

Wave Glider® SV3 Integrated Sensor User Guide

Airmar Weather Station

Revision B
March 2017
Part Number 08336

**LIQUID
ROBOTICS**
A Boeing Company

Wave Glider® SV3 Integrated Sensor User Guide: Airmar Weather Station

Copyright © 2017 Liquid Robotics, Inc.

Confidential computer software. Valid license from Liquid Robotics, Inc. required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Trademarks

Liquid Robotics®, Wave Glider®, Wave Glider Management System, WGMS™, and SHARC® are worldwide trademarks of Liquid Robotics, Incorporated. Other products and company names mentioned herein may be the trademarks of their respective owners.

Patents

The Wave Glider mechanism is patented under US Patent Number 7,371,136 and other patents pending. All aspects of the Wave Glider product have been developed at private expense by Liquid Robotics, Inc.

Warranty

The information contained herein is subject to change without notice. The only warranties for Liquid Robotics, Inc. products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as a warranty. Liquid Robotics, Inc. is not liable for technical or editorial errors or omissions contained herein.

Acknowledgements

The Airmar 200WX Weather Station that is integrated into the Wave Glider platform is manufactured by the AIRMAR Technology Corporation, 35 Meadowbrook Drive, Milford, NH 03055, USA.

Support from the National Oceanographic and Atmospheric Administration of the U.S. Department of Commerce (NOAA) facilitated development of the AirmarWeather plugin.

Revision History

Document Part Number	Revision	Publication Date	Changes
08336	A	December 2016	This is the initial release of this document. It is specific to the AirmarWeather plugin version 1.0.0.
08336	B	March 2017	This revision is specific to the AirmarWeather plugin version 1.1.0. It includes changes to the Airmar Weather Station default configuration to include Raw Pressure Measurements. For details, see the Load Defaults (Expert) command in “airmarWeatherConfig Service Commands” (page 8).

Table of Contents

Preface	vi
1. Overview	1
AirmarWeather Plugin	1
Airmar 200WX Weather Station	2
2. SV3 WGMS Airmar Weather Station User Interface	4
airmarWeatherConfig User Interface	5
airmarWeatherController User Interface	6
airmarWeatherReporter User Interface	6
airmarWeatherStation User Interface	7
3. Airmar Weather Station Commands	8
Airmar Weather Station Command Reference	8
airmarWeatherConfig Service Commands	8
Get Available NMEA Commands (Expert)	8
Get FW Version	9
Get HW Version	9
Get NMEA Definition (Expert)	9
Get Running Config	10
Get User Config (Expert)	10
Load Config (Expert)	10
Load Defaults (Expert)	10
Log	11
Produce JSON status report for this service (Expert)	11
Produce status report for this service	11
Toggle ship to shore config report (Expert)	12
Update Config (Expert)	12
airmarWeatherController Service Commands	13
Log	13
Produce JSON status report for this service (Expert)	13
Produce status report for this service	14
airmarWeatherReporter Commands	14
Get Weather Report Interval	14
Log	14
Produce JSON status report for this service (Expert)	15
Produce status report for this service	15
Send Shore P80 (Expert)	16
Send Shore P91 (Expert)	16
Set Weather Report Interval (Expert)	17
airmarWeatherStation Commands	17
Factory Defaults (Expert)	17
Log	18
Log 1Hz Weather Station Data	18
Pitch Threshold (Expert)	18
Produce JSON status report for this service (Expert)	18
Produce status report for this service	19
Roll Threshold (Expert)	20
Set Gust Averaging Window (Expert)	20
Use Filter on Data (Expert)	20
4. Pre-Launch	22
5. Troubleshooting	23
A. National Marine Electronics Association (NMEA) Definitions	24
B. Weather Record Types	25

C. Gust Definition 27

List of Figures

1.1. Airmar 200 WX Weather Station 2

2.1. WGMS SV3 Commands Window - Airmar Weather Station Services 4

2.2. Tool tip Example - Cursor Hovered Over "Get FW Version" Command Button 5

2.3. airmarWeatherConfig Service User Interface 5

2.4. airmarWeatherController Service User Interface 6

2.5. airmarWeatherReporter Service User Interface 7

2.6. airmarWeatherStation Service User Interface 7

B.1. Weather Records (PayloadType 0x80) 25

B.2. Weather Records 2 (PayloadType 0x91) 26

C.1. Gust Definition Table 27

Preface

The AirmarWeather plugin version 1.0.0 is obsolete, it is superseded by version 1.1.0. Liquid Robotics recommends upgrading to AirmarWeather plugin version 1.1.0. Contact Liquid Robotics Support for assistance. Liquid Robotics recommends installing the AirmarWeather plugin version 1.1.0 on all SV3 vehicles that have an Airmar 200WX station and are running Regulus 1.1.x. SV3 vehicles that do not have the AirmarWeather plugin installed will still operate without interruption, however their reported data will not be standardized to NOAA specifications and you will have fewer diagnostic tools at your disposal.

After October of 2016, the AirmarWeather plugin is factory installed on all Wave Glider SV3 vehicles that include an integrated Airmar 200WX weather station. To take advantage of the features included in the AirmarWeather plugin on Wave Glider SV3 vehicles purchased prior to October 2016 that include an integrated Airmar 200WX weather station, contact the Liquid Robotics Customer Support organization to arrange for installation of the AirmarWeather plugin.



Important

The AirmarWeather plugin version 1.1.0 is compatible with Regulus 1.1.x - it is not compatible with Regulus 1.0.3 or earlier and it is not compatible with Regulus 2.x.x.

This guide assumes a working familiarity with the Liquid Robotics Wave Glider SV3. The focus is primarily on information specific to the Airmar 200WX Weather Station and the Liquid Robotics, Inc. AirmarWeather plugin version 1.1.0.

If you have questions, or are having difficulty locating the technical documentation that you need, contact the Liquid Robotics Support Organization at:

- Telephone: (US) 888-574-4574 (International) 1-408-636-4205
- Email: support@liquidr.com

Before contacting Liquid Robotics support, please have available the serial number of the vehicle(s) or part(s) that you are contacting Liquid Robotics Customer Support about.

Chapter 1. Overview

The AirmarWeather plugin was developed to provide additional operator functionality to the Airmar 200WX weather station on the Wave Glider SV3 platform, and to perform correct statistical analysis on the weather station's data. The AirmarWeather plugin also provides a second weather record (Weather Record 2) that can be viewed in WGMS with more statistical analysis and measurement quality.

