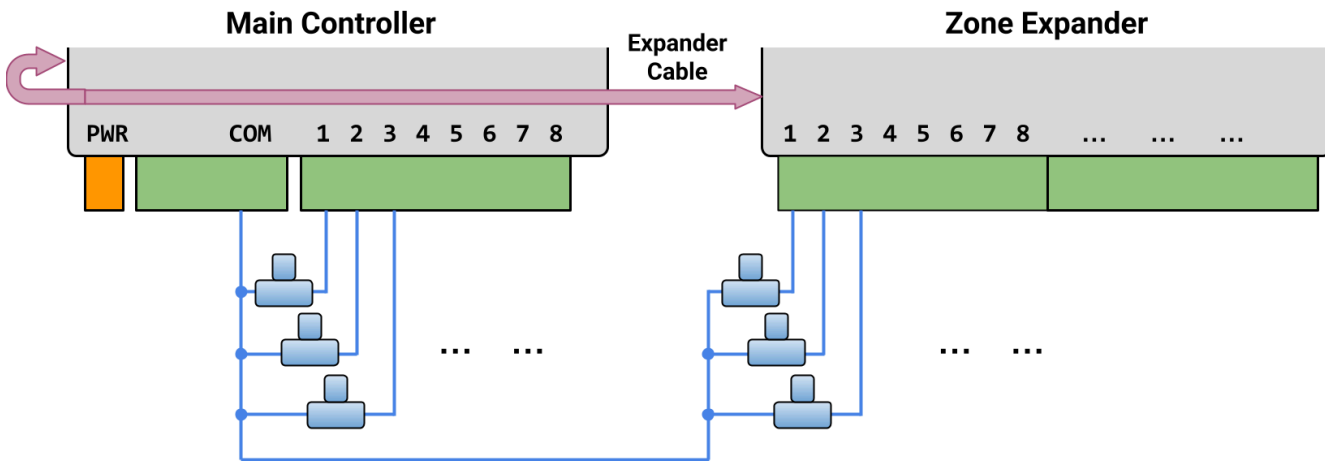
Hardware Interface



OpenSprinkler Pi (OSPi):



Zone Wire Connection Diagrams



- Each valve has two wires. Typically, one wire from each valve (regardless of whether it's on the main controller or an expander) is connected together into a COM (common) wire, and that COM wire should go to the COM terminal port (**NOT GND!!!**) on OpenSprinkler.
- OpenSprinkler has two COM ports: you can use either -- they are internally wired together so are the same.
- The other wire of each valve goes to an individual zone port.
- If you have a master or pump start relay wire, you can insert it into any zone port – OpenSprinkler uses software defined master / pump zone so you can dedicate any zone as a master zone.

Installation

Important Notes

- OpenSprinkler is **NOT waterproof**. For **outdoor use**, please purchase a [waterproof enclosure](#).
- **International customers:** for the OpenSprinkler AC-powered version, please find a 24VAC transformer compatible with your country's powerline voltage standard. Failure to observe the powerline voltage standard can result in damage to your controller. If you cannot find a 24VAC transformer, you may consider the OpenSprinkler DC-powered version, which uses a low-voltage (7.5~12VDC) adapter.

Wiring and Installation Tutorial Video:

- **Installation and tutorial videos** are available at <https://support.opensprinkler.com>

Instructions:

Step 1: Label wires and remove your existing sprinkler controller.

- Carefully label and remove the wires from your existing sprinkler controller, then remove it from the wall. Typically you will see the **power supply wires**, the **COM (common) wire**, one or more **Zone wires**. Depending on your setup, you may also find a **Master Zone** (or Pump Start Relay) wire, and **Rain / Soil / Flow Sensor wires** (if using any such sensor).

Step 2: Attach OpenSprinkler to the wall, and re-insert the wires:

- Refer to the Hardware Interface diagram and Zone Wire Connection diagrams on the previous pages. All terminal blocks on OpenSprinkler can be unplugged for easy wiring. To unplug, firmly grab both ends of the terminal block, wiggle, and pull it out.
- Insert COM and zone wires to their corresponding terminal ports on OpenSprinkler.
 - For **OpenSprinkler DC and LATCH**: OpenSprinkler's COM terminal is **positive (+)**. If your solenoid wires have polarity, make sure its positive wire (typically **red**) goes to COM.
- For **OpenSprinkler AC**, insert and tighten the 24VAC wires into the **Orange** terminal block (**AC has no polarity** so the two wires have no distinction);
For **OpenSprinkler DC or LATCH**, insert the DC power adapter into the power barrel on OpenSprinkler.
- A **sensor** should be connected **between SN1 and GND** (or **SN2 and GND** if using a second sensor).
Do NOT plug in any sensor wire to COM – OpenSprinkler uses GND (not COM) as common for sensors. For additional details on how to connect sensors (e.g. rain or flow sensor), refer to [later sections](#) in this manual.

Step 3: Link OpenSprinkler Zone Expanders (Optional):

- To link zone expanders: **first, power off the main controller**. Once the power is off, plug one end of the zone expander cable into OpenSprinkler's **Zone Expander Connector**: the connector is polarized so you can only plug it in one way.
 - For **OpenSprinkler Pi (OSPi)**: plug the other end of the cable to Zone Expander's **IN connector**. If you have multiple expanders, daisy chain them by following the **OUT → IN** links.
 - For **OpenSprinkler v3**: plug the other end of the cable to either of Zone Expander's connectors on the side. If you have multiple expanders, use additional cables to link them (again, the two ports on each expander are identical so you can use either). Because all Zone Expanders are hooked onto the same (I2C) bus, you **must set for each Zone Expander a unique index** (1, 2, 3, or 4). Each Zone Expander has a DIP switch (see picture on the right) that allows setting the index. The first expander (directly connected to the main controller) should have an index of **1 (OFF OFF)** on the DIP switch, the second expander should be **2 (ON OFF)**, the next is **3 (OFF ON)**, and the last is **4 (ON ON)**.

