LIQUID ROBOTICS
A Boeing Company

Wave Glider® Integrated Sensor User Guide

Sea-Bird Electronics Glider Payload CTD Sensor Software

Revision B
Item Number: 05574

Trademarks

Liquid Robotics®, Wave Glider®, Wave Glider Management System, WGMS, and SHARC® are worldwide trademarks of Liquid Robotics. 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.

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 Liquid Robotics Terms and Conditions Articles 11.1 Limited Hardware Warranties, 11.2. Limited Service Warranties, and 11.3 Limitations. Nothing herein should be construed as constituting any additional warranties or representations. Liquid Robotics, Inc. shall not be liable for technical or editorial errors or omissions contained herein.

Contact

LIQUID ROBOTICS
A Boeing Company

Liquid Robotics

1329 Moffett Park Drive

Sunnyvale, CA 94089

Phone: +1 408 636 4200

Customer Support: +1 888-574-4574

Fax: +1 408 747 1923

Revision History

Document Part Number	Revision	Publication Date	Changes
05574	A	October 2013	This is the initial release of this document.
05574	B	March 2020	Updated for the SMCv2 and SMC 2.8.0.

Contents

PREFACE

This guide provides information on using the Sea-Bird Electronics Glider Payload CTD (GPCTD) sensor and the corresponding software (SBGPCTD) that is a part of the Sensor Management Computer (SMC). This guide assumes a working familiarity with the Liquid Robotics Wave Glider. The focus is on using the SBGPCTD software.

Manuals on the GPCTD sensor can be found here https://www.seabird.com/moving-platform/glider-payload-ctd-gpctd/family-downloads?productCategoryId=54627473789.

Liquid Robotics manuals are available on the Open Oceans Portal (http://openoceans.liquid-robotics.com/) and by contacting Liquid Robotics Customer Support at:

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

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

1 INTRODUCTION

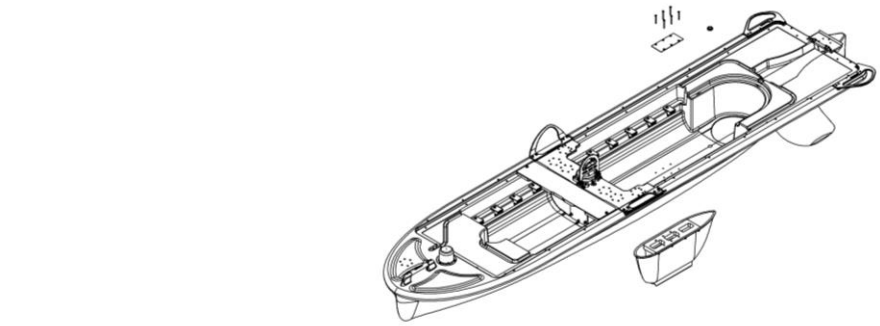
The SBGPCTD software, along with a Sea-Bird Electronics Glider Payload CTD, is not included with the base Wave Glider SV3 system. The system includes a Sensor Management Computer (SMC) payload on which the SBGPCTD software runs. Payload enclosures and electronics can be purchased for connection either to one of the SV3 Wave Glider Expansion ports (G Port) or Sensor ports (S Port).

The Liquid Robotics integration of the Sea-Bird Electronics Glider Payload CTD (GPCTD) into a Wave Glider provides a mobile, long endurance platform for collecting conductivity, temperature and pressure data (and optionally dissolved oxygen data).

1.1 Mechanical Design

The Sea-Bird Electronics GPCTD can be mounted under a Wave Glider SV3 float, attached to the sub or attached to a tow fish. Typical float and sub installations are shown below. Contact Liquid Robotics for more information about the GPCTD integrated into a tow fish.

