

Weather Station Raspberry Pi Software Setup

For the Standard and Pro Version of the MyWX Android App

V2.3.3

By David W. Enström

Contents

Weather Station Raspberry Pi Software Setup	1
Raspberry Pi.....	2
Setup Raspbian for WeeWX and the BC Robotics Hardware	2
Setup FTP	4
Setup a Static IP Address.....	4
Setup All Required Software	5
Setup Remote Desktop Access	5
Installing Required Libraries	6
Install Python V3.x	6
Install the Adafruit Python GPIO Library	6
Install the Adafruit BME280 Library	6
Install the ADS1x15 Library, the ADS1015 Analog to Digital chip (wind direction)	6
Setup the DS18B20 temperature sensor and install the software library.....	6
WeeWX Setup.....	7
Updating the WeeWX Software	9
Configuring MySQL (Maria Database).....	9
Configuring WeeWX.....	9
Raspbian Command Summary	10
BCRobotics Driver Installation.....	11
Station data.....	14
Enable Remote Access to WeeWX Device	15
Configuring PKI Security	15
Backup WeeWX	16
Weather Formulæ.....	17
Wind Direction.....	17
Wind Speed.....	17
Wind Chill and Feels-Like Temperature.....	18
Wind Chill.....	18
Feels-Like Temperature	18
Temperature	18
Humidity	19
Pressure	19
Rainfall.....	19
Wunderground	19
Raspberry Pi Documentation	20

This personal weather station uses the SparkFun weather sensors (Argent Data Systems), and the BC Robotics add-on hardware (HAT) to interface these sensors to the Raspberry Pi.

See: <https://www.sparkfun.com/products/8942> and <https://www.bc-robotics.com/tutorials/raspberry-pi-weather-station-part-1/>

The weather station monitors (and optionally shares the data with Wunderground):

- **Wind Speed** – in km/h
- **Wind Direction** – in degrees
- **Rainfall** – in mm
- **rainRate** – in mm/hr
- **Temperature** – in degrees C
- **Air Pressure** – in hecto Pascals
- **Humidity** – in percent

Instructions for constructing this weather station and all the required components are in the “Weather-Station-Raspberry-Pi-construction.pdf” PDF file.

Raspberry Pi

The weather station is connected to a Raspberry Pi ... it is actually a HAT (Hardware Attached on Top). The HAT is from BC Robotics and uses the BME280 temperature, pressure, and humidity sensor, the DS18B20 Digital temperature sensor, and the SparkFun Weather sensors from Argent Data Systems for wind and rain.

Raspbian does not have a default login, so create your own during the setup:

- **Default username:** *user*
- **Password:** *password*

Setup Raspbian for WeeWX and the BC Robotics Hardware

The following assumes that you already have Raspbian running on your Pi. If not, follow these instructions:

If setting up your SD card on Windows get the `balenaEtcher` flash utility here. I use the “portable” version, which does not require an install on Windows:

<https://www.balena.io/etcher/>

Install this utility or run Etcher from this downloaded file.

Now download the latest “Raspberry Pi OS with desktop” image from here:

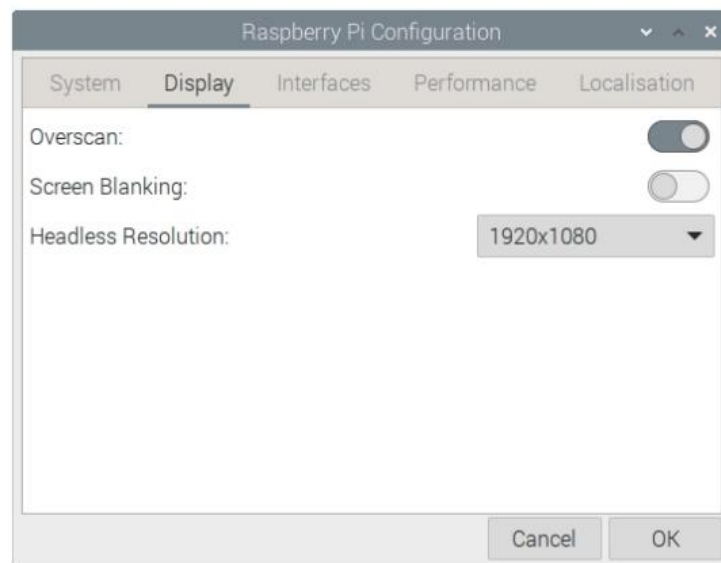
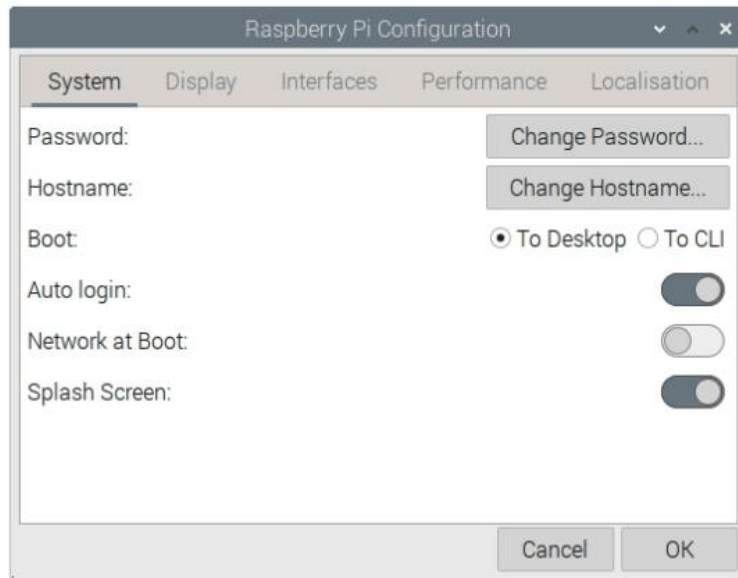
<https://www.raspberrypi.org/downloads/raspberry-pi-os/>