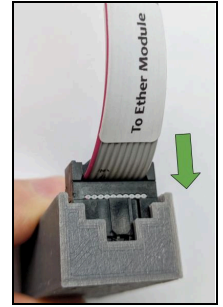


Zones on the main controller correspond to indices 1 to 8; zones on the first expander (DIP position OFF OFF) correspond to indices 9 to 24; and so on. The firmware can detect the expander with the highest index, however, **you still need to manually set the number of zones in settings**. You can set more zones than physically available, to take advantage of 'Virtual Zones' features (e.g. Remote, HTTP, RF zones).

Step 4: Setting Up WiFi or Wired Ethernet on OpenSprinkler v3:

- In **WiFi mode**, the first time you power on OpenSprinkler (or every time after a factory reset), it **starts as a WiFi AP (Access Point)**, creating an open WiFi network with SSID displayed on the LCD screen (e.g. OS_xxxxxx). Use your phone or laptop to connect to this WiFi network (**Note**: on Android phones, you may receive a warning: WiFi has no Internet connection. Select **Yes** to stay connected).
- Once your phone or laptop is connected to the AP, open a browser and type in the AP IP address **192.168.4.1**. Follow the instructions to complete the WiFi setup. In particular, you need to select (or type in) **your home WiFi router's SSID and password** (NOT OpenSprinkler's password!). The WiFi network's **BSSID and channel** are also automatically filled in (though you can optionally leave them empty). Then click on **Connect**.

- Once the device connects to your WiFi successfully, it will automatically reboot into WiFi **Station** mode. To access the controller in station mode, **click push-button B1** – that will display the **device IP** address to the LCD screen.
- On your phone or computer, open the OpenSprinkler mobile app, or simply launch a web browser and type in the device IP address, you will see the OpenSprinkler web interface and homepage.
The default device password is opendoor.



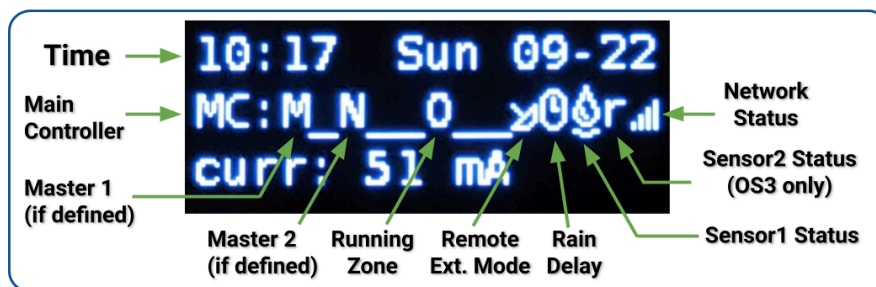
Wired Ethernet: from v3.2, the controller has dual support for WiFi and wired Ethernet. You can purchase an add-on Ethernet module. When the Ethernet module is plugged in, the controller boots in wired Ethernet mode; when the module is unplugged, it boots in WiFi mode. The controller comes with a built-in module cable: plug the cable connector firmly to the Ethernet module as shown on the right – the connector is polarized so there is only one way to plug it in. Then you can plug in a wired Ethernet cable (RJ45) to the other end of the module.

Reset WiFi / Password Reset / Factory Reset

- **Reset WiFi:** for OpenSprinkler v3, if you need to reset or switch WiFi network (without erasing program settings): after controller is powered on, press B3 followed by B2 (B3+B2 like how you press Ctrl+C) until the screen displays '**Reset to AP mode?**'. Then Click B3 to confirm. You can also trigger a WiFi reset using the OpenSprinkler mobile app. To do so, go to Edit Options -> Reset -> Reset WiFi.
- **Reset Device Password:** if you have forgotten your device password, you can use buttons to bypass the password. To do so, first remove power from OpenSprinkler; then plug the power back in, **as soon as you see the OpenSprinkler Logo, press and hold push-button B3**, and continue holding B3 until the LCD displays 'Setup Options'. Then click B3 as many times as you need until it shows 'Ignore Password'. Click B1 to select Yes, and finally **press and hold B3 until the controller reboots itself**. Now you can access the controller with any password. For security reasons, we recommend you to immediately change the password and set the Ignore Password option back to No.
- **Factory Reset:** to perform a factory reset, first remove power from OpenSprinkler; then plug the power back in, **as soon as you see the OpenSprinkler Logo, press and hold push-button B1**, and continue holding B1 until the LCD displays 'Reset?'. Make sure the answer is Yes, then **press and hold push-button B3** until the controller reboots itself and completes the resetting process.
 - Upon factory reset, the device password is reset back to the default, which is **opendoor**.

LCD and Button Functions

LCD (the illustration below is captured on OpenSprinkler 3.0's LCD)



- Master Zone 1 (if defined) is marked by **M**; and Master Zone 2 (if defined) is marked by **N**.
- By default the LCD displays the first 8 zones. If you have any expanded zones, click B3 to switch to another group of 8 zones. Each **running/open** zone is displayed with a three-letter animation: **. o O**
- If the controller is configured to Remote Extension mode, a radar icon will be shown.
- If 'pause station runs' or 'rain delay' is active, a clock icon will be shown.
- If Sensor1 (available on all versions) is defined, a letter will be shown: **r** for rain sensor, **s** for soil sensor, **p** for program switch, **f** for flow sensor. When the sensor is detected to be active, the letter will be capitalized. In addition, a rain icon will be shown for the rain sensor, and a soil icon will be shown for the soil sensor.
- If Sensor2 is defined, the display is similar to Sensor1.

Buttons

After the controller is powered on, the buttons are assigned the following functions:

| | <i>Click</i> | <i>Press and Hold (i.e. Long Press)</i> |
|----------------|---|---|
| B1: | Display device IP address | Stop all zones immediately |
| B2: | Display device MAC address | Reboot the controller |
| B3: | Toggle between the main controller (MC) and each expansion board (E1, E2, E3 etc). | Manually start an existing program or a test program. |
| B1 + B2 | Display gateway (router) IP (i.e. press and hold B1, then press B2, similar to how you press Ctrl+C etc.) | |
| B2 + B1 | Display external (WAN) IP. | |
| B2 + B3 | Display timestamp of the last weather server response | |
| B3 + B2 | For OpenSprinkler v3: reset controller to AP mode (for re-configuring WiFi). | |
| B1 + B3 | <i>(for internal testing only)</i> Start a very quick (2 seconds each zone) test program. | |
| B3 + B1 | Display the last system reboot timestamp and reboot reason. | |

Pressing and holding the following buttons while the OpenSprinkler Logo is displayed at reboot:

If **B1** is pressed and held while the logo is displayed at reboot: enter **Factory Reset**.