Figure 1.1 GPCTD Mounted Under the Float



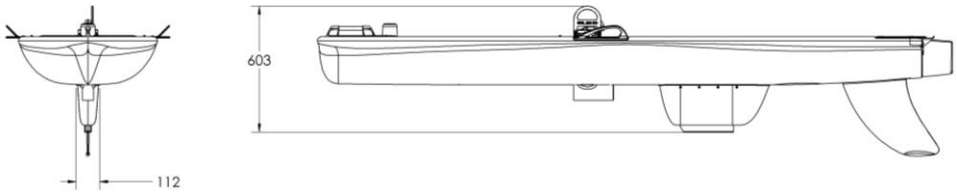
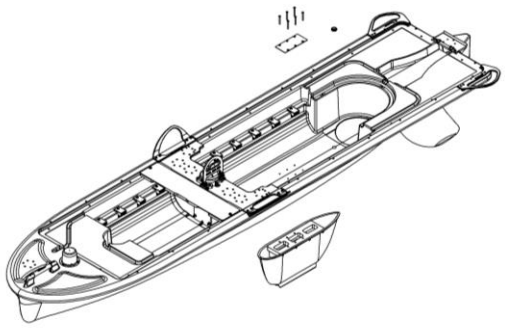
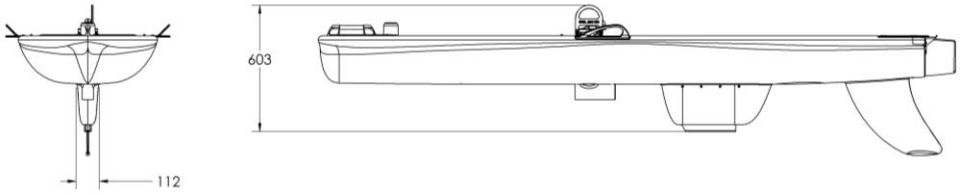


Figure 1.2 GPCTD Mounted to a Sub





1.2 Software Overview

The SeabirdGPCTD service running on the Sensor Management Computer provides an interface to a Sea-Bird Electronics Payload CTD, with or without a dissolved oxygen sensor. Its primary features are:

- A command interface that allows the user to start or stop sampling, request status, send low-level diagnostics commands and configure logging and startup actions.
- On-board log of samples and commands and responses.
- Parameterized sampling scenario that specifies how samples are concatenated into telemetry reports, sleep time between reports and input line flush times before starting to sample.

The SBGPCTD software is agnostic to where the sensor is mounted. The software receives samples from the sensor in the Sea-Bird Electronics native format and adds GPS date, time, latitude, and longitude to each sample.

On expansion port payloads, SBGPCTD commands can be issued from the WGMS SV3 Commands Window UI or by sending a Write Console Script command. For more information see Sending SBGPCTD Commands (Expansion Port Only). On sensor port payloads, SBGPCTD commands can be issued only by Write Console Script commands. For more information see Using a Write Console Script to Issue SBGPCTD Commands.

SBGPCTD commands can also be issued for both types of payloads by logging into the SMC and using ServerExec. This is typically done locally through the debug port. For more information see Issuing SBGPCTD Commands on the SMC.

2 SBGPCTD COMMANDS

This section provides information on the SBGPCTD commands. The various ways that the commands can be issued to a vehicle that is powered on along with the payload, and the syntax used, depends on how the SMC is connected to the Wave Glider (expansion port or sensor port), or if you are logged directly into the SMC in the payload.

- SBGPCTD commands can be issued from the WGMS SV3 Commands Window GUI (only for payloads connected to an expansion port). For information on how to issue commands in the WGMS Command Window see Sending SBGPCTD Commands (Expansion Port Only). For information on the SBGPCTD commands that appear in the WGMS SV3 Commands Window see SBGPCTD Commands (Expansion Port – WGMS SV3 Commands Window).
- SBGPCTD commands can be issued using WGMS Write Console Script commands when the payload is connected to a sensor port. For information on how to issue SBGPCTD commands using a Write Console Script see Using a Write Console Script to Issue SBGPCTD Commands.
- SBGPCTD commands can be issued directly to the SMC in the payload by logging into the SMC and using ServerExec to issue commands. This method will work regardless of the type of port you are connected to. For information on how to issue SBGPCTD commands directly to the SMC using ServerExec see Issuing SBGPCTD Commands on the SMC.