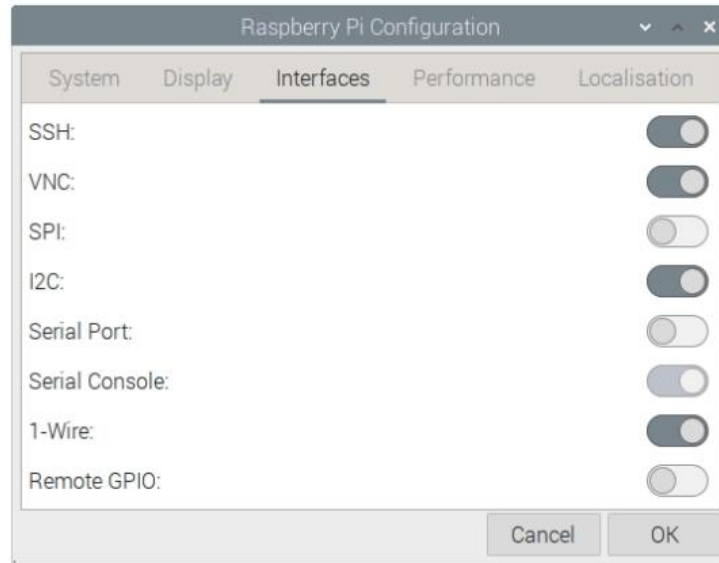
Unpack the Raspbian image, and then use the `balenaEtcher` flash utility to flash it to your SD card. Now insert the card into your Raspberry Pi and follow the instructions when it boots up, which may take a while. Setup the country, Wi-Fi network, and other settings. Then let the system update itself. Also, see:

<https://www.raspberrypi.org/documentation/installation/installing-images/>



On the main screen, go to Preferences / Raspberry Pi Configuration and define your Hostname on the System tab, and then enable SSH, VNC, I2C and 1-Wire interfaces on the Interface tab, as shown below (your choice for Hostname):





Now setup a few configurations on your Pi to make it easier to use and manage. VNC allows for remote virtual terminal access, which is necessary to manage the system remotely.

Setup FTP

Install the `vsftpd` FTP service (commands below), and then enable FTP write access to the Pi; edit `etc/vsftpd.conf` and uncomment the `write_enable` line (see below). Start the File Manager, find the `etc` folder.

```
sudo apt update
sudo apt full-upgrade
sudo apt install vsftpd      # install FTP
sudo mousepad vsftpd.conf    # go to etc folder: enable writing, etc.
    write_enable=YES          # i.e., remove the "#"
    local_umask=022           # Most FTP apps use this setting
```

Setup a Static IP Address

The Raspberry Pi is setup with a static IP, to connect to and manage the Pi easier. Do this by editing the following file (`sudo mousepad dhcpd.conf`), or right click on the Wi-Fi icon and open the “Wireless & Wired Network Settings” utility:

```
etc/dhcpd.conf
```

Also setup the static DNS server and Router IP addresses, for example:

```
static ip_address=10.0.0.70
static routers=10.0.0.1
static domain_name_servers=208.67.220.220
static domain_search=208.67.222.222
```

Note: If the syslog continuously registers approval messages:

```
dhcpd[361]: eth0: received approval for 10.0.0.200
```

then edit the `dhcpcd.conf` file (`sudo mousepad dhcpcd.conf`) and comment out the `inform` line.

Setup All Required Software

Ensure that the prerequisites are installed (Prerequisites for Debian 10 (Buster), Raspbian 10, Ubuntu 18.04-19.10):

```
sudo apt update
sudo apt-get upgrade
```

Python V3 installation (check to make sure that it is installed)

```
sudo apt install python3-configobj
sudo apt install python3-pil
sudo apt install python3-serial
sudo apt install python3-usb
sudo apt install python3-pip
```

Add the cheerah library

```
sudo apt install python3-cheetah
```

Optional: for extended almanac information

```
sudo apt install python3-ephem
```

Required: to use the MySQL (MariaDB):

```
sudo apt-get install mariadb-client
sudo apt-get install mariadb-server
sudo apt-get install python3-mysqldb
```

Setup Remote Desktop Access

VNC

VNC (Virtual Network Computing) is a graphical desktop-sharing system that uses the Remote Frame Buffer protocol to control another computer remotely. It comes bundled with the full Raspberry Pi software install, and we turned it on in the initial setup described above.