If **B2** is pressed and held while the logo is displayed at reboot: enter internal test mode

If **B3** is pressed and held while the logo is displayed at reboot: enter **Setup Options**.

Firmware 2.2.1 User Manual

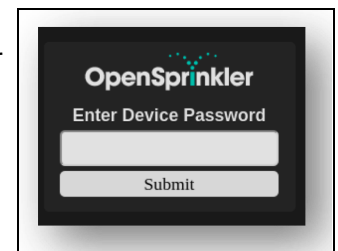
1. Overview

OpenSprinkler's built-in web interface works with both desktop and mobile browsers. It allows you to change settings and modify programs at any time using **either a web browser or the free OpenSprinkler mobile app**. To install the mobile app, just search **opensprinkler** in your app store.

- **Firmware tutorial videos** are available at <https://support.opensprinkler.com>

2. Accessing the Controller

Local Access: once the controller establishes a valid connection, you can find out its local IP address and HTTP port number on the LCD by clicking button B1. In the following we will refer to the IP address as **os-ip** (for example: *192.168.1.122*). Next, open a browser and type in the URL <http://os-ip> (e.g. <http://192.168.1.122>). In case you have set a custom HTTP port number (other than the default 80), include the port number in the URL as well. For example, if you have changed OpenSprinkler's port number to 8080, type in <http://os-ip:8080>. This way you can locally access the controller while you are on the same network as the controller.

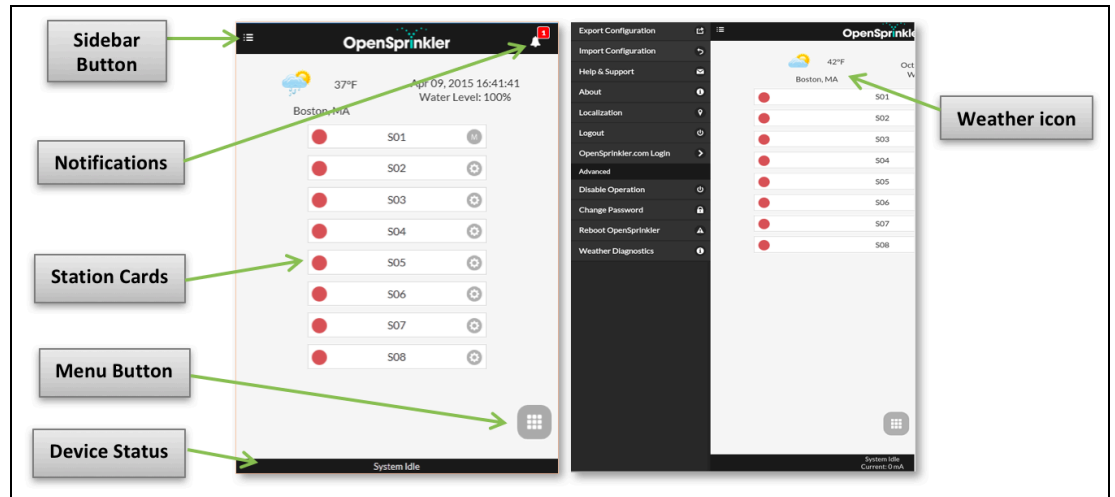


Device Password: the factory default password is opendoor. For security reasons, please change this password upon the first use.

Remote Access: after setting up OpenThings Cloud (OTC) (see **Section 5**), you can access the controller remotely using the OTC token. The OpenSprinkler mobile app supports adding a device by OTC token. Alternatively, you can open a browser and type in <https://cloud.openthings.io/forware/v1/token> where **token** is the OTC token (refer to Section 5).

3. Homepage

At the home page you will find a weather icon and the list of all stations and their current status. The bell button in the upper-right corner (when visible) indicates notifications are present and opens a sidebar with all notifications. The button on the upper-left corner activates the left-side menu, which shows additional menu items such as:



- **Export/Import Configuration:** save/restore controller settings and programs. This is useful during firmware upgrade or in preparation for factory reset.
- **About:** display the app version, firmware version, and hardware version.
- **Localization:** change the display language.
- **OpenSprinkler.com Login:** log in with your opensprinkler.com account name and password. This allows cloud synced data such as station photos, notes, site configurations etc.
- **Disable Operation:** disables zone operations. This is useful if OpenSprinkler will not be used for a period of time.
- **Change Password:** change device password (default is *opendoor*).
- **Reboot OpenSprinkler:** trigger a software reboot of the controller.
- **System Diagnostics:** displays diagnostic information, including timestamp and cause of the last reboot; timestamp of the last weather call, response received, and response code; OpenThings Cloud (OTC) status etc.

TIP: at any time, you can **activate the left-side menu** by swiping your mouse pointer from left to right (on desktop browsers) or swiping your finger from left to right (on mobile browsers).

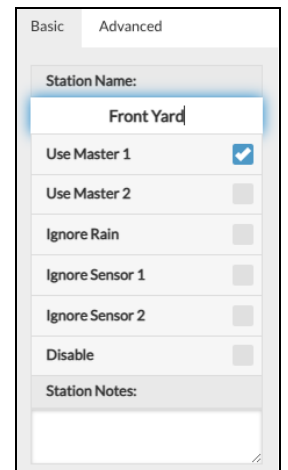
3.1 Device Status

The footer shows the device's current status. It shows information in the following priority: system enable status, currently running stations, pausing status, or active rain delay status. If no active events are detected the system will display information of the last run station and if not available will read 'System Idle'. If a flow sensor is installed, the status bar shows the real-time flow rate (updated every 30 seconds). If any zone is active, the status bar also shows the total current draw of all solenoid valves.

