

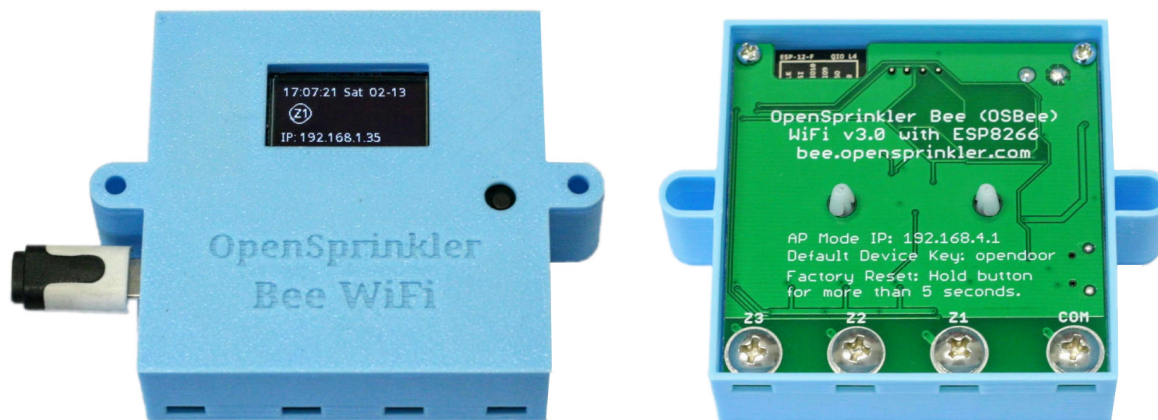
Opensprinkler Bee WiFi 3.0 User Manual (firmware 1.0.2)

(Oct 2022)

1. Introduction

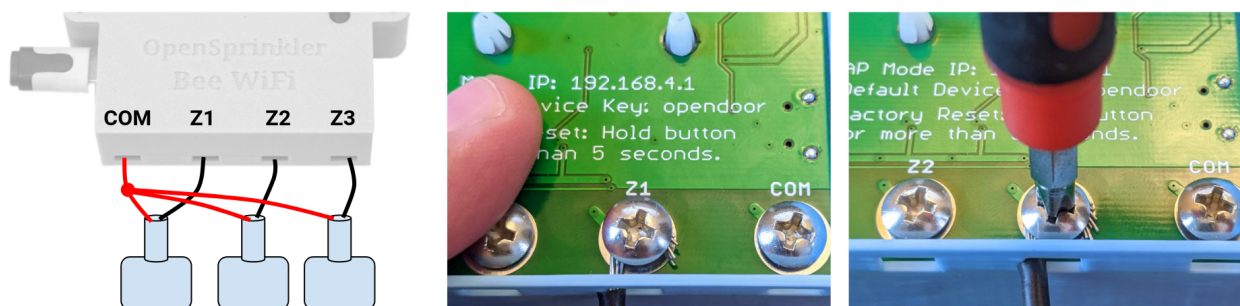
OpenSprinkler Bee (OSBee) 3.0 is an open-source, WiFi-enabled sprinkler controller for **latching** solenoid valves. It is suitable for garden and lawn watering, flower and plant irrigation, hydroponics, and other types of watering projects. It comes with built-in WiFi, OLED display, laser cut acrylic enclosure, and can switch up to **3 zones** independently. It is primarily designed for **latching** solenoid valves, although with a simple modification it can also operate **non-latching** valves (e.g. standard 24VAC sprinkler valves), low-voltage fish tank pumps, and other types of low-voltage DC or AC valves and pumps.

The package includes one assembled and tested OSBee circuit board in 3D printed enclosure, USB cable, and (optionally) a USB power adapter (5VDC output, minimum 1A current).