You can install the VNC Viewer on your Windows PC to access and administer your Pi over the network:

<https://www.realvnc.com/en/connect/download/viewer/>

XRDP

Xrdp is an open source remote desktop protocol server, which allows you to connect to the Raspbian desktop from any operating system. VNC has limited support, therefore `xrdp` is recommended. To install it:

```
sudo apt-get install xrdp
```

xrdp is present on raspberry as a service, so you can manage it with the usual controls if needed :

```
sudo service xrdp start | stop | restart | status
```

From Windows start the “Remote Desktop Connection” tool from the start menu and fill the IP address:



Click on “Connect” and you will come to the xrdp login screen.

Before installing WeeWX and the BC Robotics driver, setup the required Raspbian software by running the following commands (in this order) from the command prompt. This is made easy if you start-up a VNC connection and then just copy and past these commands.

***Note:** WeeWX now runs under Python V3, therefore you **must** follow these instructions.*

Installing Required Libraries

Install Python V3.x

Ensure that you have the latest Python V3.x installed (WeeWX now runs on Python V3).

```
sudo apt update
sudo apt-get install python3      # Just a check
```

Install the Adafruit Python GPIO Library

```
sudo apt-get update
sudo pip3 install --upgrade setuptools
pip3 install RPI.GPIO
pip3 install adafruit-blinka
```

Install the Adafruit BME280 Library

```
sudo pip3 install adafruit-circuitpython-bme280
```

Install the ADS1x15 Library, the ADS1015 Analog to Digital chip (wind direction)

```
sudo pip3 install adafruit-circuitpython-ads1x15
```

Setup the DS18B20 temperature sensor and install the software library

```
cd
```

```
sudo modprobe wl-gpio
sudo modprobe wl-therm
cd /sys/bus/wl/devices
ls
```

The “ls” command will display the contents of the devices “folder” in the window. The DS18B20 (*if connected!*) shows up as an address something like “28-0316853d8fff” — but each sensor has a unique ID.

```
cd *sensor ID*
cat wl_slave
```

The temperature is located in the second line, take the number given and divide by 1000 to get the temperature in degrees Celsius. Now install the software library:

```
cd
sudo pip3 install wlthermsensor
```

You can test all the sensors by running the included test app: BCRobotics-test-app.py

Run the Python IDE:

```
>> Thonny
```

Now open the “BCRobotics-test-app.py” test app from the IDE and hit 'F5' to run it. It will continuously print out the readings from the sensors, including the value read from the ADC for the wind direction. Use these values (and a test template for the wind direction) to double check the values used in the driver. Hit “<ctrl> c” to stop the program.

Note: This test app is available on Github, and it will run without the external temperature and wind sensors installed. See:

<https://github.com/David-Enst/WeeWX-BCRobotics>

To check the versions of all software libraries enter:

```
>> pip3 freeze
```

WeeWX Setup

The Raspberry Pi runs WeeWX, a freely available weather application written in Python. The installation instructions for WeeWX, repeated below, are here:

<http://weewx.com/docs/debian.htm>

First, tell apt to trust this source. Do this only once—the first time you install WeeWX.

```
wget -qO - https://weewx.com/keys.html | sudo gpg --dearmor --output
/etc/apt/trusted.gpg.d/weewx.gpg
```

For *Debian10 and later* (uses python3) tell apt where to find the WeeWX releases:

```
wget -qO - https://weewx.com/apt/weewx-python3.list | sudo tee /etc/apt/sources.list.d/weewx.list
```

Use `apt-get` to install WeeWX. The installer will prompt for a location, latitude/longitude, altitude, station type, and parameters specific to your station hardware.

```
sudo apt-get update      # You must do this!!
sudo apt-get install weewx
```

Enter these parameters, when prompted for by WeeWX, upon start-up:

- **Location** – short description (e.g., BCRobotics in Ottawa, Canada)
- **Latitude & Longitude** – 45.48044, -75.89585
- **Altitude** – 87, meter
- **Units** – metric
- **Station Type** = Simulator (start with this definition)

When you are done, WeeWX will be running in the background as a daemon and simulated readings.

To make sure things are running properly look in the system log for messages from WeeWX.

```
sudo tail -f /var/log/syslog
```

After about 5 minutes, open the station web page in a web browser. You should see your station information and data:

```
file:///var/www/html/weewx/index.html
```

The installation of WeeWX results in the following layout:

Role	Symbolic Name	Location
WeeWX Root Directory	WEEWX_ROOT	/
Executables	BIN_ROOT	/usr/bin/weewxd
Configuration directory	CONFIG_ROOT	/etc/weewx/weewx.conf
Skins and templates	SKIN_ROOT	/etc/weewx/skins
SQLite databases	SQLITE_ROOT	/var/lib/weewx/
MySQL databases		/var/lib/mysql/weewx/
Web pages and images	HTML_ROOT	/var/www/html/weewx/
Documentation	DOC_ROOT	/usr/share/doc/weewx/
Examples	EXAMPLE_ROOT	/usr/share/doc/weewx/examples/
Utilities		/usr/share/wee_*
MySQL database setup files		/etc/mysql/*
Log files		/var/log/syslog
PID File		/var/run/weewx.pid
Extensions (i.e., the driver)		/usr/share/weewx/user/

WeeWX logs many events to the system log. On Debian systems, this is in:

```
/var/log/syslog
```


Updating the WeeWX Software

We have setup the WeeWX apt repository above as `/etc/apt/sources.list.d/weewx.list`, therefore, to get the latest version and install it do this (or just install updates as prompted by Raspbian):

```
sudo apt-get update
```

Configuring MySQL (Maria Database)

This section sets up the mandatory MySQL database for MyWX Pro, instead of the default SQLite database. Using the MySQL database is required for multi-user applications (i.e., sharing the data with another client over your local LAN or the Internet).