3.2 Zone Attributes

Click the gear icon next to each zone name, the zone attribute box will pop up. On the **Basic tab**:

- **Station Name:** give each zone a **custom name** (up to 32 characters).
- **Use Master:** when checked, the associated Master zone(s) will turn on whenever this zone is activated. This attribute is only visible if a Master zone is defined.
- **Ignore Rain / Sensor1 / Sensor2:** when checked, this zone will bypass manual raindelay, sensor1 or sensor2. These attributes allow you to customize if a zone is affected by any of them. By default these flags are off.
- **Disable:** when checked, this zone will not run and will be hidden from the user interface.



On the **Advanced tab**: you will see the **Sequential Group** attribute. A zone can be assigned to any of the **4 sequential groups (A, B, C, or D)**, or to the **parallel (P) group**. Zones in the same sequential group will be automatically serialized so no two zones in the same group will run at the same time. In contrast, zones in different groups can run simultaneously. Any zone in the parallel (P) group can run simultaneously with any other zones. By default, all zones are in Sequential Group A. The sequential group attribute generalizes and

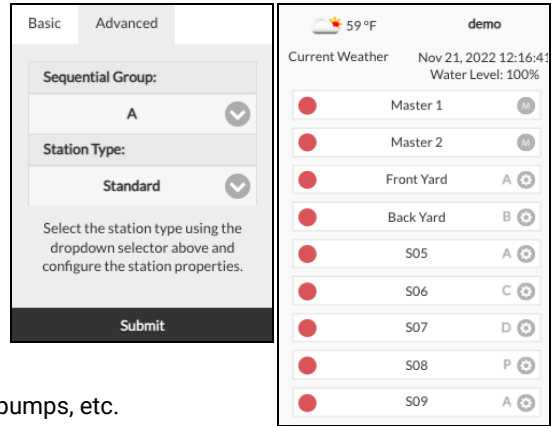
replaces the previous per-zone 'Sequential' flag, providing more flexibility in simultaneous zone runs. The group label of each zone is shown on the homepage, next to the zone name.

The other attribute on the **Advanced tab** is **Station Type**. The default type is **Standard**; you can also choose from several types of **special/virtual stations** as explained below:

- **RF zone:** by installing a RF (radio frequency) transmitter (see [Advanced Topics](#) section), OpenSprinkler can switch many common RF remote power sockets. To do so, you need to purchase an **RFtoy** and use it to decode RF signal code from your remote power sockets. Once the code is set, whenever this zone is turned on/off, OpenSprinkler will send the on/off code through an RF transmitter to remotely switch the power socket. This allows your OpenSprinkler to control powerline devices, such as Christmas lights, heaters, pumps, etc.
- **Remote Station (IP):** a remote station is a physical station on another OpenSprinkler. This allows one master controller to send commands to remote controllers in order to turn on their valves. Each remote station is specified by the *IP address*, *port number*, and *station index* on the remote controller. For example, you can map master controller's zone 1 to remote controller's zone 5. However, the master and remote controllers **must share the same password**.
- **GPIO zone:** allows the zone to directly switch a spare GPIO pin available on the controller. Only valid GPIO pins are visible. You can also define the active state (i.e. Active High or Active Low).
- **HTTP zone:** allows the zone to trigger a generic HTTP GET command. To define an HTTP station, you need to provide a *server name* (either domain name or IP address), *port number*, *on command* (without the starting slash /), and *off command*. When this zone is turned on, it automatically sends a HTTP GET command **server:port/on_command**; similarly, when it's turned off, it sends a HTTP GET command in the form of **server:port/off_command**.
- **HTTPs zone:** similar to HTTP zone but for secure (https) servers.
- **Remote Station (OTC):** similar to remote station (IP) but the remote controller is specified using its OTC token. This allows easy management of master and remote controllers that are located on different WiFi / LAN networks. The master and remote controllers must share the same password.

Note that the above special zones are **virtual** – you do NOT need to have physical zones to use them. In other words, even if you don't have any zone expander, you can still use special zones up to the maximum number of zones supported.

Cloud Synced Attributes: once you log in to opensprinkler.com in the UI/app (through sidebar menu), cloud-synced attributes will become available, including station photos and notes. You can take a custom photo for each station by using the OpenSprinkler mobile app.



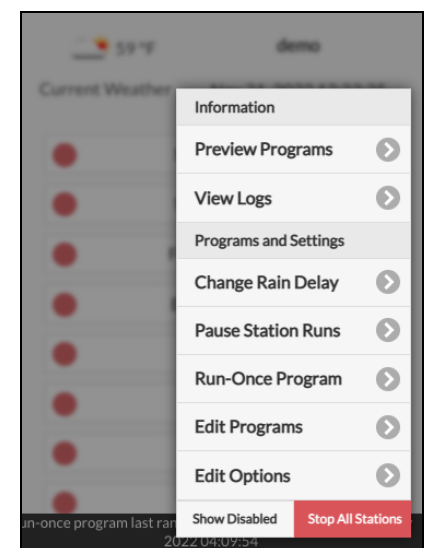
4. Menu Buttons

The menu button is on the bottom right corner of all pages and allows quick navigation between pages relevant to the controller. This menu has the following options:

- Preview Programs, or use keyboard shortcut ALT+V
- View Logs, or ALT+L
- Change Rain Delay, or ALT+D
- Pause Station Runs, or ALT+U
- Run-Once program, or ALT+R
- Edit Programs, or ALT+P
- Edit Options, or ALT+O
- Stop All Stations (Section 4.4 below).

Additionally, when you are at the home page, the menu has a button labeled **Show Disabled**, which allows disabled zones (if any) to show up in the zone list.

TIP: at any time, you can activate the menu by pressing the **M** key on your keyboard.



4.1 Rain Delay

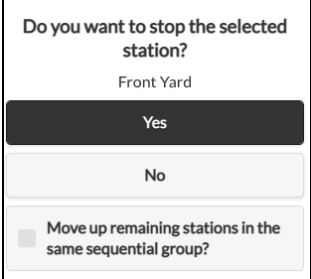
To manually trigger a rain delay time, click on the menu button, and select **Change Rain Delay** to set a custom rain delay time (in hours). Any zone affected by rain delay will immediately stop, and will not run until the rain delay time is over. To cancel an existing rain delay, simply set 0 as rain delay time.