The algorithms used in the AirmarWeather plugin attempt to follow the standards laid out by NOAA <http://www.ndbc.noaa.gov/measdes.shtml>

AirmarWeather Plugin

The AirmarWeather plugin:

- Standardizes data handling and reporting based on consultation with National Oceanic and Atmospheric Administration
- Utilizes Regulus 1.1.x plugin to decouple Airmar weather station software development from Regulus allowing for quicker turnaround time for new releases of features
- Improves diagnostic capabilities reducing troubleshooting time
- Provides additional control over the weather station configurations and management
- Supports additional filtering capabilities based on vehicle dynamics
- Provides a new comprehensive weather sensor data type (Weather Records 2)

Airmar 200WX Weather Station

Figure 1.1. Airmar 200 WX Weather Station



Airmar 200WX Weather Station Specifications:

- Operating Temperature Range: -25°C to 55°C (-13°F to 131°F)
- Pitch and Roll Range/Accuracy: $\pm 50^\circ$ / $< 1^\circ$
- Air Temperature Range: -25°C to 55°C (-13°F to 131°F)
 - Air Temperature Resolution: 0.1°C (0.1°F)
 - Air Temperature Accuracy: $\pm 1^\circ\text{C}$ ($\pm 1.8^\circ\text{F}$)* @ > 4 knots (> 4.6 MPH) wind
- Barometric Pressure Range: 850 mbar to 1150 mbar (25 inHg to 34 inHg, 850 hPa to 1150 hPa)
 - Barometric Pressure Resolution: 0.1 mbar (0.029 inHg, 0.1 hPa)
 - Barometric Pressure Accuracy: ± 2 mbar (± 0.059 inHg, ± 2 hPa) when altitude correction is available
- Wind Speed Range: 0 knots to 80 knots (0 MPH to 92 MPH)
 - Wind Speed Resolution: 0.1 knots (0.1 MPH)

- Wind Speed Accuracy @ 0°C to 55°C (32°F to 131°F), no precipitation*:
 - Low Wind Speeds: 0 knots to 10 knots (0 MPH to 11.5 MPH); RMS error of 1 knot (1.1 MPH) +10% of reading
 - High Wind Speeds: 10 knots to 80 knots (11.5 MPH to 92 MPH); RMS error of 2 knots (2.3 MPH) or 5% RMS, whichever is greater
- Wind Speed Accuracy in wet conditions**: 5 knots (5.7 MPH) RMS
- Wind Direction Range: 0° to 360°
 - Wind Direction Resolution: 0.1°
 - Wind Direction Accuracy @ 0°C to 55°C (32°F to 131°F), no precipitation*:
 - Low Wind Speeds: 4 knots to 10 knots (4.6 MPH to 11.5 MPH)—5° RMS typical
 - High Wind Speeds: >10 knots (>11.5 MPH)—2° RMS typical
 - Wind Direction Accuracy in wet conditions**: >8 knots (9.2 MPH)—8° RMS typical

Additional information on the Airmar 200WX Weather Station is available at:

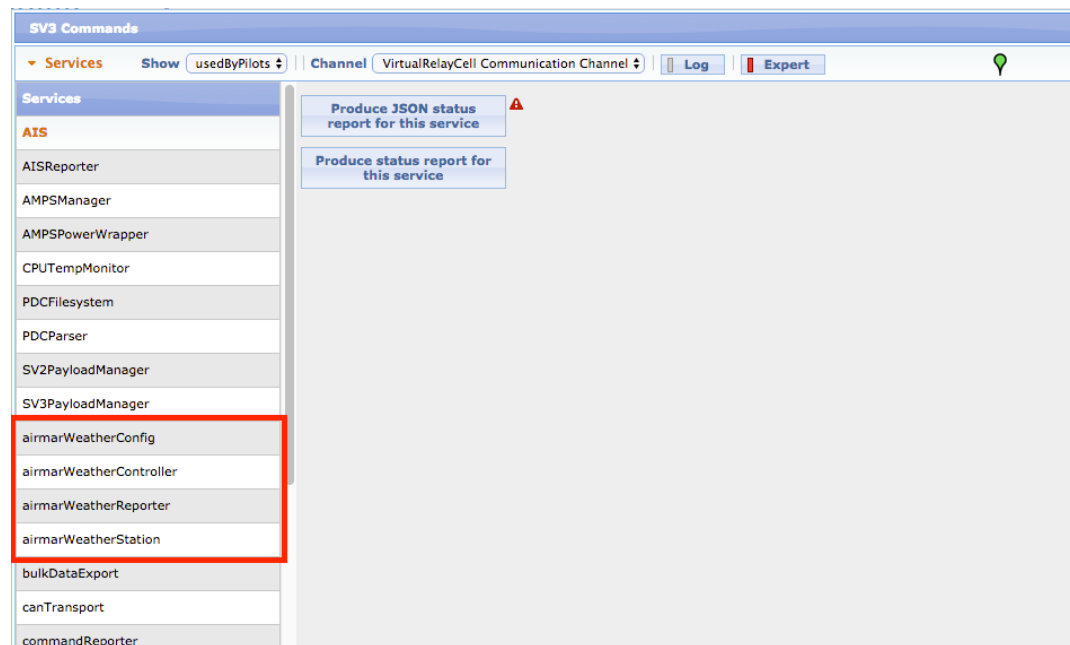
<http://www.airmartechology.com/productdescription.html?id=154>

Chapter 2. SV3 WGMS Airmar Weather Station User Interface

If you are using WGMS as a remote Wave Glider management application for an SV3 vehicle with an Airmar Weather Station, when the AirmarWeather plugin is installed and registered with Regulus, four weather related services will appear in the Services menu of the SV3 Commands Window:

- airmarWeatherConfig
- airmarWeatherController
- airmarWeatherReporter
- airmarWeatherStation

Figure 2.1. WGMS SV3 Commands Window - Airmar Weather Station Services

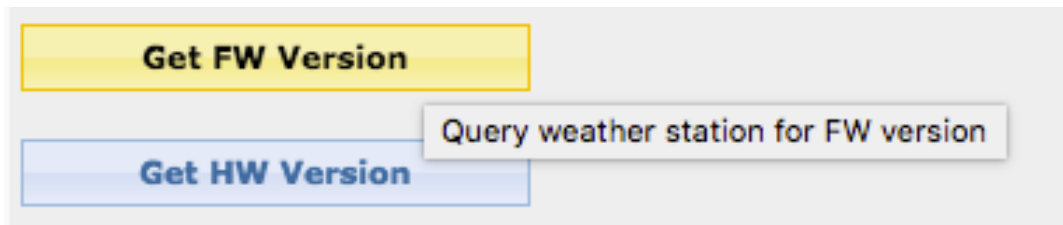


Note

Do not include quotation marks when entering HHmm values in SV3 Commands Window fields.

Tool tips appear when you hover your cursor over objects in the UI.

Figure 2.2. Tool tip Example - Cursor Hovered Over "Get FW Version" Command Button



For more information on using the WGMS SV3 Commands window see the *WGMS® User Guide*, Liquid Robotics Part Number 05910.