First, run these commands to upgrade the current packages to the latest version.

```
sudo apt update
sudo apt upgrade
```

Verify that the MySQLdb python package is installed:

```
python -c "import MySQLdb"
```

If this results in an import error

```
ModuleNotFoundError: No module named 'MySQLdb'
```

Then install the MySQLdb package. For example, on Debian systems:

```
sudo apt-get install mariadb-client // installs the Maria SQL database
sudo apt-get install mariadb-server // installs the Maria SQL server
sudo apt-get install python3-mysqldb // installs the Python SQL software
```

Configuring WeeWX

Next, you *must* change the WeeWX configuration to use MySQL instead of SQLite. In the WeeWX configuration file `weewx.conf`, change the `[[wx_binding]]` section to point to the MySQL database, `archive_mysql`, instead of the SQLite database `archive_sqlite`.

After the change, it will look something like this (change highlighted):

```
[[wx_binding]]
# The database should match one of the sections in [Databases]
database = archive_mysql
# The name of the table within the database
table_name = archive
# The class to manage the database
manager = weewx.wxmanager.WXDaySummaryManager
# The schema defines to structure of the database contents
schema = schemas.wview.schema
```

Assuming that you want to use the default database configuration, the `[[MySQL]]` section should look something like this:

```
[[MySQL]]
driver = weedb.mysql
host = localhost
user = weewx
password = "weewx"
```

This assumes user WeeWX has the password “weewx”. Adjust as necessary.

You will need to give the necessary permissions to the `weewx` database to whatever MySQL user you choose, by default the WeeWX software uses user `weewx` with the password `weewx`. Defining the default

weewx user with appropriate permissions, and defining a user with necessary permissions for remote access, for example:

```

sudo mysql -u root                                # Start the interactive access to the database
Enter password:                                   # no password by default initially! Just hit <enter>
MariaDB [(none)]> connect weewx                  # select the weewx database (optional)
MariaDB [weewx]> ALTER USER 'root'@'localhost' IDENTIFIED BY 'pwd';           # Change the root password!!
MariaDB [weewx]> GRANT ALL ON weewx.* TO 'weewx'@'localhost' IDENTIFIED BY 'pwd'; # Basic login
MariaDB [weewx]> GRANT ALL ON weewx.* TO 'weewx'@'%' IDENTIFIED BY 'pwd';      # WAN access acct
MariaDB [weewx]> GRANT ALL ON weewx.* TO 'weewx'@'10.0.0.%' IDENTIFIED BY 'pwd'; # LAN access
MariaDB [weewx]> GRANT ALL ON weewx.* TO 'wasweewx'@'%' IDENTIFIED BY 'pwd' REQUIRE X509; # WAN Two way SSL
MariaDB [weewx]>
MariaDB [weewx]> SELECT User, Host FROM mysql.user WHERE Host <> 'localhost'; # Verify accesses
MariaDB [weewx]> ALTER USER 'weewx'@'localhost' IDENTIFIED BY 'newPwd';      # FYI, if needed
MariaDB [weewx]> FLUSH PRIVILEGES;                                           # ensures the accesses are updated
MariaDB [weewx]> SHOW VARIABLES LIKE '%ssl%';                               # Check SSL access
+-----+-----+
| Variable_name | Value |
+-----+-----+
| have_openssl  | YES   |
| have_ssl      | YES   |
| ssl_ca        | /etc/mysql/certs/ca-chain.pem |
| ssl_capath    |       |
| ssl_cert      | /etc/mysql/certs/server-cert.pem |
| ssl_cipher    |       |
| ssl_crl       |       |
| ssl_crlpath   |       |
| ssl_key       | /etc/mysql/certs/server-key.pem |
| version_ssl_library | OpenSSL 1.1.1n 15 Mar 2022 |
+-----+-----+
MariaDB [weewx]> exit;

#
# IMPORTANT: (to avoid issues with your ISP) now run this script to secure the MARIADB
#
pi@MyPi: mysql_secure_installation          # Change Root password and eliminate the test DB etc.
pi@MyPi:
pi@MyPi: sudo mysql -u root -p              # the way to log into the MARIADB now (password prompt)

```

See: <https://www.tecmint.com/mysql-mariadb-security-best-practices-for-linux/>

The above commands create the weewx user accounts with password pwd for use by the WeeWX app and for LAN access to the Maria DB, and the wasweewx user with password pwd for WAN access from the MyWX Pro app. The weewx user account may connect from all possible addresses on the local LAN: 10.0.0.* (adjust as required for your LAN). The wasweewx user account may connect from all possible addresses on the WAN, but only if the required X509 certificates are verified.