2. Hardware Setup

OSBee has four terminals marked **COM** (common), **Z1** (zone 1), **Z2**, and **Z3**. The positive (+) wire (often colored red) of each valve should be connected together and go to the COM terminal; the negative (-) wire (often colored black) of each valve should go to an individual zone (1, 2, or 3). At the back of the controller there are four big screws. Un-tighten a screw, insert the wire through the opening below it, wrap the wire around the screw, then tighten the screw. OSBee is powered by a USB adapter through a microUSB cable.



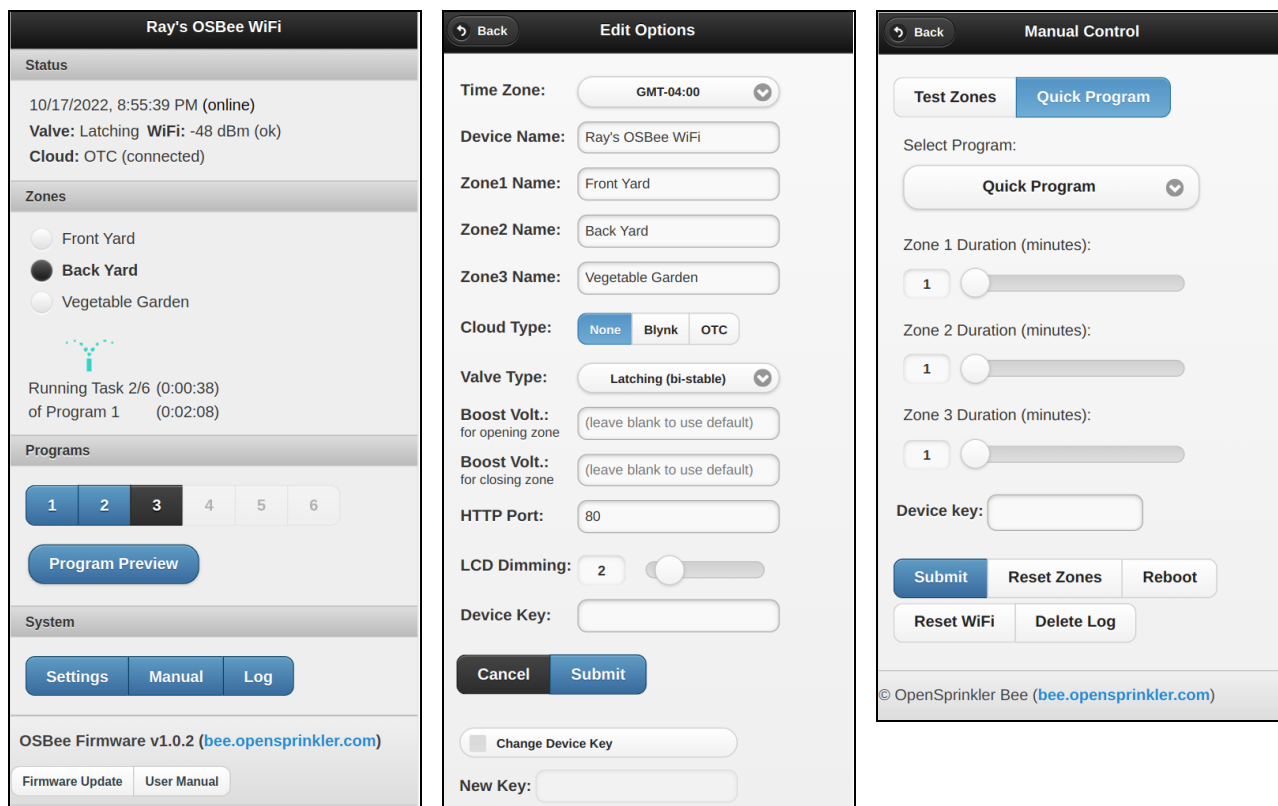
3. Software Setup

The first time you power on OSBee, or after each factory reset, the controller starts in AP (Access Point) mode. In this mode, OSBee creates a WiFi SSID, the name of which is printed on the LCD screen (e.g. OSB_XXXXXX). Use your smartphone, or laptop to connect to this WiFi SSID (there is no WiFi password). Once connected, open a web browser on your phone (or laptop) and type in the IP address 192.168.4.1. This should open the WiFi configuration page. The main purpose of the WiFi configuration is to let OSBee know your home WiFi's name and password, so that it can subsequently connect to your WiFi. So choose, or directly type in your home WiFi's SSID and WiFi password (Note: it only connects to 2.4G WiFi). Click on

