



OurWeather and WeatherPlus Advanced Usage Manual

January 2018

| | |
|--------------------------------------------------------|----|
| Revisions | 3 |
| Using the REST Interface built into OurWeather | 3 |
| Testing the REST Interface | 3 |
| REST Variables | 3 |
| Structure of the FullDataString Return Value | 8 |
| REST Commands | 10 |
| How to Program Your OurWeather | 14 |
| Installation of the IDE | 14 |
| Download the WeatherPlus Software for OurWeather | 14 |
| To Set the WeatherPlus Board into Bootload Mode | 15 |
| Settings for Alexa and OurWeather | 15 |
| Settings for WXLink / Solar WXLink | 15 |
| Settings for OurWeather Solar Power Extender Kit | 15 |
| Setting up the OurWeather Twitter Interface | 15 |
| Setting up the OurWeather CWOP Interface | 16 |
| Setting up the OurWeather WeatherUnderground Interface | 16 |

Revisions

Version 3.0 - January 2018 - Added Alexa to OurWeather Interface Information

Using the REST Interface built into OurWeather

Set up your OurWeather system as shown in the OurWeather Assembly Manual.

Testing the REST Interface

Now that you have OurWeather setup and working, it's time to test the first REST command.

Open a browser (Chrome, Safari, Internet Explorer or Edge - if your browser doesn't work, download and install Chrome) and type in the following (substitute your OurWeather IP address for 192.x.x.x):

```
http://192.168.1.118/FullDataString
```

returns:

```
{"FullDataString":  
"21.30,36.70,25.63,101714.00,620.44,0.00,0.00,0.00,0.70,0.00,0.00,0.00,0.00,0.00,0.04/2  
4/2016 11:56:10,SwitchDoc Labs", "id": "1", "name": "OurWeather", "connected": true}
```

Which contains a bunch of weather information from the OurWeather station. Note: Your FullDataString will have more information in it if you are using software version 019 or later.

After doing that, you now have access to all of these REST commands.

The supported REST commands are:

REST Variables

For example, to access the OurWeather REST variables type into a browser:

```
http://192.168.1.118/FullDataString
```

returns:

```
{"FullDataString":  
"21.30,36.70,25.63,101714.00,620.44,0.00,0.00,0.00,0.70,0.00,0.00,0.00,0.00,0.00,0.04/2  
4/2016 11:56:10,SwitchDoc Labs", "id": "1", "name": "OurWeather", "connected": true}
```

| REST Variables | | |
|-----------------------|-------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------|
| Variable | Return | Example |
| OurWeatherTime | OurWeather Time and Date | <pre>{"OurWeatherTime": "05/21/2016 10:09:59", "id": "1", "name": "OurWeather", "connected": true}</pre> |
| FullDataString | String of Weather Data | See description of the FullDataString after this table |
| FirmwareVersion | Which software version is OurWeather running | <pre>{"FirmwareVersion": "016", "id": "1", "name": "OurWeather", "connected": true}</pre> |
| IndoorTemperature | Temperature sensor on BMP280 on the WeatherPlus board. Always reports in degrees C. | <pre>{"IndoorTemperature": 27.47, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| BarometricPressure | Barometric Pressure sensor on BMP280 on the WeatherPlus board. Always reports in Pascals. | <pre>{"BarometricPressure": 100958.00, "id": "1", "name": "OurWeather", "connected": true}</pre> |

| REST Variables | | |
|-----------------------|---------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| Altitude | Altitude calculated from BMP280. Always reports in meters. | <pre>{"Altitude": 682.52, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| OutdoorTemperature | Outdoor Temperature from the AM2315. Always reports in degrees C. | <pre>{"OutdoorTemperature": 23.60, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| OutdoorHumidity | Outdoor Relative Humidity from the AM2315. Always reports in % RH. | <pre>{"OutdoorHumidity": 35.70, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| CurrentWindSpeed | Current Wind Speed from the WeatherRack - Always reports in kph (kilometer per hour). | <pre>{"CurrentWindSpeed": 0.00, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| CurrentWindGust | Current Wind Speed from the WeatherRack - Always reports in kph (kilometer per hour). | <pre>{"CurrentWindGust": 18.00, "id": "1", "name": "OurWeather", "connected": true}</pre> |

| REST Variables | | |
|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| CurrentWindDirection | Current Wind Direction from the WeatherRack - Always reports in degrees. 0 = due North. | <pre>{"CurrentWindDirection": 90.00, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| EnglishOrMetric | Is the Weather DISPLAY set to English or Metric units. 0 means English units, 1 means Metric units. The REST interface always sends Metric unit. | <pre>{"EnglishOrMetric" : 0, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| RainTotal | Current Rain Total since last reboot of WeatherPlus. Always reports in mm. | <pre>{"RainTotal": 0.28, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| WindSpeedMin | Wind Speed Minimum as calculated by WeatherPlus over the past 50 seconds. Always reports in kph. | <pre>{"WindSpeedMin": 0.00, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| WindSpeedMax | Wind Speed Maximum as calculated by WeatherPlus over the past 50 seconds. Always reports in kph. | <pre>{"WindSpeedMax": 2.50, "id": "1", "name": "OurWeather", "connected": true}</pre> |