The MySQL database files will be in the following location:

```
/var/lib/mysql/weewx
```

If you need privileged access to these folders, start the file explorer from the command line as follows:

```
sudo pcmanfm
```

If needed, you can start MySQL as follows:

```
sudo service mysqld start
```

Raspbian Command Summary

To edit a system file in Raspbian: `sudo Mousepad filename`. This command may be executed from the File Manager: *Tools / Run a Command in Current Folder ...*)

Note: *sudo* – provides privileged access to files and commands, and
Mousepad – starts up a simple GUI text editor.

To make a folder or file writable (via FTP): `sudo chmod a+rwX filename` — the following might be useful to define after installing WeeWX:

```
sudo chmod a+rwX /usr/share/weewx/user/ (for copying BCRobotics.py driver)
sudo pcmanfm (start File Manager with privileges)

sudo chmod a+rwX /etc/weewx/skins/Seasons/skin.conf (for updating the web page)
sudo chmod a+rwX /var/www/html/weewx/
sudo chmod a+rwX /etc/weewx/weewx.conf (to copy / edit configuration file)
sudo chmod a+rwX /etc/mysql/mariadb.conf.d/50-server.cnf (config mariadb)
sudo chmod a+rwX /etc/mysql/mariadb.conf.d/50-client.cnf (config mariadb)
sudo chmod a+rwX /etc/mysql/certs/ (for copying certs to the Pi)
sudo chmod ugo+r /etc/mysql/certs/* (you must change protections back to
read-only or MySQL will not use them!)
```

To shutdown the Pi:

```
sudo shutdown -h now
```

To start/stop WeeWX:

```
sudo /etc/init.d/weewx start
sudo /etc/init.d/weewx stop
sudo /etc/init.d/weewx restart
```

BCRobotics Driver Installation

Edit the [Station] section in `etc/weewx/weewx.conf` to add the following parameters, adjusted for your particular situation:

- `station_type = BCRobotics`
- `station_url = https://www.wunderground.com/dashboard?ID=IOTTAWA98`

The value in bold is mandatory definitions for the BCRobotics driver.

Once done, download the BCRobotics driver file to your Raspberry Pi from here (select the version3 branch):

```
https://github.com/David-Enst/WeeWX-BCRobotics/blob/master/bin/user/BCRobotics.py
```

Copy the `BCRobotics.py` file to `\usr\share\weewx\user`

Now manually define and update the parameters in `weewx.conf` as described below.

Definition of the BCRobotics driver by replacing the [Simulator] section with the following:

```
[BCRobotics]
# This defines the "Spark Fun" SEN-08942 / BC Robotics weather stations.
# See: https://www.sparkfun.com/products/8942
# https://www.bc-robotics.com/tutorials/raspberry-pi-weather-station-part-1/
```

```

# The time (in seconds) between LOOP packets.
loop_interval = 3

# Driver mode - tcp, udp, or serial
mode = serial

# If serial, specify the serial port device. (ex. /dev/ttyS0, /dev/ttyUSB0,
# or /dev/cuaU0)
# If TCP, specify the IP address and port number. (ex. 192.168.36.25:3000)
port = /dev/ttyS0

# The amount of time, in seconds, before the connection fails if
# there is no response
timeout = 3

# Debug level - the level of message logging. The higher
# the number, the more info is logged.
debug_read = 0

# The driver to use:
driver = user.BCRobotics

```

Update the Weather Underground (or other feed) definition, if desired:

```

[[Wunderground]]
# This section is for configuring posts to the Weather Underground.
# If you wish to do this, set the option 'enable' to true,
# and specify a station (e.g., 'KORHOODR3') and password.
# To guard against parsing errors, put the password in quotes.
enable = true
station = IOTTAWA98
password = 'abcdebfg'

```

Verify the definition of [[[Units]]] to ensure the web pages generated display the correct units:

```

[[StandardReport]]
# The StandardReport uses the 'Standard' skin, which contains the
# images, templates and plots for the report.
skin = Standard

[[[Units]]]
[[[Groups]]]
group_altitude      = meter
group_pressure       = hPa
group_rain           = mm
group_rainrate       = mm_per_hour
group_temperature    = degree_C
group_degree_day     = degree_C_day
group_speed          = km_per_hour
group_speed2         = km_per_hour2

```

Verify the definition of [[[Labels]]] to ensure the web pages generated display the correct labels for your data:

```

[[[Labels]]]
# Generic labels, keyed by an observation type.
[[[Generic]]]
barometer           = Barometer
dewpoint            = Dew Point
ET                  = ET
heatindex           = Heat Index
inHumidity          = Case Humidity

```

```

inTemp      = Case Temperature
outHumidity  = Humidity
outTemp     = Outside Temperature
radiation   = Radiation
rain        = Rain
rainRate    = Rain Rate
UV          = UV Index
windDir     = Wind Direction
windGust    = Gust Speed
windGustDir = Gust Direction
windSpeed   = Wind Speed
windchill   = Wind Chill
windgustvec = Gust Vector
windvec     = Wind Vector
extraTemp1  = Temperature1
extraTemp2  = Temperature2
extraTemp3  = Temperature3