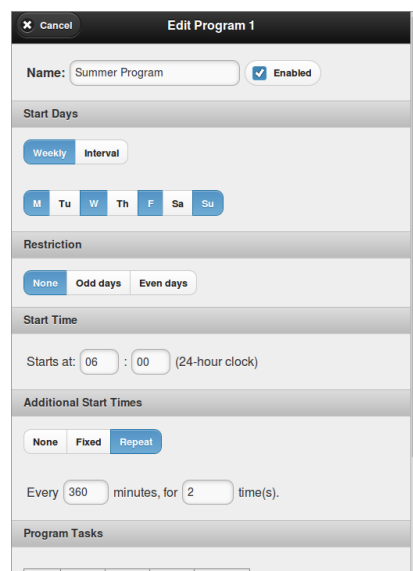
'Submit'. At this point, OSBee will attempt to connect to your WiFi, and if successful, it will reboot itself, and now WiFi configuration is complete. From now on, it remembers your WiFi and will always attempt to connect to it when it's powered on. If you want to change the WiFi network, you can perform either a WiFi reset or factory reset, and it will go back to AP mode.

In WiFi station mode, OSBee obtains an IP address from your home WiFi router. This IP address is printed on the LCD screen at the bottom. Open a browser and type in the IP address, it should open OSBee's homepage shown below. The **default device key is opendoor**, which you can change in Settings.

4. Using the Built-in Web Interface

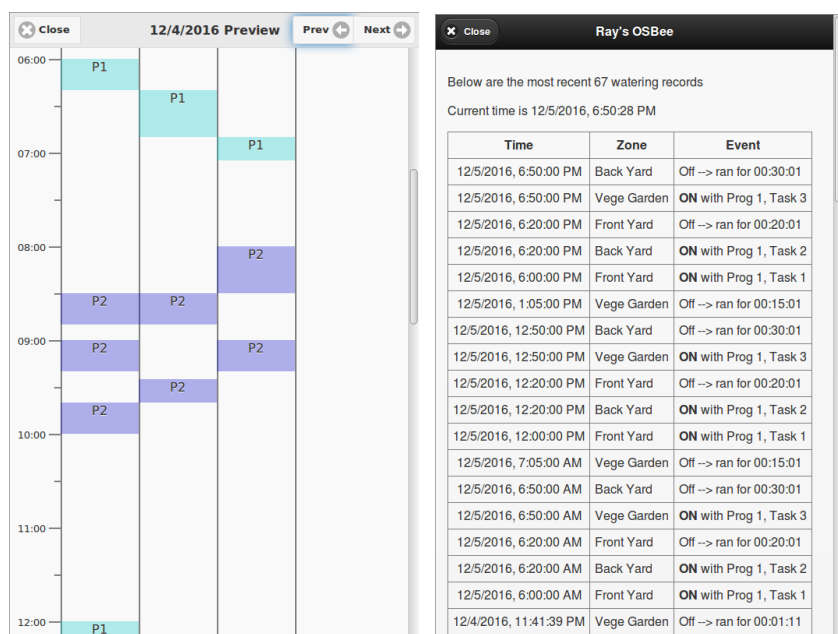


The **Homepage** (left image above) shows the current time, status of each zone, and buttons that lead to other pages. Click on "Settings" at the lower-left of the homepage, this will open the **Settings/Options** page (middle image above), where you can configure the time zone, device name, zone name, and cloud connection options. At the homepage, you can also click on "Manual" to open the **Manual Control** page, where you can start a test program or any of the existing programs, reboot the controller, reset zones, delete log, or trigger WiFi reset. Again, the **default device key is opendoor**.