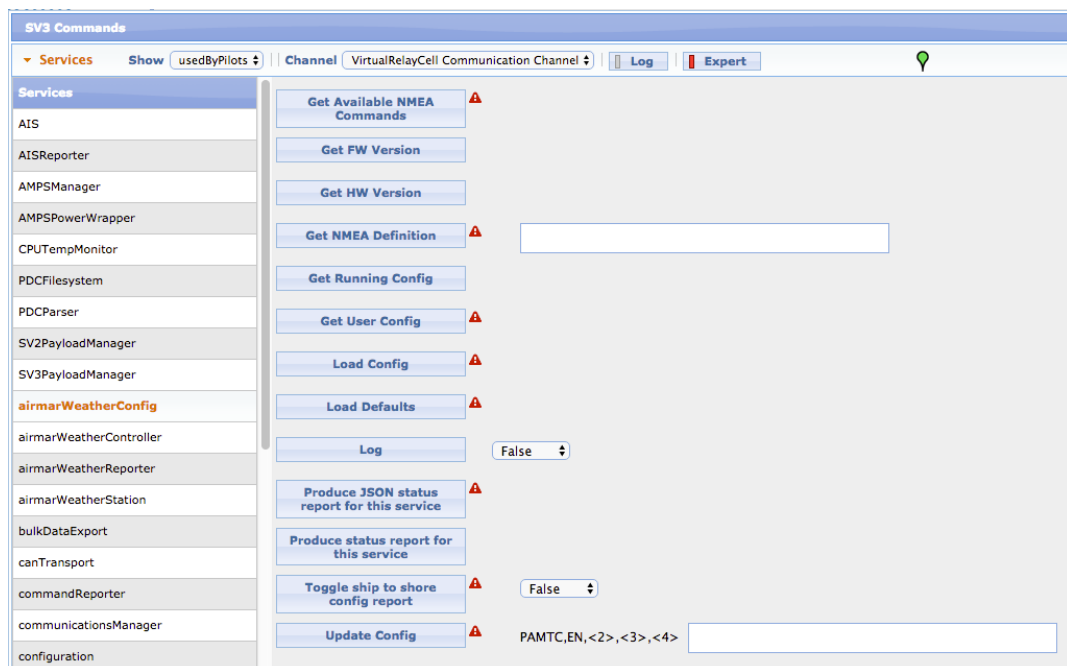
airmarWeatherConfig User Interface

The airmarWeatherConfig service UI includes:

- Airmar Weather Station configuration commands
- Commands for determining the HW and FW versions running on the Airmar weather station
- Commands for determining the status of the airmarWeatherConfig service
- A command that lists available NMEA commands.
- A command for turning the service log ON or OFF

There are 13 commands, of which 8 are expert level commands.

Figure 2.3. airmarWeatherConfig Service User Interface



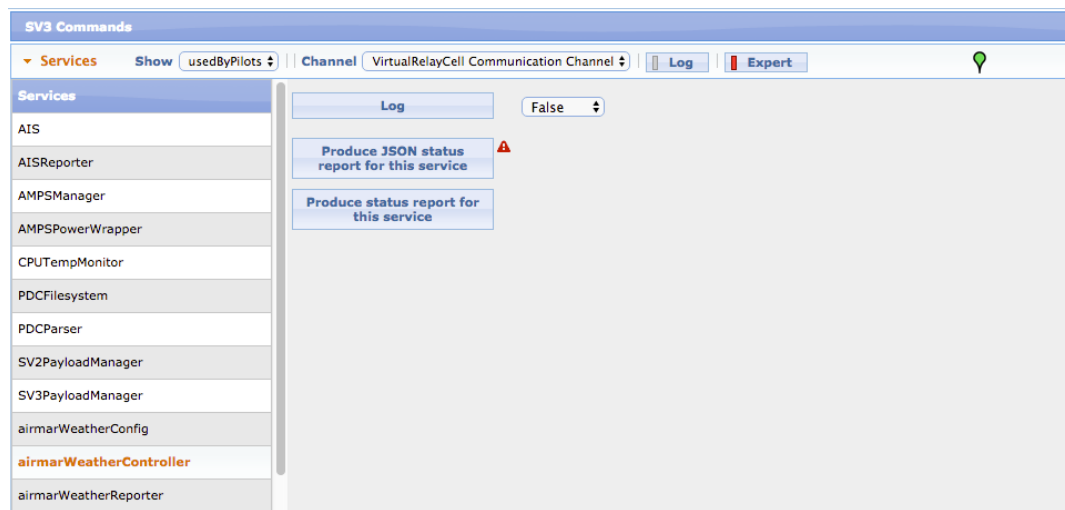
airmarWeatherController User Interface

The airmarWeatherController service UI includes:

- Commands for determining the status of the airmarWeatherController service.
- A command for turning the service log ON or OFF.

There are 3 commands, of which 1 is an expert level command.

Figure 2.4. airmarWeatherController Service User Interface



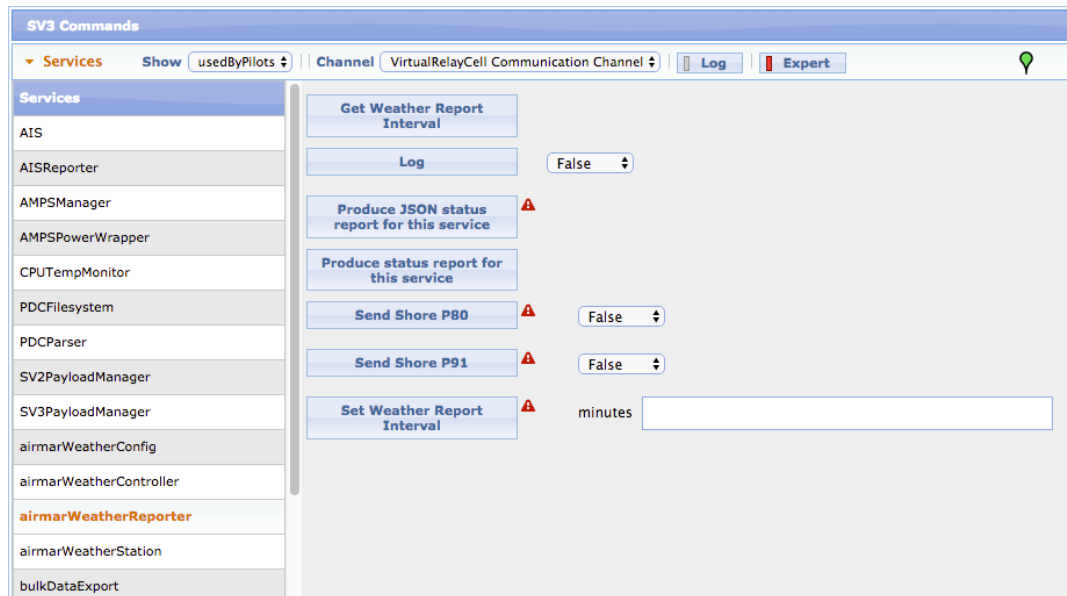
airmarWeatherReporter User Interface

The airmarWeatherReporter service UI includes:

- Commands for determining the status of the airmarWeatherReporter service
- Airmar Weather Station reporting configuration commands
- A command for turning the service log ON or OFF

There are 7 commands, of which 4 are expert level commands.