2.1 Sending SBGPCTD Commands (Expansion Port Only)

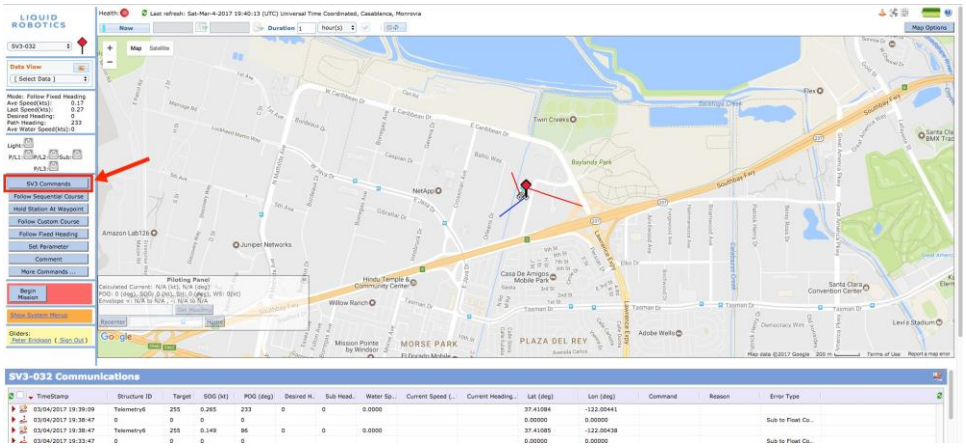
2.1.1 From WGMS

On SV3 vehicles, SBGPCTD commands can be issued from the WGMS SV3 Commands UI.

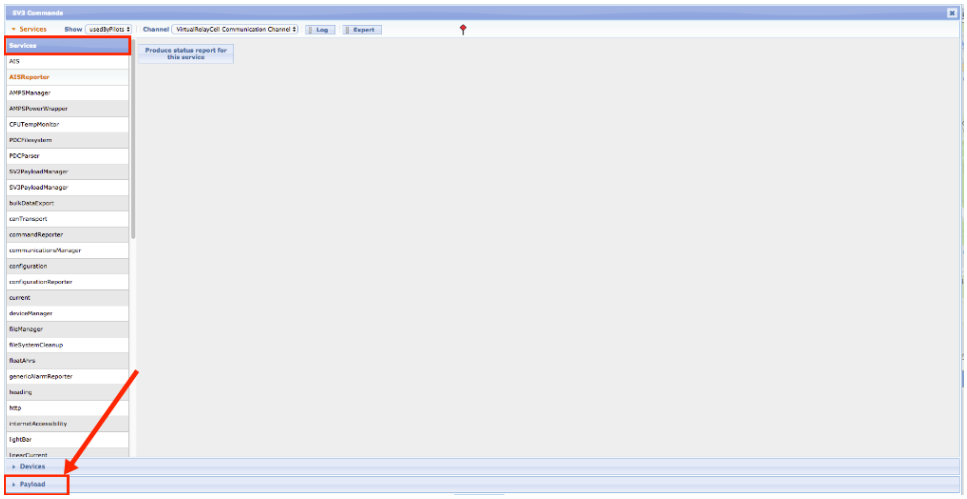
Note

This section focuses on issuing SBGPCTD commands. For general information on using the SV3 Commands Window, refer to the *WGMS User Guide* available on the Liquid Robotics **Open Oceans Portal**.

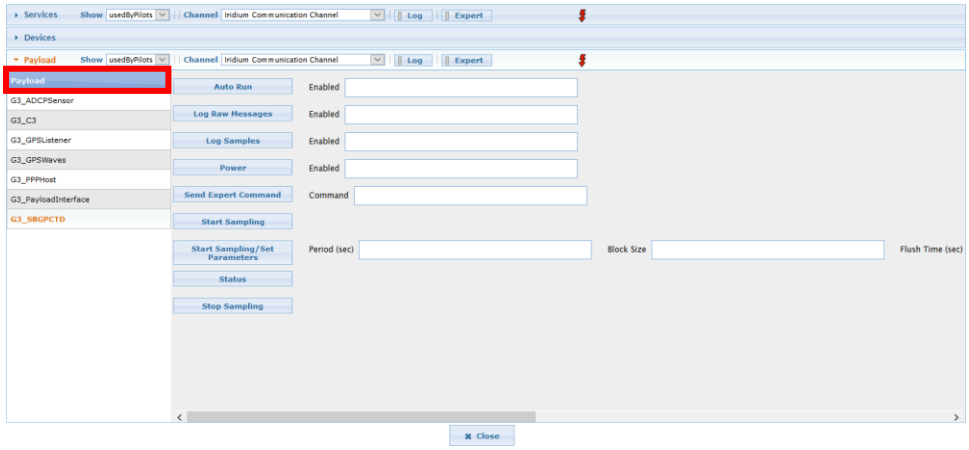
To open the SV3 Commands window, click on **SV3 Commands** in the WGMS Dashboard:



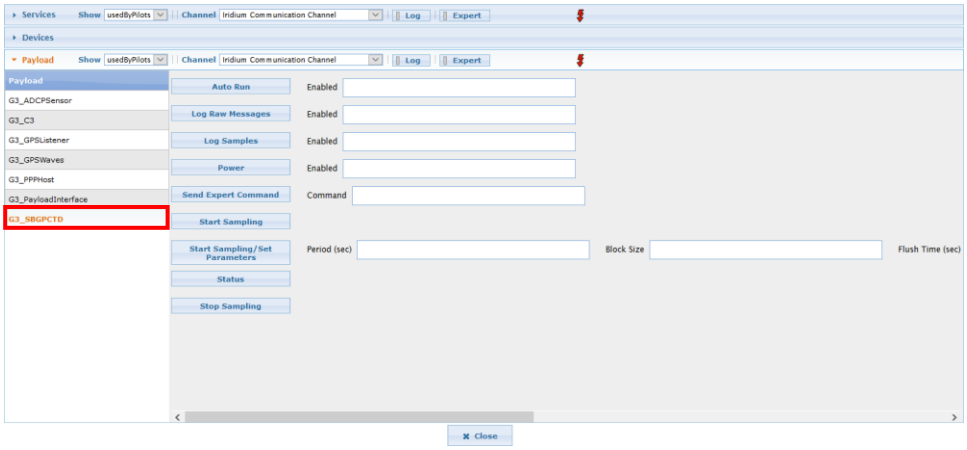
The **SV3 Commands** Window opens to the **Services** menu by default.



Click on **Payload** to open the **Payload** menu.



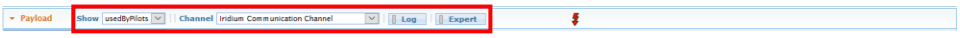
Click on **G1_SBGPCTD**, **G2_SBGPCTD**, or **G3_SBGPCTD** in the **Payload** menu, depending on which expansion port the payload is connected to, to display the SBGPCTD commands.



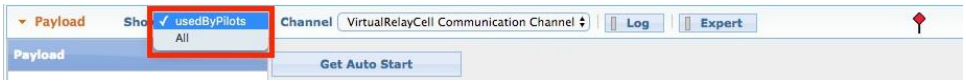
By default,

- **usedByPilots** will be selected in the **Show** drop-down menu.
- The communications channel currently in use by the Wave Glider will be selected in the **Channel** drop-down menu.
- The **Log** button will be OFF.

- The **Expert** button will be OFF.



The **Show** drop-down menu has 2 options, **usedByPilots** and **All**. For SBGPCTD use the **usedByPilots** option.



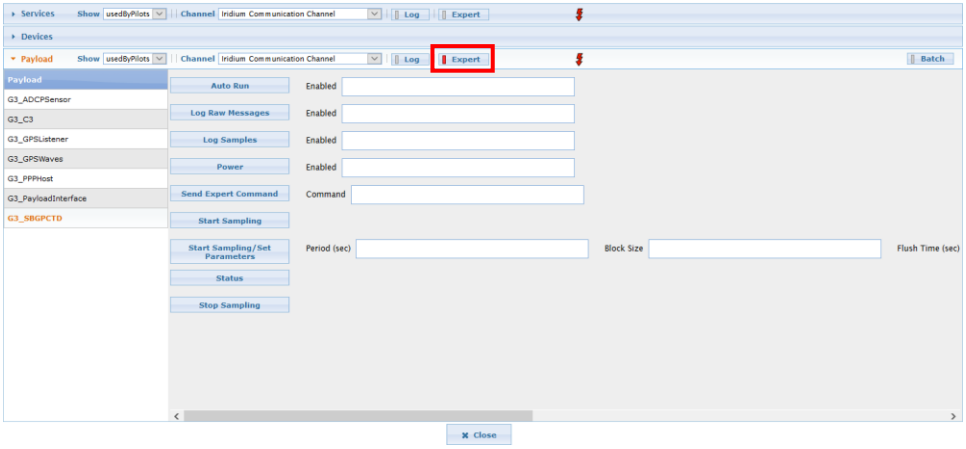
Use the **Channel** drop-down to select the communications channel you want to use to transmit commands and receive acknowledgements and data using WGMS. The **Channel** drop-down menu includes the communication channel options that are configured for the Wave Glider. All these communication channels may not be usable, depending on settings and on where the Wave Glider is located. For example, if you are inside a building, the **Iridium Communication Channel** will probably not be usable. If you attempt to send a command over the Iridium channel when the vehicle does not have a full view of the sky, transmission will be attempted, but the command will not be received by the vehicle. On the other hand, if you select the **VirtualRelay Cell Communication Channel** when the cell modem is disabled, transmission will be attempted but will fail.

Click on the **Log** button to display a log of the communication between WGMS and the vehicle.