4.2 Pause Station Runs

Select **Pause Station Runs** and specify the amount of time, this will trigger a pause where existing zones will stop running immediately, and resume after the pause timer counts down to 0. The start times of all zones, including those in the queue waiting to run are adjusted accordingly. If a program's start time falls during the pause time, the program will also be queued and paused until the end of pause timer. During pause, the footer will display the Pause status. You can cancel the pause by **clicking the footer**, or menu -> **Resume Station Runs**.

4.3 Manually Stop a Zone from Running

If a zone is currently running or scheduled to run, you can click on its zone card. A dialog will pop up to confirm whether you want to stop it from running. If there are other zones in the same sequential group as the one you are stopping, you will see a checkbox "**Move up remaining zones in the same sequential group**"? Once checked and confirmed, the remaining zones in the same sequential group will be shifted up, so the next zone will start running immediately instead of waiting for its originally scheduled start time.



Do you want to stop the selected station?
Front Yard

Yes

No

Move up remaining stations in the same sequential group?

4.4 Stop All Zones

Click **Stop All Stations** to stop all zones immediately, including those in the queue waiting to run.

5. Edit Options

Click menu -> **Edit Options** (or use keyboard shortcut ALT+O) to configure settings in the following categories.

System Settings

- **Location:** click on Location to open a map, where you can search, locate, and select your address.
 - **PWS location:** if you prefer using WUnderground PWS location, you need to provide a valid WU key in the **Advanced** tab, submit, then come back to Location settings, and the blue dots will show PWS locations.
- **Time Zone:** OpenSprinkler uses your location to automatically detect your time zone, Daylight Savings Time (DST) setting, and obtain weather data online. If you would like to manually set the timezone, you should: 1) leave the Location field empty, and 2) disable NTP Sync (in the **Advanced** tab).
- **Enable Logging:** enable logging (data will be stored in the internal flash or microSD card). Default is on.
- **Use Metric:** the UI will automatically detect unit system (imperial vs. metric), but you can also manually set unit system.
- **Orders Stations by Groups:** when checked, the zones will be ordered by their sequential groups at the homepage.

Configure Master

This firmware supports **up to 2 independent** Master Stations, each with their own settings as listed below.

- **Master Station:** define a master station/zone, also known as pump station. Master is a zone that turns on together with other zones. Any zone can be set as a master. Default is none (i.e. not using master station).
- **Master On Adjustment:** fine tune the exact time at which the master turns on. The **acceptable range is -600 to 600 seconds, in precision of 5 seconds**. For example, when set to 15, the master station will turn on 15 seconds **after** a station opens; when set to -60, the master station will turn on 60 seconds before an associated station turns on.
- **Master Off Adjustment:** fine tune the exact time at which the master turns off. The **acceptable range is -600 to 600 seconds, in precision of 5 seconds**. For example, when set to -15, the master station will turn off 15 seconds **before** a station closes; when set to 60, the master station will turn on 60 seconds after an associated station has turned off.

Station Handling

- **Number of Stations:** OpenSprinkler can automatically detect the number of available expansion boards; nonetheless, the user still **needs to manually set the number of zones**, because it's allowed to exceed the number of physical zones. This is particularly useful when using the virtual stations feature (e.g. remote or RF stations). The default number of zones is 8.
- **Station Delay:** the delay time between two consecutive stations. The **acceptable range is -600 to 600 seconds, in precision of 5 seconds**. For example, when set to 60, the second zone will open 1 minute after the first zone closes. When set to -15 seconds, the second zone will open 15 seconds prior to the closing of the first. A negative station delay is sometimes useful to solve water throttling issues. Default is 0 (i.e. the next zone will run immediately after the previous).

Weather and Sensors

- **Weather Adjustment Method:** select weather-based adjustment method. If **Manual** is selected, the % *Watering* is configured manually. Otherwise the % *Watering* is set automatically by the selected method. Explanations of supported adjustment methods can be found on the [support page of OpenSprinkler.com](#).
- **Adjustment Method Options:** tap to configure options for the selected adjustment method.
- **Weather-Based Restriction:** select weather-based restriction. If **None** is selected, no restriction will take effect. If **California Restriction** is selected, no watering will occur if more than 0.1" of rain has accumulated in the past 48 hours.
- **% Watering:** global scaling factor applied to station water times. For example, if set to 150%, all station water times will be multiplied by 150% (except in programs that do not use weather adjustment). Default is 100%. This option is disabled if automatic adjustment method has been selected.
- **Sensors:** OpenSprinkler supports several types of sensors: Rain, Soil (digital output only), Flow, or Program Switch. OpenSprinkler v3 supports two independent sensors (SN1 and SN2), allowing you to use two sensors at the same time (except flow sensor is only supported on SN1). Note that a sensor should be connected between SN1 and GND (or SN2 and GND). **Do NOT** plug in any sensor wire to COM as that may damage the controller.