Figure 2.5. airmarWeatherReporter Service User Interface



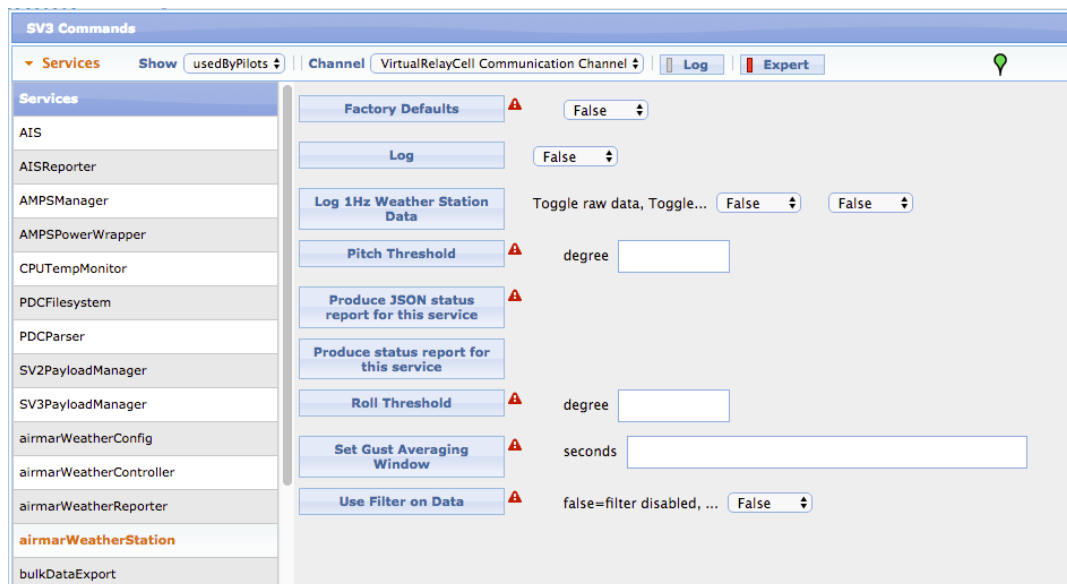
airmarWeatherStation User Interface

The airmarWeatherStation service UI includes:

- Commands for determining the status of the airmarWeatherStation service
- Commands for setting the Airmar Weather Station operating parameters
- A command for turning the service log ON or OFF

There are 9 commands, of which 6 are expert level commands.

Figure 2.6. airmarWeatherStation Service User Interface



Chapter 3. Airmar Weather Station Commands

This chapter is a reference for all of the Airmar Weather Station commands available in the SV3 Commands Window. For information on how these commands appear in the WGMS SV3 Commands Window see [Chapter 2: SV3 WGMS Airmar Weather Station User Interface](#).

The Airmar Weather Station commands will not appear in the remote application UI (for example the WGMS UI) unless the Airmar Weather Station is properly connected to the CCU and the AirmarWeather plugin has been registered with Regulus. If you receive an Unable to find AirmarWeather_1_1 console command NACK, the AirmarWeather plugin may not be properly registered with Regulus. If you believe the Airmar Weather Station is installed correctly and the AirmarWeather plugin is properly registered, try rebooting Regulus. If that doesn't work, contact Liquid Robotics customer support.

Airmar Weather Station Command Reference

This command reference is divided into 4 sections, one for each of the 4 Weather Station services:

- [“airmarWeatherConfig Service Commands” \(page 8\)](#)
- [“airmarWeatherController Service Commands” \(page 13\)](#)
- [“airmarWeatherReporter Commands” \(page 14\)](#)
- [“airmarWeatherStation Commands” \(page 17\)](#)

Each command is listed by its name as it appears in the WGMS SV3 Commands Window, followed by the command syntax, an example, and expected output.

airmarWeatherConfig Service Commands

AirmarWeatherConfig service is primarily used to configure the Airmar Weather Station, allowing the operator to fine tune specific settings. This service is also used to query the sensor for its hardware and software version.

Its Regulus class is:

```
com.liquidr.regulus.airmar.services.AirmarWeatherConfig
```

Get Available NMEA Commands (Expert)

Returns all of the available NMEA sentences that can be used to configure or update the weather station.

Console Command	<code>%airmarWeatherConfig.getCommandList()</code>
Example	<code>%airmarWeatherConfig.getCommandList()</code>
Expected Output	• Valid NMEA sentences, for example:

Airmar Weather
Station Commands

	DTM, GGA, GLL, GSA, GSV, HDG, HDT, MDA, MWD, MWVR, MWVT, RMC, ROT, VTG, VWR, VWT, XDRA, XDRB, ZDA, ALL See Appendix A: National Marine Electronics Association (NMEA) Definitions for more information on NMEA sentences.
--	--

Get FW Version

Returns the FW version running on the weather station.

Console Command	<code>%airmarWeatherConfig.reqFwVersion()</code>
Example	<code>%airmarWeatherConfig.reqFwVersion()</code>
Expected Output	Firmware version running on the attached weather station, for example: <code>{B1: 2.006, A1: 2.205, B2: 2.003, A2: 0.224, GPS: 13.5.31, COMP: 1.15}</code>

Get HW Version

Returns the HW version running on the weather station.

Console Command	<code>%airmarWeatherConfig.reqHwVersion()</code>
Example	<code>%airmarWeatherConfig.reqHwVersion()</code>
Expected Output	Hardware version of the attached weather station, for example: <code>{Part: 44-848-1-02, SN: 3359997, Model: 6}</code>

Get NMEA Definition (Expert)

Returns the NMEA definition of any valid command that the weather station supports. You can only query one command at a time.

Console Command	<code>%airmarWeatherConfig.getDefinition(<NMEA_DEFINITION>)</code> Where NMEA_DEFINITION can be any one of the following: DTM, GGA, GLL, GSA, GSV, HDG, HDT, MDA, MWD, MWVR, MWVT, RMC, ROT, VTG, VWR, VWT, XDRA, XDRB, ZDA, ALL
Example	<code>%airmarWeatherConfig.getDefinition(MWVR)</code>
Expected Output	NMEA definition of command, for example, in the case of MWVR: Wind Speed and Angle, in relation to the vessel's bow/centerline (relative)

Get Running Config

Returns the weather station configuration.

Console Command	<code>%airmarWeatherConfig.getRunConfig()</code>
Example	<code>%airmarWeatherConfig.getRunConfig()</code>
Expected Output	The configuration that is currently loaded and running on the weather station, for example: <code>[ZDA:1:10][GGA:1:10][HDG:1:10][XDRB:1:10][MDA:1:10]</code>

Get User Config (Expert)

Returns the set of NMEA sentences that will be programmed, if the Load Config command is executed. This is not necessarily the running configuration - it includes configuration changes that have not yet been implemented.