```

Update the [StdConvert] section is to reflect the fact that all data in the database is stored in METRIC by default:

```

target_unit = METRIC    # Options are 'US', 'METRICWX', or 'METRIC'

```

The database and driver is setup to store values in METRIC by default. Therefore, the following updates to the quality control parameters are required:

[StdQC]

```

[[MinMax]]
barometer = 800, 1110, mbar # = hPa
pressure  = 800, 1110, mbar # = hPa
outTemp   = -50, 49, degree_C
inTemp    = -25, 49, degree_C
outHumidity = 0, 100
inHumidity = 0, 100
windSpeed = 0, 300, km_per_hour
rain      = 0, 250, mm

```

Finally, update the archive section to force WeeWX to generate archive records through software generation, since the BCRobotics hardware does not generate them:

[StdArchive]

```

# If possible, new archive records are downloaded from the station
# hardware. If the hardware does not support this, then new archive
# records will be generated in software.
# Set the following to "software" to force software record generation.
record_generation = software

```

And also specify that the MySQL database should be used along with the username and password to use:

[DataBindings]

```

[[wx_binding]]
# The database must match one of the sections in [Databases].
# This is likely to be the only option you would want to change.
database = archive_mysql

```

[DatabaseTypes]

```

# Defaults for SQLite databases
[[SQLite]]
driver = weedb.sqlite
# Directory in which the database files are located

```

```

SQLITE_ROOT = /var/lib/weewx

# Defaults for MySQL databases
[[MySQL]]
driver = weedb.mysql
# The host where the database is located
host = localhost
# The user name for logging in to the host
user = weewx
# The password for the user name (quotes guard against parsing errors)
password = "12345678"

```

These parameters should all be verified — now restart WeeWX (sudo /etc/init.d/weewx start) and check to see if it is running correctly. You can do this by opening the Pi web browser and going to the following page:

```
file:///var/www/html/weewx/index.html
```

Or by looking at the end of the syslog file for WeeWX entries:

```
sudo tail -f /var/log/syslog
```

Station data

The following table shows the data provided by the station hardware and those calculated by WeeWX.

BCRobotics Station Data			
Database Field	Observation	Loop	Archive
dateTime	-	D	
usUnits	-	D	
interval	-	D	
barometer		S	
pressure	pressure	H	
altimeter		S	
inTemp (Case)	case_temp	H	
outTemp	outTemp	H	
outHumidity	outHumidity	H	
windSpeed	windSpeed	H	
windDir	windDir	H	
rain	rain	D	
rainRate	rainRate	D	
rainTotal	-	S	
dewpoint	-	S	
windchill	-	S	
heatindex	-	S	

H indicates data provided by **H**ardware

D indicates data calculated by the **D**river

S indicates data calculated by the StdWXCalculate **S**ervice

Some references:

<https://www.bc-robotics.com/tutorials/raspberry-pi-weather-station-part-2/>
<http://www.weewx.com/docs/usersguide.htm#about>
<https://docs.python.org/2.7/contents.html>
<https://www.lenntech.com/calculators/humidity/relative-humidity.htm>
<https://sourceforge.net/p/raspberry-gpio-python/wiki/Inputs/>
<https://github.com/David-Enst/WeeWX-BCRobotics>
<https://mariadb.com/resources/>
<https://mariadb.com/kb/en/about-mariadb-connector-j/>
<https://javarevisited.blogspot.com/2016/09/javasqlsqlexception-no-suitable-driver-mysql-jdbc-localhost.html>
<https://mariadb.com/kb/en/mariadb-vs-mysql-compatibility/>

Enable Remote Access to WeeWX Device

Follow these procedures to allow remote access to the MySQL server on your Pi. Note that by default the Maria Data Base only allows local access (i.e., to localhost 127.0.0.1).

1. In the `/etc/mysql/mariadb.conf.d/50-server.cnf` file change the line as follows to allow remote TCP/IP connections (also make sure that it is not commented out (i.e., #):

```
bind-address = *    # or 0.0.0.0
```

2. As illustrated above, create a new MySQL user with required privileges:

Start the MariaDB app, and enter these commands (the user account “weewx” is created by the GRANT ALL command; note that the “%” character is a wildcard):

```
sudo mysql -u root          # Start the interactive access to the database
Enter password:             # no password by default initially
MariaDB [(none)]> connect weewx # select the weewx database (optional)
MariaDB [weewx]> GRANT ALL ON weewx.* TO 'secweewx'@'%' IDENTIFIED BY 'pwd' REQUIRE X509; # 2-way SSL
MariaDB [weewx]> exit;
```

3. To change a password, if needed:

```
> SET PASSWORD FOR 'weewx'@'%'10.0.0.%' = PASSWORD('newpass');    or
> ALTER USER 'weewx'@'localhost' IDENTIFIED BY 'NEWPASSWORD';
```

Configuring PKI Security

The MyWX Pro android app supports secure remote access to your WeeWX device. In order to provide this secure remote (from the Internet) access several definitions and configurations need to doing.