- **Sensor Installation:** all sensors described below have two wires (except the 3-wire flow sensor). To install Sensor1: insert the two wires to the sensor terminals. On OpenSprinkler v3, insert the two wires to SN1 and GND. To install a second sensor, insert the two wires to SN2 and GND.
- **Rain/Soil Sensor:** rain or soil sensor allows the controller to automatically stop zones from running if rain or soil moisture is detected. You can additionally select the sensor option: *normally open* or *normally closed*. A rain sensor is essentially a rain-activated switch. They are typically 'normally closed'. Note that while many soil sensors on the market output analog signals, OpenSprinkler only supports those that output digital signals (binary ON or OFF). If your soil sensor is analog, you can use a threshold adapter to convert it to a digital sensor.
- **Delayed On / Off Times:** you can optionally set delayed on / off times for rain and soil sensors. For example, 'delayed on time of 10 minutes' means the sensor must remain on for at least 10 minutes for it to be regarded as activated; 'delayed off time of 30 minutes' means the sensor must remain off for 30 minutes for it to be regarded as deactivated. The 'delayed on time' can prevent false triggering of sensors; and 'delayed off time' allows you to extend the sensor activation for a specified amount of time (e.g. extend rain sensor activation after rain stops).
- **Program Switch:** if this sensor is selected, you can connect a switch / button to the sensor port. When the switch is pressed for at least 1 second, the controller will trigger Program 1 (or Program 2 if the switch is installed on SN2).
- **Flow Sensor:** when using a flow sensor, the controller will detect flow sensor pulses, display real-time flow rate at the footer, and log the flow volume at the end of each station run and program run. By default **Dry-contact, 2-wire flow sensors are supported:** these sensors are essentially flow-activated reed switches which close and open repeatedly as water flows through the meter. They do not need power, and typically come with 2 wires. Insert the two wires to the sensor terminals on OpenSprinkler (there is no polarity). (Note: on OpenSprinkler v3, only SN1 supports flow sensor, SN2 does NOT support flow sensor). Then set the 'flow pulse rate' – which you can find out in your flow sensor's datasheet – it's used to convert the pulse count to actual volume. We recommend you to keep the unit as L/pulse, even if you normally use Gallon/pulse (i.e. you can keep it as 1L/pulse even if the actual unit should be 1Gallon/pulse): only the numbers matter, the unit is for display only.
- Flow sensors with 3 wires and runs on 5V can also be used with OpenSprinkler. In this case, insert the ground wire (typically black colored) GND, 5V wire (typically red colored) to VIN, and the sensor wire (typically yellow) to SN1.



Integrations: OpenThingsCloud (OTC)

- **OTC:** configure remote access via OpenThings Cloud (OTC) token. For details, please refer to this [OTC support article](#).
- **MQTT:** configure MQTT settings. For details, please refer to this [How to use MQTT support article](#).
- **Email Notifications:** configure Email settings. For details, please refer to this [Setting up Email notifications support article](#)
- **IFTTT Notifications:** configure IFTTT settings. For details, please refer to this [Setting up IFTTT support article](#).
- **Notification Events:** configure the set of events that the controller will send notifications on.
- **Device Name:** this device name will be included in all notification messages to help identify which controller sent the messages.

Advanced Settings

- **Wunderground Key:** Wunderground (WU) API key. Useful if you want to use WU as a weather data source. If you have an existing WU key, fill it here and use the **Verify** button to check if the key is valid. If you don't have an existing WU key, follow [this support article to create one](#). When this key is provided, AND if you have selected a PWS station as your location, the OpenSprinkler weather algorithm will automatically use WU as a data source. Otherwise the default data source is Apple WeatherKit.
- **HTTP Port:** customize the device's HTTP port. Default is 80. **There is typically no need to change this setting.**
- **Boost Time:** this option is **only applicable to DC-powered OpenSprinkler**. It sets the DC voltage boost time (from 0 to 1000ms). Default is 320ms. When using a low-voltage DC power adapter, such as 5VDC, to drive 24VAC sprinkler valves, you may want to suitably increase the boosting time to allow valves to be successfully energized.
- **Ignore Password:** when enabled the device password will be ignored (i.e. equivalent to setting an empty password).
- **Special Station Auto-Refresh:** whether to periodically auto-refresh special stations (e.g. RF, remote, HTTP stations etc.)
- **NTP Sync:** automatic time-sync based on your location. When off, you can set the device time manually. Default is on.
- **NTP IP Address:** set a custom NTP time-sync server.

- **Use DHCP:** when checked, OpenSprinkler's IP is dynamically assigned by the router. When off, the IP is statically assigned, in which case you will need to manually set a **Static IP**, **Gateway IP**, **Subnet Mask**, and **DNS IP**. Default is on.

Note: instead of turning off DHCP, it's highly recommended that you use the router's DHCP reservation, or Bind IP to MAC feature to manage static IP assignment.

Reset

- **Clear Log Data:** Clear all log data.
- **Reset All Options:** Restore all options back to factory settings.
- **Delete All Programs:** Delete all programs.
- **Reset Station Attributes:** Restore all station settings to factory settings.
- **Reset Wireless Settings:** for OpenSprinkler v3 only, reset controller to WiFi AP mode.

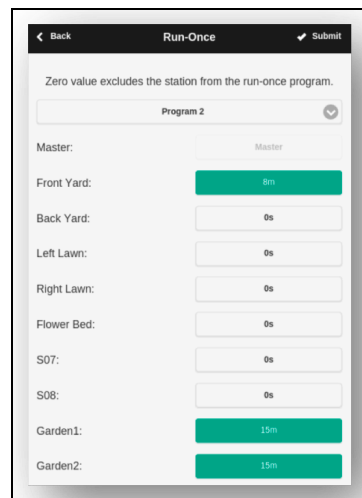
6. Run-Once Program

To manually start a program, go to the home page, and click **Run-Once Program** (or use keyboard shortcut **ALT+R**). You can load preset water times from any of the existing programs, or a test program (e.g. 1 minute each station). You can also manually edit the water time for each station.

- If the controller is already running an existing program, you will be prompted to stop it in order to continue.
- Zone attributes are obeyed, such as *Use Master1 or Master2 etc.* Controller settings, such as *Sequential Group, Station Delay Time, Master On / Off Time*, are also obeyed.
- *Rain Delay* and *All Sensor* settings are ignored, and *% Watering Adjustment* is ignored, because the run-once program is treated as a manual override program.

TIP 1: you can start a run-once program on the controller using buttons. This is useful for landscapers/gardeners to perform routine checks without having to give them WiFi access. To do so, press and hold button B3 until the LCD displays 'Run a Program'. Then click B3 to navigate through the list of available programs. Once you have found the program, press and hold B3 until the controller starts to run that program.

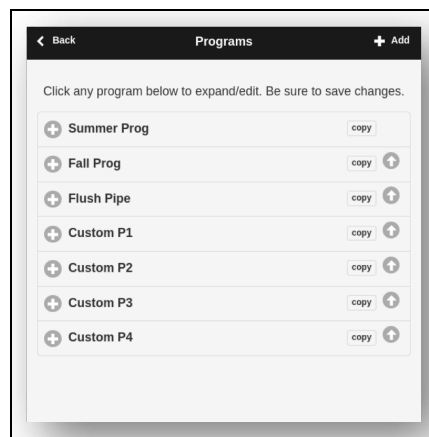
TIP 2: If you'd like to set a test program which is not intended to run normally, but only for run-once program, or for starting using buttons on the controller manually: you can create a new program, and set it as 'disabled' (refer to the next section for setting a program as disabled.). This way the program will not run normally, but will appear in the list of available programs that you can load as a run-once program, or using buttons to start the program manually.