Console Command	<code>%airmarWeatherConfig.getUsrConfig()</code>
Example	<code>%airmarWeatherConfig.getUsrConfig()</code>
Expected Output	The configuration settings that are queued for programming the next time the Load Config command is issued.
Console Command	<code>%airmarWeatherConfig.setLog()</code> Parameters: true or false

Load Config (Expert)

Loads configuration changes made using the Update Config command onto the weather station.

Console Command	<code>%airmarWeatherConfig.configureStation()</code> Parameters: true or false
Example	<code>%airmarWeatherConfig.configureStation(true)</code>
Expected Output	Returns OK to confirm that the configuration has been loaded.

Load Defaults (Expert)

Loads the following Liquid Robotics factory defaults:

```

GPZDA      1 Hz // Time and Date
GPGGA      1 Hz // GPS Fix Data
HCHDG      1 Hz // Heading Relative
YXXDR(A)    1 Hz // Raw Pressure measurements
YXXDR(B)    1 Hz // Roll/Pitch measurements

```


WIMDA 1 Hz // Meteorological Composite. Barometric Pressure,
Air Temperature, Wind Direction, Wind Speed

Console Command	<code>%airmarWeatherConfig.loadDefaultConfig()</code>
Example	<code>%airmarWeatherConfig.loadDefaultConfig()</code>
Expected Output	Returns OK to confirm that the Liquid Robotics factory defaults have been loaded.

Log

Enables or disables logging for the airmarWeatherConfig service on the Regulus file system.

Example	<code>%airmarWeatherConfig.setLog(false)</code>
Expected Output	Confirms that the setting (enable or disable) has been applied, for example: "OK"

Produce JSON status report for this service (Expert)

Returns a JSON formatted status report for the airmarWeatherConfig service.

Console Command	<code>%airmarWeatherConfig.serviceReport()</code>
Example	<code>%airmarWeatherConfig.serviceReport()</code>
Expected Output	JSON formatted status, for example: <pre>{ "fw": "{B1: 2.006, A1: 2.205, B2: 2.003, A2: 0.224, GPS: 13.5.31, COMP: 1.15}", "hw": "{Part: 44-848-1-02, SN: 3359997, Model: 6}", "id": 100, "info": "HW={Part: 44-848-1-02, SN: 3359997, Model: 6}, FW={B1: 2.006, A1: 2.205, B2: 2.003, A2: 0.224, GPS: 13.5.31, COMP: 1.15}, CFG=[ZDA:1:10][GGA:1:10][HDG:1:10][XDRB:1:10] [MDA:1:10], POSTS=5, RPTS=6", "postCnt": 5, "reportCnt": 6, "runCfg": "[ZDA:1:10][GGA:1:10][HDG:1:10][XDRB:1:10] [MDA:1:10]", "state": "Happy", "tag": "airmarWeatherConfig" }</pre>

Produce status report for this service

Returns comprehensive information about the airmarWeatherConfig service and its status, including weather station information.

Console Command	<code>%airmarWeatherConfig.statusReport()</code>
-----------------	--

Example	%airmarWeatherConfig.statusReport()
Expected Output	Information about the airmarWeatherConfig service and the weather station, for example: <pre>fw: {B1: 2.006, A1: 2.205, B2: 2.003, A2: 0.224, GPS: 13.5.31, COMP: 1.15} hw: {Part: 44-848-1-02, SN: 3359997, Model: 6} id: 100 info: HW={Part: 44-848-1-02, SN: 3359997, Model: 6}, FW={B1: 2.006, A1: 2.205, B2: 2.003, A2: 0.224, GPS: 13.5.31, COMP: 1.15}, CFG=[ZDA:1:10] [GGA:1:10][HDG:1:10][XDRB:1:10][MDA:1:10], POSTS=5, RPTS=6 postCnt: 5 reportCnt: 6 runCfg: [ZDA:1:10][GGA:1:10][HDG:1:10][XDRB:1:10][MDA:1:10] state: Happy tag: airmarWeatherConfig</pre>

Toggle ship to shore config report (Expert)

This command is used to enable or disable automatic transmission of the weather station configuration report. When the airmarWeatherConfig service detects that the weather station has been powered on, or reset, it will reconfigure the workstation automatically to the last used stored configuration. Once configuration process has completed, if Toggle ship to shore config report is enabled, the weather station's information is transmitted back to shore and logged in WGMS.

Console Command	%airmarWeatherConfig.setConfReport() Parameters: true or false
Example	%airmarWeatherConfig.setConfReport(true)
Expected Output	Confirmation that the setting (enable or disable) has been applied, for example: OK

Update Config (Expert)

Use this command to add/delete/modify NMEA sentences that are published by the weather station. For a list of valid NMEA sentences, issue the Get Available NMEA Commands command. You may not be able to modify sentences that are deemed critical to the system, as specified in the default configuration. Use the Load Config command to upload configuration changes to the weather station.

Console Command	%airmarWeatherConfig.updateConfig(PAMTC,EN,<NMEA_CMD>,<ON/OFF>,<FREQUENCY>) Where: <pre><NMEA_CMD> can be any one of the following: DTM, GGA, GLL, GSA, GSV, HDG, HDT, MDA, MWD, MWVR, MWVT, RMC, ROT, VTG, VWR, VWT, XDRA, XDRB, ZDA, ALL</pre>
-----------------	--

	<ON/OFF> must be 1 or 0, where 1 = ON and 0 = OFF <FREQUENCY> must be a positive whole number representing 1/10ths of a second units, for example entering a 10 would represent a frequency of 1 Hz.
Example	%airmarWeatherConfig.updateConfig("PAMTC,EN,MWVR,1,10")
Expected Output	- If the sentence is valid, the new configuration will be returned, for example: [ZDA:1:10][GGA:1:10][HDG:1:10][XDRB:1:10][MWVR:1:10][MDA:1:10] - If a bad sentence is detected, an unable to process message will be returned with BAD in place of the unprocessed sentence, for example: Unable to process: PAMTC,EN,BAD,1,10

airmarWeatherController Service Commands

The airmarWeatherController service is used to pass data between the Airmar Weather Station and the airmarWeatherStation service. Typically the factory default settings are used without any changes.

Its Regulus class is:

`com.liquidr.regulus.airmar.services.AirmarWeatherController`

Log

Enables or disables logging for the airmarWeatherController service on the Regulus file system.

Console Command	%airmarWeatherController.setLog() Parameters: true or false
Example	%airmarWeatherController.setLog(true)
Expected Output	OK - confirms that logging has been enabled or disabled, depending on which parameter was issued.

Produce JSON status report for this service (Expert)

Returns a JSON formatted status report for the airmarWeatherController service.