The process of setting up the required Certificate Authority (CA), server certificates, client certificates, and public keys is rather involved; it is therefore described in a separate document. See the PDF file:

“Weather Station Raspberry Pi security setup for MyWX Pro.pdf”

A summary of what is needed as a preview:

- Define a specific username that requires “X.509”
e.g. GRANT ALL ON weewx.* TO 'secweewx'@'%' IDENTIFIED BY 'pwd' REQUIRE X509;
- Define PKI certificates and keys for the Maria DB server
- Enable SSL/TSL in the Maria DB and pointing to the defined certs and keys
- Copy the certs and keys to your MyWX client device

Backup WeeWX

Backup the WeeWX files to your PC or other location, if desired / required:

1. Backup the setup files
 /etc/dhcpd.conf
 /etc/weewx/weewx.conf
 /etc/weewx/skins/seasons/index.html.tpl (if modified, plus skin.conf if needed)
2. Backup the driver
 /usr/share/weewx/user/BCRobotics.py
3. Backup the database (MySQL must be stopped and permissions on the file set)
 /var/lib/mysql/weewx (MySQL files)

For step three, backing up the entire database files, you must do the following steps:

1. Stop WeeWX – sudo /etc/init.d/weewx stop
2. Stop MySQL (so the files are not locked) – sudo /etc/init.d/mysql stop
3. Copy the files – copy the files to your backup location (e.g., use FileZilla to a Windows system)
4. Start MySQL – sudo /etc/init.d/mysql start
5. Start WeeWX – sudo /etc/init.d/weewx start

Weather Formulæ

Wind Direction

The wind direction sensor outputs a different number read by an ADC (analog-to-digital converter) based upon the direction—but this number will change a small amount based on temperature, power fluctuations, and the actual resistors in the sensor. so you can't just check if the ADC output matches a number in a list. Instead, they fall within a range as shown in the table.

Actual readings from my sensor, used to fine-tune the Python code in the driver, are shown below. Note that in the `BCRobotics.py` driver the following selections are used to try to cover *all* possible values. This eliminates an out-of-range error, and therefore a faulty wind direction (or program crash!). It is quite likely that the driver will work properly with the range selection values shown, but it does not hurt to do a little test to see the actual values from your device. The numbers are found by manually moving the wind direction sensor to each of the 16 directions and reading the output. Write these numbers down. See the compass rose template provided at the end of this document for your convenience.

Direction (Degrees)	Resistance (Ohms)	Voltage (V=5v, R=10k)	ADC Reading	Range Selection
N (0)	33k	3.84v	20352	19600-20999
NNE (22.5)	6.57k	1.98v	10512	9000-10799
NE (45)	8.2k	2.25v	11952	10800-13999
ENE (67.5)	891	0.41v	2176	2000-2299
E (90)	1k	0.45v	2416	2300-2999
ESE (112.5)	688	0.32v	1712	1000-1999
SE (135)	2.2k	0.90v	4768	4000-4999
SSE (157.5)	1.41k	0.62v	3264	3000-3999
S (180)	3.9k	1.40v	7440	6600-8999
SSW (202.5)	3.14k	1.19v	6336	5000-6599
SW (225)	16k	3.08v	16336	15900-16999
WSW (247.5)	14.12k	2.93v	15536	14000-15899
W (270)	120k	4.62v	24464	24000-24999
WNW (292.5)	42.12k	4.04v	21440	21000-21999
NW (315)	64.9k	4.78v	22992	22000-23999
NNW (337.5)	21.88k	3.43v	18208	17000-19599

Wind Speed

Wind speed is the rate of horizontally moving air past a fixed point, and measured by an anemometer. Some definitions for the weather app (on the Raspberry Pi):

Wind Speed – Current wind speed is the average wind speed recorded over a defined period, 3 seconds for example (i.e., the “LOOP” period).

Average Speed – An average of all wind speed-readings (i.e., LOOP readings) collected over a defined period, 5 minutes for example (i.e., the ARCHIVE period).

Peak Speed – A rolling value showing the highest wind speed recorded over a given interval (usually the ARCHIVE interval).

Wind Chill and Feels-Like Temperature

Wind Chill

Wind chill temperature only works for temperatures at or below 10°C (50°F) and wind speeds above 4.8kph (3.0 mph). The standard Wind Chill formula from Environment Canada (using °C and km/h; but the same formula converted to °F and MPH is used in the USA) is:

$$T_{wc} = 13.12 + 0.6215T_a - 11.37v^{+0.16} + 0.3965T_av^{+0.16}$$

Where:

T_{wc} is the wind chill index, based on the Celsius temperature scale;

T_a is the air temperature in degrees Celsius; and

v is the wind speed at 10m (33 ft.) standard anemometer height, in kilometres per hour.

When the temperature is -20 °C (-4 °F) and the wind speed is 5 km/h (3.1 mph), the wind chill index is -24. If the temperature remains at -20 °C and the wind speed increases to 30 km/h (19 mph), the wind chill index falls to -33.

Feels-Like Temperature

A heat index value is valid for temperatures at or over 27° Celsius.

There are many formulae devised to approximate the original tables by Steadman. Anderson et al (2013), NWS (2011), Jonson and Long (2004), and Schoen (2005) have lesser residuals in this order. The former two are a set of polynomials, but the third one is by a single formula with exponential functions.