| REST Variables | | |
|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| WindGustMin | Wind Gust Minimum as calculated by WeatherPlus over the past 50 seconds. Always reports in kph. | <pre>{"WindGustMin": 0.00, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| WindGustMax | Wind Gust Maximum as calculated by WeatherPlus over the past 50 seconds. Always reports in kph. | <pre>{"WindGustMax": 6.50, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| WindDirectionMin | Wind Direction Minimum as calculated by WeatherPlus over the past 50 seconds. Always reports in degrees - 0 is due north. | <pre>{"WindDirectionMin": 0.00, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| WindDirectionMax | Wind Direction Minimum as calculated by WeatherPlus over the past 50 seconds. Always reports in degrees - 0 is due north. | <pre>{"WindDirectionMax": 275.00, "id": "1", "name": "OurWeather", "connected": true}</pre> |
| | | |
| AirQualitySensor | Instantaneous Air Quality. Reports a value from 0 (very clean) to ~32000 (very dirty). See the Our Weather Air Quality Extender Manual on www.switchdoc.com on the store. | <pre>{"AirQualitySensor": 1400, "id": "1", "name": "OurWeather", "connected": true}</pre> |

Structure of the FullDataString Return Value


```
{"FullDataString":  
"21.30,36.70,25.63,101714.00,620.44,0.00,0.00,0.00,0.70,0.00,0.00,0.00,0.00,0.00,0.00,0.04/2  
4/2016 11:56:10,SwitchDoc Labs,  
0,4,3.62,35.20,2.78,0.00,4.96,25.20,0.00,0.00,0.00,0.00,0.00,0.00,V:0,WXLMB ","", "id": "1",  
"name": "OurWeather", "connected": true}
```

The values in the FullDataString are all in metric. See metric units in the above table.

- Outdoor Temperature
- Outdoor Humidity
- Indoor Temperature
- Barometric Pressure
- Altitude
- Current Wind Speed
- Current Wind Gust
- Current Wind Direction
- Rain Total
- Wind Speed Minimum
- Wind Speed Maximum
- Window Gust Minimum
- Window Gust Maximum
- Wind Direction Minimum
- Wind Direction Maximum
- Is Display English (1) Or Metric (0)
- Current Date/Time on OurWeather
- OurWeather Station Name
- Current Air Quality Sensor Reading
- Current Air Quality Qualitative Reading
- SunAirPlus Battery Voltage
- SunAirPlus Battery Current
- SunAirPlus Solar Panel Voltage
- SunAirPlus Solar Panel Current
- Load Voltage (into OurWeather)
- Load Current (into OurWeather)
- WXLink Battery Voltage
- WXLink Battery Current
- WXLink Solar Panel Voltage
- WXLink Solar Panel Current
- Always 0.00
- WXLink Load Current
- WXLink AM2315 Temperature Validation IVF: Means Invalid Temperature Found, V: Means Valid found
- WXLink Message Status: WXLMSG - WXLink Last Message Good, WXLMB - WXLink Last Message Bad
- Alexa Enabled (1) or Not (0)

Note: As more devices are added and supported by OurWeather, this string will continue to increase beyond these values, but only at the end of the string.

REST Commands

| REST Commands | | | |
|---------------|-------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| Variable | Function | Parameters | Returns |
| arduino | For ArduinoiConnect | Not Implemented | |
| | | | |
| | External commands | | |
| | | | |
| led | Controls red LED connected to GPIO0 on the WeatherPlus Board | http://192.168.1.118:/LED?params=1 1 means on, 0 means off | {"return_value": 1, "id": "1", "name": "OurWeather", "connected": true} |
| setID | sets the ID of the OurWeather at the IP address. Returned in "id" field | http://192.168.1.118/setID?params=2 Parameter is the new ID. Alphanumeric allowed | {"return_value": 1, "id": "2", "name": "OurWeather", "connected": true} - Note new ID# |

REST Commands

| | | | |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| resetOurWeather | Restarts OurWeather to default. Works most of the time on the ESP8266. | http://192.168.1.118/resetOurWeather?params=adminpassword Parameter is administration password | return_value will be 1 if reset command was accepted, 0 if not (as in password failure) |
| setAdminPassword | reset Administration Password | http://192.168.1.118/setAdminPassword?params=oldpassword,newpassword Parameters are old and new password, separated by a “,” | return_value will be 1 if the command succeeded and 0 if it did not (such as invalid password) |
| setDateTime | sets Data and Time on the DS321 Real Time Clock on WeatherPlus | http://192.168.1.118/setDateTime?params=password,Jan 31 2016,14:03:00 Admin password is first parameter. Note the exact non-tolerant format of the data time. Spaces are important. | As of OurWeather software version 017, return_value will be 0 if password fails, 2 if the date or time is bad and 1 if the set date/time succeeds |
| resetToDefaults | reset Our Weather to default values | Not Implemented | Not Implemented |
| resetWiFiAccessPoint | resets OurWeather Access Point values to initial values. OurWeather will be at 192.168.1.4 and act as an access point. See OurWeather Assembly Manual to set up. | http://192.168.1.118/resetWiFiAccessPoint?params=adminpassword Parameter is administration password | return_value is 1 if it succeeds (and you will never receive it if it does) or 0 if the password fails. |