Console Command	%airmarWeatherController.serviceReport()
Example	%airmarWeatherController.serviceReport()
Expected Output	JSON formatted status, for example: {"id":101, "info":"airmarWeatherDev:true, airmarLocationDev:true, lastSentence:WIMDA,29.9493,I,1.0142,B,24.5,C,,

Airmar Weather Station Commands

	<pre> ,,,,,130.8,T,117.2,M,1.8,N,0.9,M*", "state":"Happy", "tag":"airmarWeatherController"} </pre>
--	--

Produce status report for this service

Returns comprehensive information about the airmarWeatherController service and its status, including weather station information.

Console Command	<code>%airmarWeatherController.statusReport()</code>
Example	<code>%airmarWeatherController.statusReport()</code>
Expected Output	Information about airmarWeatherController service and the weather station, for example: <pre> id: 101 info: airmarWeatherDev:true, airmarLocationDev:true, lastSentence:WIMDA,29.9641,I,1.0147,B,24.5,C,, ,,,,,129.7,T,116.1,M,1.8,N,0.9,M* state: Happy tag: airmarWeatherController </pre>

airmarWeatherReporter Commands

The airmarWeatherReporter service is used to determine which types of reports are sent to shore, and how frequently reports are transmitted back to shore. The weather station plugin produces two types of reports. The simplified version (PayloadType 0x80) can be found in WGMS under Data->Weather Records. The second type (PayloadType 0x91), which is more comprehensive, can be found in WGMS under Data->Weather Record 2. If "Weather Record 2" is missing in the vehicle's respective ORG, contact the Liquid Robotics Customer Support organization to have Weather Record 2 added.

Its Regulus class is:

```
com.liquidr.regulus.airmar.services.AirmarWeatherReporter
```

Get Weather Report Interval

Returns the current reporting interval setting for weather reports.

Console Command	<code>%airmarWeatherReporter.getReportInterval()</code>
Example	<code>%airmarWeatherReporter.getReportInterval()</code>
Expected Output	Current interval set for reporting, for example: 10m

Log

Enables or disables logging for the airmarWeatherReporter service on the Regulus file system.

Console Command	<code>%airmarWeatherReporter.setLog()</code> • Parameters: true or false
Example	<code>%airmarWeatherReporter.setLog(true)</code>
Expected Output	• OK - confirms that logging has been enabled or disabled, depending on which parameter was issued.

Produce JSON status report for this service (Expert)

Returns a JSON formatted status report for the airmarWeatherReporter service. Use the following key to interpret the report:

reportInterval = Interval for reports to be sent back to shore
oldRpt = Old format of the report, which is only used for comparison and should remain disabled
newRpt = New format of the report with contents CSV encoded in the following order
Time of report (HHMMSS),
Average Temperature (ATEMP), value,
Pressure (PRES), value,
Wind Speed (WSPD), value,
Wind Speed Standard Deviation (WSPD_STD), value,
Wind Direction (WDIR), value
Wind Direction Standard Deviation (WDIR_STD), value,
Wind Speed Max (WSPD_MAX), value,
Gust Direction (GDR), value,
Gust (GST), value,
Gust Time (GTIME-HHMMSS),value
Wind Speed Samples used for report (WSPD_CNT), value

Console Command	<code>%airmarWeatherReporter.serviceReport()</code>
Example	<code>%airmarWeatherReporter.serviceReport()</code>
Expected Output	JSON formatted status, for example: <pre>{ "id": 103, "info": "reportInterval:10m oldRpt:DISABLED newRpt:232000, ATEMP,24.8,PRES,1013.9,WSPD,1.8,WSPD_STD,0.0, WDIR,129.1,WDIR_STD,2.2,WSPD_MAX,2.1,GDR,126.8, GST,1.9,GTIME,231645,WSPD_CNT,600", "state": "Happy", "tag": "airmarWeatherReporter"} </pre>

Produce status report for this service

Returns comprehensive information about the airmarWeatherReporter service and its status, including values from the previous report. Use the following key to interpret the report:

reportInterval = Interval for reports to be sent back to shore
oldRpt = Old format of the report, which is only used for comparison and

should remain disabled
 newRpt = New format of the report with contents CSV encoded in the following order
 Time of report (HHMMSS),
 Average Temperature (ATEMP), value,
 Pressure (PRES), value,
 Wind Speed (WSPD), value,
 Wind Speed Standard Deviation (WSPD_STD), value,
 Wind Direction (WDIR), value
 Wind Direction Standard Deviation (WDIR_STD), value,
 Wind Speed Max (WSPD_MAX), value,
 Gust Direction (GDR), value,
 Gust (GST), value,
 Gust Time (GTIME-HHMMSS), value
 Wind Speed Samples used for report (WSPD_CNT), value

Console Command	<code>%airmarWeatherReporter.statusReport()</code>
Example	<code>%airmarWeatherReporter.statusReport()</code>
Expected Output	Information about airmarWeatherReporter service and the previous report, for example: <pre>id: 103 info: reportInterval:10m oldRpt:DISABLED newRpt:232000, ATEMP,24.8,PRES,1013.9,WSPD,1.8,WSPD_STD,0.0, WDIR,129.1,WDIR_STD,2.2,WSPD_MAX,2.1,GDR,126.8, GST,1.9,GTIME,231645,WSPD_CNT,600 state: Happy tag: airmarWeatherReporter</pre>

Send Shore P80 (Expert)

Enables or disables sending payload 80 reports to shore. For more information see [Appendix B: Weather Record Types](#)

Console Command	<code>%airmarWeatherReporter.setSendPayload80()</code> Parameters: true or false
Example	<code>%airmarWeatherReporter.setSendPayload80(true)</code>
Expected Output	Confirms command was successful, for example: OK

Send Shore P91 (Expert)

Enables or disables comprehensive Weather Records 2. For more information see [Appendix B: Weather Record Types](#)

Console Command	<code>%airmarWeatherReporter.setSendPayload91()</code> Parameters: true or false
-----------------	---

Example	<code>%airmarWeatherReporter.setSendPayload91(true)</code>
Expected Output	Confirms command was successful, for example: OK

Set Weather Report Interval (Expert)

Sets how frequently weather reports are transmitted to shore. By default, in order to be compliant with NOAA's continuous wind measurements, reports are sent at 10 minute intervals starting at the top of the hour. You can change this interval at any time, however the data samples used for averaging will also change. For example, the sensor produces 1 sample / second (1Hz), therefore if reports are taken every 10 min, the total number of samples averaged for that report will be 600 samples. The interval cannot exceed 60min (hourly reports) or be less than 2min.

Console Command	<code>%airmarWeatherReporter.setReportInterval(<WEATHER_INTERVAL>)</code> Where <WEATHER_INTERVAL> is a positive whole number from 2 to 60 that determines how frequently reports are generated in minutes.
Example	<code>%airmarWeatherReporter.setReportInterval("10")</code>
Expected Output	Returns the updated reporting interval or an error message if the specified value is out of range, for example: <pre>interval set: 10m out of range interval (min 2, max 60): 1h1m</pre>

airmarWeatherStation Commands

The airmarWeatherStation service processes meteorological data generated by the Airmar weather station. The stored data can be used for analysis when generating reports. The computation of wind gust speed is also handled by the airmarWeatherStation service.