7. Programs

At the home page, click **Edit Programs** (or ALT+P) to see the list of existing programs. Here you can:

- **Create** a program by either adding a new program or copying an existing program.
- **Modify** or **Delete** an existing program.
- **Reorder** existing programs by using the arrow keys.
- **This firmware supports a maximum number of 40 programs.**



7.1 Program Data

Click on the **+ Add** button at the upper-right corner to create a new program. Each program contains the following data:

Basic Settings

- **Program Name:** a custom program name, up to 20 characters. See the next page for special program names.
- **Enabled:** program enable flag. Default is enabled.
- **Use Weather Adjustment:** when checked, the % *Watering* applies to all station water times in this program. Default is on.
- **Enable Date Range:** when checked, you can define a date range, including the start date and end date, when the program is enabled. For example, you can set a program to run from 05/15 to 09/15 (May 15 to Sep 15 each year), or from 11/10 to 02/20 (Nov 10 to Feb 20 the following year).
- **Start Time:** the first start time of the program (e.g. 8:00 AM). Also supports sunrise or sunset time +/- offset as start time.

Program Type

- **Weekday:** program will run on the selected weekdays.
- **Interval Day:** program will run every N days, where N is between 1 to 128. You will also need to set the **Starting in** days. For example, a program that repeats every 5 days starting in 0 days will run today, and every 5th day from today. A program that repeats every 3 days starting in 1 day will run tomorrow, and every 3 days from then.
- **Restrictions:** odd/even day restrictions. **Odd** day means the program will run only if it's an odd day of the month (except the 31st, or Feb 29th). Similarly for even days. Default is no restrictions.

Station Water Times

- Set the water time for each station. A value of 0 means the station will not run. The range of water time is 0 to 64800 seconds (18 hours). Unlike previous firmware, this firmware does not use compression of water time anymore, so it allows full precision of 1 second.
- Also supports sunrise-to-sunset and sunset-to-sunrise durations.

Additional Start Times

There are two choices of additional start times:

- **Fixed:** up to 3 given additional start times. Any time of a day.
- **Repeating:** repeating start times at regular time intervals. For example, repeat every 90 minutes for 7 times. This is useful for breaking down long water times into shorter cycles. Unlike previous firmwares, this firmware allows repeating start times to go overnight to the next day.

The screenshot shows the 'Add Program' configuration interface. It includes sections for Basic Settings, Program Type, Restrictions, Stations, and Additional Start Times. Green arrows indicate the mapping between the text descriptions and the UI elements.

Special program names can be used to trigger a controller reboot on a regular basis:

- If the program name is **>:reboot** it will trigger a reboot when the controller is idle (i.e. no program is running).
- If the program name is **>:reboot_now** will trigger a reboot regardless of its running status.

Both reboots will be delayed by 1 minute or so from the start time to avoid the action triggered again right after rebooting. As an example: create a program that starts every day at 2:00am with name **>:reboot** will trigger a reboot every day at 2am. When creating the program, at least one zone must be selected to run but it will be ignored because the firmware recognizes the special program name and does not actually run any zone included in this program.

7.2 Program Preview

To verify that all programs are set correctly, go to the home page and click **Preview Programs** to visualize how the programs are scheduled to run each day.

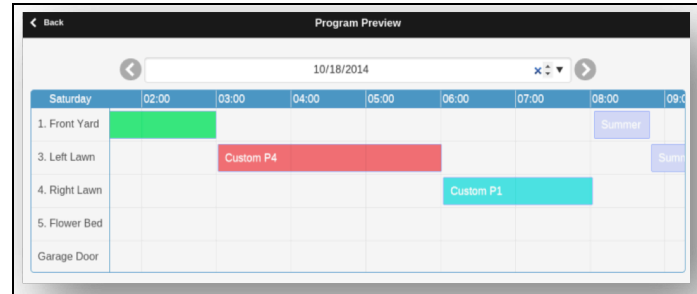
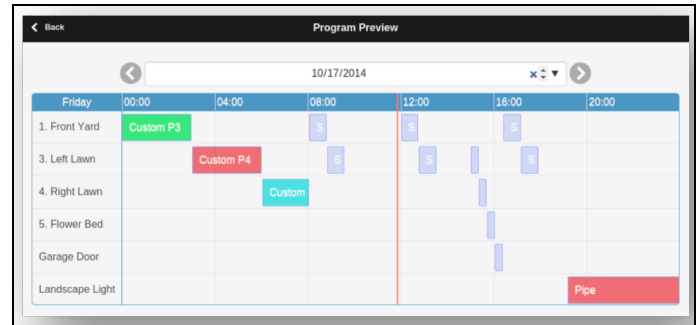
- **Today's schedule** is shown by default. Click on the left and right arrows at the top to change to a different day.
- **Current time** is indicated by a pink line. You can zoom in/out or drag the plot left/right to check details.
- **Colored bars** show the program name and time span of each station run. Clicking on each bar directs you to the specific program editing page.

NOTE 1: The program preview is implemented using a software simulation of the scheduling algorithm:

- What you see accurately reflects how the programs are scheduled to run on the controller.
- All controller settings, such as *Master zones*, *Sequential Group*, *Station Delay Time*, *Master On / Off Time*, are observed; and all station attributes such as *Use Master Station*, *Activate Relay* are also observed.

NOTE 2:

- **Rain Delay and Rain Sensor are ignored** because these dynamic events cannot be predicted during preview.
- Programs that are set to **Use Weather Adjustment** will be scaled by the current % **Watering** parameter.
 - When using the **Manual Adjustment** method, the same % **Watering** is applied to every day.
 - When using the **Zimmerman or ETo Adjustment** method, the current % **Watering** (dynamically calculated on a day-to-day basis) is only applied on today's schedule, while 100% is applied to all other days.
 - When the watering level is less than 20%, any station with a resulting water time less than 10 seconds will be skipped (due to the water time being too short).