The formula below approximates the heat index in degrees Fahrenheit, to within ±1.3 °F (0.7 °C). It is the result of a multivariate fit to a model of the human body. The temperature must be equal to or greater than 80 °F (27 °C) and relative humidity equal to or greater than 40%. This equation reproduces the above NOAA National Weather Service table (except the values at 90 °F (32 °C) and 45%-70% relative humidity vary unrounded by less than ±1, respectively). (*WeeWX uses this one*)

$$HI = c_1 + c_2T + c_3R + c_4TR + c_5T^2 + c_6R^2 + c_7T^2R + c_8TR^2 + c_9T^2R^2$$

Where:

HI = heat index (in degrees Fahrenheit)

T = ambient dry-bulb temperature (in degrees Fahrenheit)

R = relative humidity (percentage value between 0 and 100)

$$c_1 = -42.379$$

$$c_2 = 2.04901523$$

$$c_3 = 10.14333127$$

$$c_4 = -0.22475541$$

$$c_5 = -6.83783 \times 10^{-3}$$

$$c_6 = -5.481717 \times 10^{-2}$$

$$c_7 = 1.22874 \times 10^{-3}$$

$$c_8 = 8.5282 \times 10^{-4}$$

$$c_9 = -1.99 \times 10^{-6}$$

Temperature

The DS18B20 temperature sensor shows up on the Raspberry Pi with an address something like 28-0517c13642ff as noted in the setup instructions above. The sensor provides a value in centigrade.

The simple formula to convert between Fahrenheit (°F) and degrees Celsius (°C):

$$T_c = \frac{9}{5} \times (T_f - 32)$$

$$T_f = (\frac{5}{9}) \times T_c + 32$$

Where:

T_c is temperature in Celsius

T_f is temperature in Fahrenheit

WeeWX looks after all these conversions ... see the documentation on the Standard web report.

Humidity

The humidity is measured inside the case containing the Pi, therefore this relative humidity (RH) reading is relative to the temperature inside the case (i.e., `case_temp`) and must be converted to a RH for the temperature outside (i.e., the `outTemp`). The following formulae are used:

See: <https://www.lenntech.com/calculators/humidity/relative-humidity.htm>

```
# This humidity is measured inside the case, which is warmer than the
# ambient air. Therefore it is converted to external humidity based
# upon the case_temp, in_humidity, pressure, and out_Temp.
#
# Use NOAA formulae:
VapPress = (6.112*math.exp(17.67*case_temp / (case_temp + 243.5)))*(in_humidity/100)
DewPoint = (243.5*math.log(VapPress / 6.112))/(17.67 - math.log(VapPress / 6.112))
absVapPress = 6.11*math.pow(10, (7.5*DewPoint / (237.7 + DewPoint)))
actMixRatio = 621.97*absVapPress / (pressure - absVapPress)

# Adjust humidity reading to the outside temperature
# using NOAA values:
out_humidity = actMixRatio * 10 / (0.42 * math.exp(out_Temp * 0.06235394))
    if out_humidity > 100:
        out_humidity = 100.0
```

Pressure

The current pressure displays in hectopascals (hPa). A trend of the changing pressure shows what has happened over the last 3 hours.

Rainfall

The rainfall amount accumulated over the LOOP period is tracked. Weather Underground displays:

- **Rain** – the accumulation of rain in the past 1 hr and the past 24 hrs
- **Rate** – the current rate of rainfall in mm/hr for the LOOP period

There are certain situations where a single random rain “tick” may occur:

- Certain gusts of strong wind could cause a random tick—if the “bucket” is not exactly level and when at rest on the high side a strong gust could knock it back;
- Melting frost or dew may accumulate enough to cause a random tick over time;
- An almost full “bucket” and a sudden gust of wind can also cause it to tick over.

The WeeWX driver therefore has a random tick detector to catch single ticks. The downside is that the first tick when it actually starts to rain will be ignored.

Wunderground

The WeeWX software can share its reading with your own Weather Underground web page. Register your Personal Weather Station (PWS) at wunderground.com, where you will receive a station ID along with a password. Enter these into the `weewx.conf` file as directed here:

[http://weewx.com/docs/usersguide.htm#\[StdRESTful\]](http://weewx.com/docs/usersguide.htm#[StdRESTful])

Wunderground registration detail example:

Your Station ID: **ABCDEFGGxx**

Your Station Key/Password: **abxycdij**

Example WU PWS URL:

<https://www.wunderground.com/personal-weather-station/dashboard?ID=IOTTAWA98>

Once your PWS is setup you can use this WU URL in your [Station] section's `station_url` setting.

Raspberry Pi Documentation

Follow these links:

<https://github.com/David-Enst/WeeWX-BCRobotics/tree/version3> # The BCRobotics Driver

<https://www.raspberrypi.org/documentation/configuration/>

<https://www.raspberrypi.org/documentation/remote-access/>

<https://domoticproject.com/accessing-raspberry-ddns/> (Accessing from the Internet)

Each segment is a $22\frac{1}{2}^\circ$ arc; therefore, NNE ($22\frac{1}{2}^\circ$) would encompass readings from $11\frac{1}{4}^\circ$ to $33\frac{3}{4}^\circ$.