Its Regulus class is:

```
com.liquidr.regulus.airmar.services.AirmarWeatherStation
```

Factory Defaults (Expert)

Restores all user configurable settings for the airmarWeatherStation service to factory defaults as follows:

```
Log = true
Log 1Hz Weather Station Data = false
Use Filter on Data = false
Roll Threshold = 50
Pitch Threshold = 50
Set Gust Averaging Window = 5s
*Reports will be set back to 10min interval
```

Console Command	<code>%airmarWeatherStation.resetDefaults()</code> Parameters: true or false
-----------------	--

Example	<code>%airmarWeatherStation.resetDefaults(true)</code>
Expected Output	Confirms command was successful, for example: OK

Log

Enable or disable logging for the airmarWeatherStation service.

Console Command	<code>%airmarWeatherStation.setLog()</code>
Example	<code>%airmarWeatherStation.setLog()</code>
Expected Output	Confirms if logging has been enabled or disabled, for example: enabled

Log 1Hz Weather Station Data

Enable or disable logging of raw 1Hz weather station data and legacy 1Hz weather data. Also enables the debug log if it is not already on. The legacy formatted log contains WIMDA data in CSV format only.

Console Command	<code>%airmarWeatherStation.setLogRaw()</code> - Raw data parameters: true or false - Legacy format data parameters: true or false
Example	<code>%airmarWeatherStation.setLogRaw(true, false)</code>
Expected Output	Confirms command was successful, for example: OK

Pitch Threshold (Expert)

Wind sample data that occur when the weather station is at a greater pitch angle (fore or aft) than the pitch threshold setting, are not processed during data analysis provided the Use Filter on Data command is enabled.

Console Command	<code>%airmarWeatherStation.setPitchThreshold()</code> Parameter: Pitch threshold from 0 to 50 degrees.
Example	<code>%airmarWeatherStation.setPitchThreshold(30)</code>
Expected Output	Confirms command was successful, for example: OK

Produce JSON status report for this service (Expert)

Returns a JSON formatted description of the WeatherReporter service.

Use the following key to interpret the status report:

T = Temperature
 P = Pressure
 WSPD = Wind Speed
 WDIR = Wind Direction
 HD = Heading
 SMPLD = Samples processed in the previous report back to shore
 DSMPLD = Time elapsed between data samples
 MIS = Count of how many wind samples that could not be time correlated with attitude data
 FLTR = Filter enabled/disabled
 RRNG = +/- degrees considered valid when the sensor observes roll to use the wind sample
 PRNG = +/- degrees considered valid when the sensor observes pitch to use the wind sample

Console Command	<code>%airmarWeatherReporter.serviceReport()</code>
Example	<code>%airmarWeatherReporter.serviceReport()</code>
Expected Output	JSON formatted status, for example: <pre>{ "id":102, "info":"T=24.1\\u002DAC, P=1013.4mb, WSPD=1.80kn, WDIR=123.8\\u002DA, HD=340.5\\u002DA, SMPLD=600, DSMPLD=1s, MIS=0, FLTR=false, RRNG=\\u00B130.0\\u002DA, PRNG=\\u00B150.0\\u002DA", "state":"Happy", "tag":"airmarWeatherStation"} </pre>

Produce status report for this service

Returns the last meteorological sample processed along with comprehensive information about the airmarWeatherStation service and its status. Use the following key to interpret the status report:

T = Temperature
 P = Pressure
 WSPD = Wind Speed
 WDIR = Wind Direction
 HD = Heading
 SMPLD = Samples processed in the previous report back to shore
 DSMPLD = Time elapsed between data samples
 MIS = Count of how many wind samples that could not be time correlated with attitude data
 FLTR = Filter enabled/disabled
 RRNG = +/- degrees considered valid when the sensor observes roll to use the wind sample
 PRNG = +/- degrees considered valid when the sensor observes pitch to use the wind sample

Console Command	<code>%airmarWeatherStation.statusReport()</code>
Example	<code>%airmarWeatherStation.statusReport()</code>

Expected Output	Information about airmarWeatherStation service along with the previous meteorological report, for example: <pre>id: 102 info: T=23.9°C, P=1013.2mb, WSPD=1.80kn, WDIR=124.3°, HD=340.5°, SMPLD=600, DSMPLD=1s, MIS=0, FLTR=false, RRNG=?30.0°, PRNG=?50.0° state: Happy tag: airmarWeatherStation</pre>
-----------------	---

Roll Threshold (Expert)

Wind sample data that occur when the weather station is at a greater roll angle (port or starboard) than the roll threshold setting, are not processed during data analysis provided the Use Filter on Data command is enabled.

Console Command	<pre>%airmarWeatherStation.setRollThreshold()</pre> Parameter: Roll threshold from 0 to 50 degrees.
Example	<pre>%airmarWeatherStation.setRollThreshold(25)</pre>
Expected Output	Confirms command was successful, for example: OK

Set Gust Averaging Window (Expert)

Sets the time frame in seconds of a "rolling window" used for gust computation. The rolling window is used to break all samples into smaller sets for the duration of the rolling window time frame. When a new rolling window is defined, the existing rolling window time frame is replaced - all future computations will be based on the new rolling window time frame. Existing data will not be recalculated based on the new time frame setting. For more information see [Appendix C: Gust Definition](#)

Console Command	<pre>%airmarWeatherStation.setGustWindow()</pre> Parameter: 2 to (no greater than report interval) seconds
Example	<pre>%airmarWeatherStation.setGustWindow(5)</pre>
Expected Output	Confirms command was successful, for example: OK.

Use Filter on Data (Expert)

When the filter is enabled, wind data samples that are outside of the pitch and roll threshold settings are not processed.