Back to the homepage, click on a **Program** button (the **black-colored** button creates a new program, and **blue-colored** buttons are existing programs) to add a new program or edit an existing program. The program editing interface (shown on the left) allows you to change the program name, configure it as a weekly or interval program, set odd/even day restrictions, set the first start time and additional start times. Each program consists of a number of **program tasks**. First append a new task, then click a zone to enable or disable that zone from the task, and finally set the duration. Program tasks are flexible: you can set multiple zones to turn on at the same time, and you can have the same zone turn on multiple times in different tasks. You can also leave all zones off in a task, to create a delay/gap for a specified

amount of duration. To delete or edit an existing task, click the index of the task, which will highlight that task in yellow, then you can edit zones or the duration again. When you are done, click on Submit to finish editing the program.



The homepage also has a **Program Preview** button which opens a new page showing a graphical preview of the programs. At the top-right corner of the program preview page, you can navigate to different days.

The **Log** button opens a new page showing the history of recent watering events, including the time, zone, and program/task information of each event. If you want to trigger a software reboot of the controller, go to Settings pages, and use the 'Reboot' button there.

LCD and button functions: OSBee has a built-in OLED display. It shows the current time, and zone status. At the bottom it shows the IP address. Clicking the black pushbutton (lower-right to the LCD) will cycle through additional information, such as MAC address etc.

If you press and hold the button for 3 seconds but less than 6 seconds, it triggers a **WiFi Reset** where the controller goes back to Access Point (AP) mode, so you can reconfigure WiFi. Existing settings, programs, and log data are preserved. If you press and hold the button for more than 6 seconds, it triggers a **Factory Reset** where the controller goes back to AP mode, and all settings/programs/log data are erased.

Firmware Update: when new firmware becomes available, you can either update firmware over WiFi (at the homepage, lower-left corner, **Update Firmware** button; or in a web browser, type controller's IP address followed by /update); you can also update a new firmware using the microUSB port (the controller has built-in USB serial). Details on how to update firmware through USB can be found on OSBee's Github page: <https://github.com/OpenSprinkler/OSBeeWiFi-Firmware>

Customize boosted voltage for valve opening/closing: by default OSBee generates a boosted voltage of 21V to open/close a latching solenoid valve. This usually works fine for all latching valves, regardless of the brand/type. However, some solenoid valves require a different voltage for opening the valve vs. closing the valve. This voltage can be customized in the Settings page, where you can specify a different voltage for opening vs. closing.

Interfacing with Non-Latching solenoids and low-voltage DC pumps: although OSBee is designed primarily for latching solenoid valves, it CAN work with non-latching solenoid valves as well (such as standard 24VAC sprinkler valves, low-voltage DC pumps such as fish tank pumps, and other low-voltage valves) with a simple modification. To do so, 1) locate the **NL Jumper** (non-latching jumper) on the circuit board (close to the microUSB port), solder that jumper so that the two pins are connected; and 2) in Settings set the valve type to 'Non-Latching'. When operating in non-latching mode, once the valve is open, the controller will continue supplying holding current to the valve to keep it open. **NOTE:** never connect a