7.3 Zone's Sequential Group Attribute

Firmware 2.2.0 supports **running multiple zones in sequence (serialized, or one after another) or parallel (concurrently)**. This can be set using each zone's **Sequential Group** attribute. Zones that are assigned in the same sequential group will be automatically serialized. For example, if zones 1, 2, 3 are in sequential group A, the controller will make sure only one of them is running at any given time. If their scheduled times overlap, for example, zone 2 is scheduled to open while zone 1 is still running, it will be automatically pushed behind zone 1. This is the most common way as it helps conserve the water pressure by running one zone at a time.

Zones belonging to different sequential groups can run in parallel. For example, if zones 4, 5, 6 are in sequential group B, these three zones will be automatically serialized, however, they are not serialized with zones 1, 2, 3 in group A. So these two groups can run simultaneously with respect to each other. In other words, a program that runs zones 1, 2, 3 can run alongside another program that runs zones 4, 5, 6. In previous firmwares, each zone has a 'sequential' attribute, this is equivalent to having only one sequential group. Therefore this firmware generalizes the 'sequential' attribute by allowing more sequential groups and thus more flexibility in running zones simultaneously.

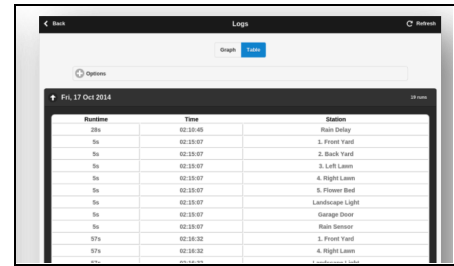
If a zone is assigned to the Parallel (P) group, it can run simultaneously with any other zones. In other words, the zone can run at any time regardless of whether other zones are running or not. This is equivalent to turning the 'sequential' attribute off in previous firmwares. Simultaneous zone runs are particularly useful when you are using OpenSprinkler to switch non-sprinkler devices, such as lights, pump, and heater, which should not be serialized with sprinkler zones.

8. Logging

OpenSprinkler supports logging, which stores zone runs as well as rain delay, sensor status changes, and watering percentage changes.

At the home page, click **View Logs** (or ALT+L) to see a graphical plot of the log:

- In the **Options** tab, select the start and end dates of the query. The default is the past 7 days.
- Select **Table** at the top of the page to switch to a table view of the data.



The screenshot shows the 'Logs' page in the OpenSprinkler web interface. At the top, there are navigation buttons for 'Back', 'Options', and 'Refresh'. Below that, there are input fields for 'Start' and 'End' dates, and a 'Table' button. The main content is a table with columns for 'Runtime', 'Time', and 'Station'. The table contains several rows of log entries, including zone runs and sensor status changes.

| Runtime | Time | Station |
|---------|----------|-----------------|
| 0% | 02:18:05 | Rain Delay |
| 0% | 02:18:07 | 1. Front Yard |
| 0% | 02:18:07 | 2. Back Yard |
| 0% | 02:18:07 | 3. Left Lawn |
| 0% | 02:18:07 | 4. Right Lawn |
| 0% | 02:18:07 | 5. Flower Bed |
| 0% | 02:18:07 | Landscape Light |
| 0% | 02:18:07 | Garage Door |
| 0% | 02:18:07 | Rain Sensor |
| 0% | 02:18:32 | 1. Front Yard |
| 0% | 02:18:32 | 4. Right Lawn |

9. Firmware Update, FAQ, Support, and Github

Go to www.opensprinkler.com to find [Firmware Update Guide](#) and detailed [FAQs](#). You can submit a support ticket on our [Support Website](#) and also find various support articles, including details of the weather adjustment methods, IFTTT, and firmware compilation instructions. OpenSprinkler is a fully open-source product. Source code and all hardware design files are available at the [OpenSprinkler Github repository](#).

Specifications

| | OpenSprinkler v3.x | OpenSprinkler Pi (OSPi) |
|---------------------------------|--|-------------------------|
| Input Voltage: | 22~28V AC (AC-powered version and OSPi); 7.5~12V DC (DC and LATCH versions). | |
| Power Consumption: | 1~1.5 Watt | |
| Number of Zones: | 8 on the main controller, expandable to 72 zones (or 200 zones on OSPi) | |
| Max Output Current: | 800mA continuous per zone (AC); 2A continuous per zone (DC). | |
| Over-voltage Protection: | Bi-directional TVS and RC snubber on each zone and power input | |
| Product Size: | 140mmx68mmx34mm | 135mmx105mmx38mm |
| Product Weight: | 140g | 200g |
| Expander Size: | 130mmx75mmx25mm | |
| Expander Weight: | 100g | |

Advanced Topics

Installing Radio Frequency (RF) Transmitter

OpenSprinkler supports standard 434MHz or 315MHz Radio Frequency (RF) transmitters. This can be used to replicate RF signals sent to remote power sockets for switching powerline devices, such as lights, heaters, fans, pumps. To do so, you need to purchase an [RFtoy](#) and use it to decode RF signal code from your remote power sockets. Each code is a 16-digit long string (e.g. 51001A0100BA00AA) that encodes the on-signal, off-signal, and signal timing information.

- The RFToy package contains a pair of 433MHz transmitter and receiver, and also a pair of 315MHz ones. Pick the one that matches the frequency of your wireless device. To maximize transmission range, you should **solder a 17cm-long wire** antenna to the **ANT pin** on the transmitter. The wire can be straight or curled like shown in the picture below.
- **OpenSprinkler v3** has a built-in receptacle pin header for the RF transmitter. You can directly plug in an RF transmitter (3-pin) on the top side. The orientation is such that **it's facing up**. See the [Hardware Interface](#) diagram for the locations of the RF pin header.
- **OSPi** do not have a receptacle pin header for the transmitter, but PCB pins. You can solder a transmitter to the pins. Specifically, open the enclosure, then locate the RF pins on the circuit board: **A3 (or DATA), VIN, GND**, and solder the transmitter to the corresponding pins. Finally, carefully reassemble the enclosure.
- Additional details about the RF Station Feature can be found on this [blog post](#).