Console Command	<pre>%airmarWeatherStation.setFilterData()</pre> Parameters: true or false
-----------------	--

Airmar Weather
Station Commands

Example	<code>%airmarWeatherStation.setFilterData(true)</code>
Expected Output	• Confirms filter was enabled or disabled, for example: enabled

Chapter 4. Pre-Launch

Before launching the Wave Glider, log in to the vehicle Web page and confirm that all Airmar services are Happy, for example:

	Happy
● airmarWeatherConfig	HW=(Part: 44-848-1-02, SN: 3359992, Model: 6), FW={B1: 2.006, A1: 2.206, B2: 2.003, A2: 2.208, GPS: 13.5.31, COMP: 1.15}, CFG=[ZDA:1:10][GGA:1:10][HDG:1:10][XDRB:1:10][XDRA:1:10][MDA:1:10], POSTS=1, RPTS=1
● airmarWeatherController	airmarWeatherDev:true, airmarLocationDev:false, lastSentence:WIMDA,30.1708,I,1.0217,B,25.4,C,,,,,193.4,T,179.8,M,1.5,N,0.8,M*
● airmarWeatherDeviceDriver	WIMDAHandler,lastRx=0.021s,checksumErr=2,noNmeaLineStart=3,shortLineErr=1, powerState=ON (1d1h31m41.774s), ampsPower=true, Requested by: pilotRequest, SubDevices: HeadingAirmarWeather GPSAirmarWeather
● airmarWeatherReporter	reportInterval:10m oldRpt:DISABLED newRpt:225000,ATEMP,25.5,PRES,1020.6,WSPD,1.4,WSPD_STD,0.0,WDIR,193.4,WDIR_STD,2.8,WSPD_MAX,1.7,GDR,194.3,GST,1.7,GTIME,224416,WSPD_CNT,600
● airmarWeatherStation	T=25.4°C, P(R)=1020.6mb, WSPD=1.50kn, WDIR=193.4°, HD=88.9°, SMPLD=600, DSMPLD=1s, MIS=0, FLTR=false, RRNG=±50.0°, PRNG=±50.0°

Also confirm that the airmarWeatherDeviceDriver shows last Rx as less than 1 second.

Chapter 5. Troubleshooting

If the weather station is not transmitting data, make sure it is turned on. You can confirm the weather station is ON or OFF by viewing **Power Devices** in the vehicle Web page, for example:

	Device	Requires	Requesters
	airmarWaterspeed		
	airmarWeatherDeviceDriver		pilotRequest

You can also confirm if the weather station is ON or OFF by viewing AMPS in the vehicle Web page, for example:

Host Domain

Found in Manifest by Matching a Domain Template.
Domain is Located at CAN Node 1 (Domain Index for Power Reports = 1)
S/N=0x313247163932373900190032, Uptime: 2 days, 22:16:07 (Expected
S/N=0x313247163932373900190032):

Number of Input Bridges = 0, Number of Output Bridges = 5

Host PDC Rebooted @ 2016-10-14T07:00:29Z, Reason=0x15000000, Last Tick was
0x00000000708682b, ResetCodes=KKKK-, KKKK-, --K--, K-K--,,,,, ,
Current ELDB=0x000050e8

Reset PDC OnlyReset PDC & modules

- Slot 4: Regulated Output (13.4V, 6 Ports)
Firmware Version: 0x00003918, Boot Loader Version: 0x0000381e
Mfg. S/N=Dr.weeks#4
HW ID=0x0022002d3032313302473232
Reported Arena Part Number = 4189A (matches manifest)
Module Type = 3
Found in Manifest by Serial Number.
 - #1: GPSAirmarWeather Status=0x00000001 : Power Used= 1.718 W: Port is On:
[Turn Off](#)

Appendix A. National Marine Electronics Association (NMEA) Definitions

DTM Datum Reference

ZDA Time and Date

GGA GPS Fix Data

GLL Geographic Position – Latitude/Longitude

HDG Heading, Deviation and Variation

HDT Heading relative to True North

XDRA Transducer Measurements: wind chill, heat index, and station pressure

XDRB Transducer Measurements: vessel attitude (pitch and roll)

ROT Rate of Turn

VTG Course Over Ground and Ground Speed

VWR Relative Wind Speed and Angle

VWT True Wind Speed and Angle

MWD Wind Direction and Speed, with respect to north

MWVR Wind Speed and Angle, in relation to the vessel's bow/centerline
(relative)

MWVT Wind Speed and Angle, in relation to the vessel's bow/centerline
(theoretical)

MDA Meteorological Composite. Barometric pressure, air temperature,
relative humidity, dew point, wind direction, wind speed

GSA GNSS DOP and Active Satellites

GSV GNSS Satellites in View

RMC Recommended Minimum Specific GNSS Data

ALL Select all sentences

Appendix B. Weather Record Types

The plugin has the ability to transmit two types of data records to shore. Either one can be enabled or disabled using the SV3 Commands window, or the vehicle plugin configuration page. For data analysis, all averages are computed utilizing a simple point average algorithm. No interpolation is done on the data - two data points must be observed within the reporting period, before an average can be computed. Both types of data records are accessible in the WGMS Data page included in the System Menus tab. If Weather Records 2 is not displayed, contact the Liquid Robotics Support organization to have the Weather Records 2 view added.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} (x_1 + \dots + x_n)$$

Figure B.1. Weather Records (PayloadType 0x80)

	Unit	Averaging
TimeStamp	Month,Day,Year, Hour,Minute,Second	N/A
Latitude	0.0001 deg +=N, -=S	N/A
Longitude	0.0001 deg +=E, -=W	N/A
Temperature Average	0.1 C	Point Average
Pressure Average	mBar	Point Average
Wind Speed Average	0.1 knots	Point Average
Wind Speed Standard Deviation	0.1 knots	N/A
Wind Direction Average	0.1 deg	Point Average (Angular)
Wind Direction Standard Deviation	0.1 deg	N/A
Gust Time	Hour,Minute	N/A
Gust Speed	0.1 deg	Point Average
Gust Direction	0.1 deg	Point Average
Samples	sample	N/A

Figure B.2. Weather Records 2 (PayloadType 0x91)

	Unit	Averaging
TimeStamp	Month,Day,Year, Hour,Minute,Second	N/A
Latitude	0.0001 deg +=N, -=S	N/A
Longitude	0.0001 deg +=E, -=W	N/A
Temperature Average	0.1 C	Point Average
Pressure Average	mBar	Point Average
Wind Speed Average	0.1 knots	Point Average
Wind Speed Standard Deviation	0.1 knots	N/A
Wind Direction Average	0.1 deg	Point Average (Angular)
Wind Direction Standard Deviation	0.1 deg	N/A
Gust Time	Hour,Minute	N/A
Gust Speed	0.1 deg	Point Average
Gust Direction	0.1 deg	Point Average
Samples	sample	N/A

Appendix C. Gust Definition

To be comparable with NOAA measurement guidelines, a gust is defined as the maximum 5 second (default) average wind speed measurement taken at time T. However, you can adjust this value. If you change the gust averaging window, all previous averages will be dropped and those samples will not get recomputed until the next report interval. All reports thereafter will use the new gust averaging window specified by the user.

Figure C.1. Gust Definition Table

GUST MEASUREMENT															
	REPORT PERIOD														
T (sec)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Sample[]	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	0	1	2	3	4										
		1	2	3	4	5									
			2	3	4	5	6								
				3	4	5	6	7							
					4	5	6	7	8						
						5	6	7	8	9					
							6	7	8	9	10				
								7	8	9	10	11			
									8	9	10	11	12		
										9	10	11	12	13	
											10	11	12	13	14
												10	11	12	13
													10	11	12
														10	11
Gust[T]	NO VALUE					4	5	6	7	8	9	10	11	12	13

Gust direction averaging should be Point Average (Angular), because the sums of sines and cosines are used to compute the average.