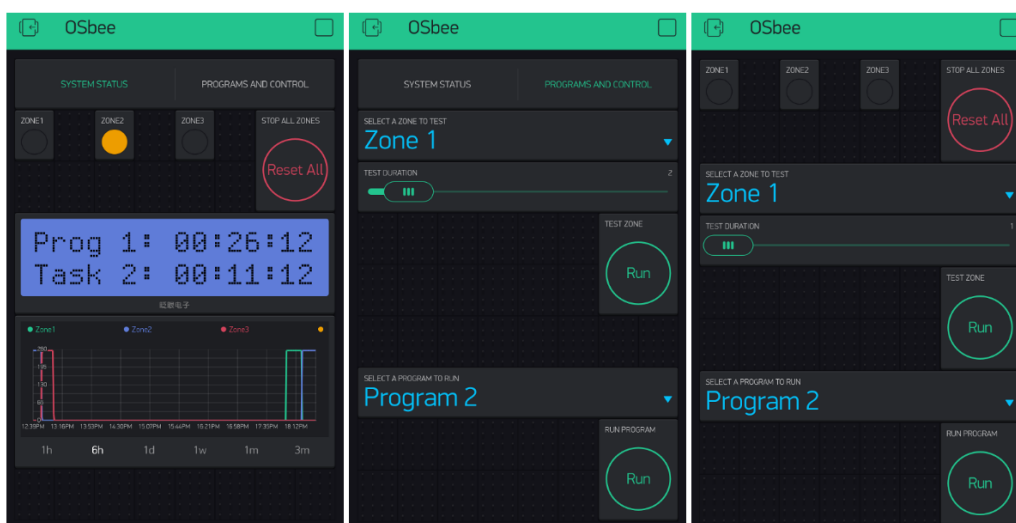
latching solenoid to the controller when it's configured in non-latching mode -- as latching solenoids have very low resistance, using them in non-latching mode will result in shorting.

5. Cloud Connection and Remote Access

OSBee firmware supports remote access using either Blynk legacy app or OpenThingsCloud (OTC) connection. Blynk allows you to remotely check the controller's current status, and run a program. OTC allows remote access of the full built-in user interface.

1. Using Blynk legacy app: first you need to install the Blynk legacy app on your smartphone. This app is no longer available in the app store. If you are an Android user, you can still search and find its apk and install it manually. Once you have the app installed, scan the OSBee Blynk project QR code, available at: <https://github.com/OpenSprinkler/OSBeeWiFi-App/tree/master/Blynk> this will import the project to your Blynk app.

Once the Blynk project is created, you can go to the project settings to obtain the cloud token. Then paste this token at OSBee's Settings page, submit, and reboot OSBee controller. This way the firmware will communicate with the Blynk cloud using the token, and allow the Blynk app to access the controller remotely, even if you are not at home.



2. Using OTC: using OTC does NOT need any app installation. First, if you don't already have an account at opensprinkler.com please go there and click on 'Register' at the top to register an account. Then go to openthings.io, use your opensprinkler.com account to log in. Then at the left side, click 'My OpenThings OTC devices'. Type in a descriptive name, then click 'Add New Device'. This will create a new token. Copy this token, and go to your OSBee's Settings page, select **OTC** as cloud connection option, and paste the token there. Then save and reboot the controller.

To remotely access the built-in user interface, use:

https://cloud.openthings.io/forward/v1/your_token

where your_token is the token you created. You can bookmark this link, or add it to your smartphone's homescreen, so it can work almost like a native app.

6. Specification and Open-Source Links

Input voltage (typical):	5VDC through USB
Input voltage (maximum):	12VDC (e.g. if using solar power)
Current consumption:	80~140mA (depending on WiFi signal strength)
Power consumption:	input voltage x current (typically 5V x 100mA = 0.5 Watt)
Product dimension:	63mm x 63mm (2.5in x 2.5in)
Product weight:	50g (1.7oz)
Hardware components:	ESP8266 (MCU+WiFi), MC34063 (voltage booster), PCF8563 (RTC), CH340C (USB serial), SSD1306 (OLED), 4x half H-bridges.

OpenSprinkler Bee is completely open-source. Its hardware design files, firmware code, and Blynk project QR code can be found at the following Github repositories:

- <https://github.com/OpenSprinkler/OSBeeWiFi-Hardware>
- <https://github.com/OpenSprinkler/OSBeeWiFi-Firmware>
- <https://github.com/OpenSprinkler/OSBeeWiFi-App>