REST Commands

| | | | |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| updateOurWeather | Does an Over-The-Air (OTA) on OurWeather from the latest software located at www.switchdoc.com | http://192.168.1.118/?params=adminpassword Parameter is administration password | return_value is 0 if the password fails, 1 if the update failed, 2 if there are no updates, 3 if it succeeds. You generally will not receive 3 as the ESP8266 will be rebooting. |
| | External Interfaces | | |
| enableCWOP | Enables the CWOP interface for OurWeather | Not Implemented | Not Implemented |
| enableTwitter | Enables the Twitter Interface for OurWeather | Not Implemented | Not Implemented |
| | | | |
| | Weather Display Functions | | |
| WeatherSmall | Sets the OurWeather OLED display to small characters | http://192.168.1.118/WeatherSmall | return_value is 1 |
| WeatherMedium | Sets the OurWeather OLED display to medium characters | http://192.168.1.118/WeatherMedium | return_value is 1 |
| WeatherLarge | Sets the OurWeather OLED display to large characters | http://192.168.1.118/WeatherLarge | return_value is 1 |

REST Commands

| | | | |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| WeatherDemo | Sets the OurWeather OLED display to Demo mode. Senses and moves fast showing values that people can quickly change like outside temperature/ humidity, wind speed, direction and rain. Used in classroom demonstrations where the instruments are located for people to change and turn. | http://192.168.1.118/WeatherDemo | return_value is 1 |
| EnglishUnits | Sets the OurWeather OLED display to show English Units | http://192.168.1.118/EnglishUnits | return_value is 1 |
| MetricUnits | Sets the OurWeather OLED display to show Metric | http://192.168.1.118/MetricUnits | return_value is 1 |
| EnablePubNub | Enable/disable PubNub (Alexa) on OurWeather - sets the Pub and Sub keys | http://http://192.168.1.102/EnablePubNub?params=admin,1,pub-c-cc4d6662-190a-42af-a14d-3be4e6040ff6,sub-c-47725a16-f0e7-11e7-9869-a6bd95f83dd1 Admin password is first parameter. Second parameter is Enable (1) or Disable (0) Alexa. Third Parameter is the MQTT PubNub Publish Key and the Fourth Parameter is the MQTT PubNub Subscribe Key | As of OurWeather software version 026, return_value will be 0 if password fails, 1 if the set pub/sub key succeeds |

| REST Commands | | | |
|-----------------|----------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|
| SendPubNubState | Sends an MQTT Message to the PubNub site. Great to start things out. | http://192.168.1.102/SendPubNubState?params=admin First Parameter is the Admin password | return value is 1 |

How to Program Your OurWeather

Download the Arduino IDE 1.6.9 (or higher) to start from arduino.cc

Installation of the IDE

With the new 1.6.9+ releases of the Arduino IDE, it is much simpler to add and manage new boards. In the case of the Adafruit Huzzah ESP8266, it a simple process. Rather than repeat the tutorial here, we would suggest following the [Adafruit tutorial](#).

Use Version 1.6.9+ of the Arduino IDE for better results.

Download the software at <https://learn.adafruit.com/adafruit-huzzah-esp8266-breakout/using-arduino-ide>

Make sure you have selected Adafruit Huzzah as the board under the tools menu.

Download the WeatherPlus Software for OurWeather

Move to where you wish to store your Arduino program and use the command line to clone the OurWeather/WeatherPlus software

```
git clone https://github.com/switchdoclabs/OurWeatherWeatherPlus
```

This will create a directory called SDL_ESP8266_WeatherPlus.

Now use the Arduino File menu to open the Compile and download the program to the WeatherPlus Board .

To Set the WeatherPlus Board into Bootload Mode

- Hold down the **GPIO0** button, the red LED will be lit
- While holding down **GPIO0**, click the **RESET** button
- Release **RESET**, then release **GPIO0**
- When you release the RESET button, the red LED will be lit dimly, this means its ready to bootload

Then compile and load the software. Make sure you have selected the “Adafruit Huzzah” board under the tools menu.

Settings for Alexa and OurWeather

See the Tutorial on switchdoc.com

<http://www.switchdoc.com/2018/01/tutorial-voice-time-ourweather-and-amazon-alexa-part-1/>

Settings for WXLink / Solar WXLink

See the OurWeather Solar WXLink Manual

Settings for OurWeather Solar Power Extender Kit

See OurWeather Solar Power Extender Kit Manual

Setting up the OurWeather Twitter Interface

Not Implemented as of OurWeather Software Version 021

Setting up the OurWeather CWOP Interface

Not Implemented as of OurWeather Software Version 021

Setting up the OurWeather WeatherUnderground Interface

See the OurWeather WeatherUnderground Manual