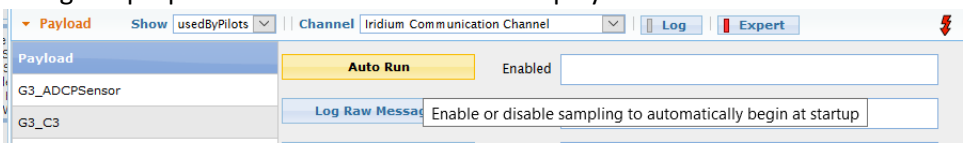
Click on the **Expert** button to display expert level commands, in addition to pilot commands. A warning icon is displayed next to each expert command button.

Note

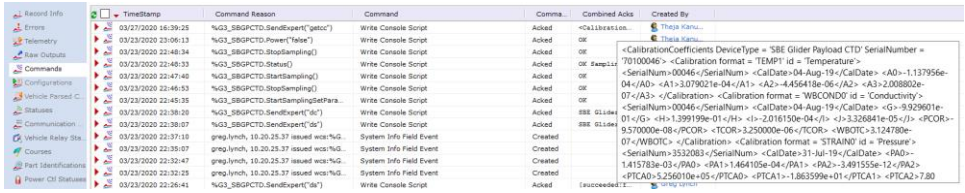
There are no expert level SBGPCTD commands currently.



When you hover your cursor over a command button, the button color will change blue to yellow and a tool tip explaining the purpose of the command will be displayed.

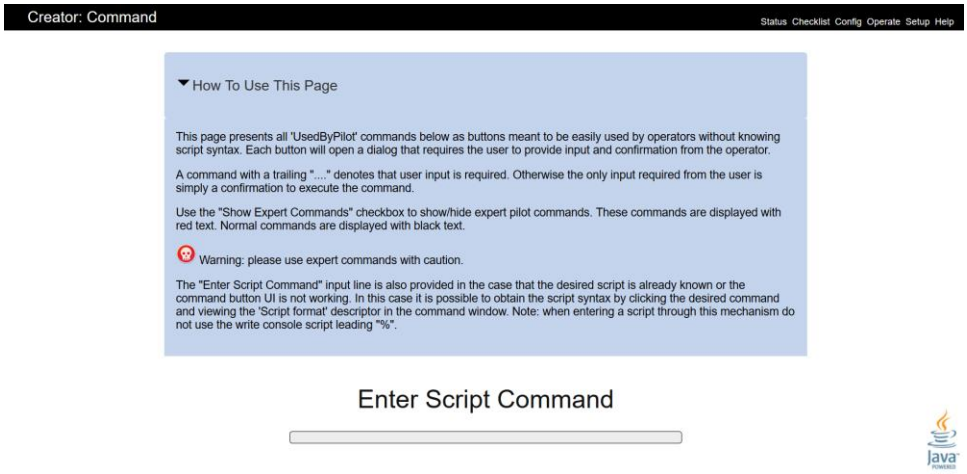


When you click on a command button, a write console script is issued to call the corresponding method in the SBGPCTD software. For example, the **Start Sampling** button in the **G1_SBGPCTD**, **G2_SBGPCTD**, or **G3_SBGPCTD** payload UI calls the `StartSampling` method in the SBGPCTD software to start sampling the GPCTD sensor. The result of the command is displayed in the **Combined Acks** column in the **Commands** log at the bottom of the **WGMS** dashboard and in the vehicle specific **<Vehicle Name> Commands** window available from the **Vehicle** page.

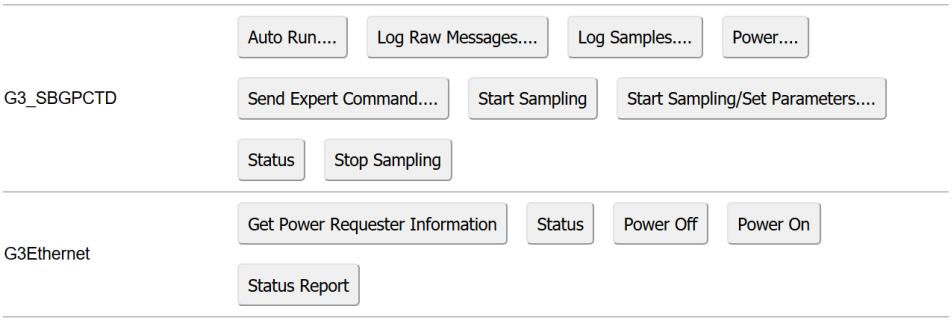


2.1.2 From the Vehicle Web Page

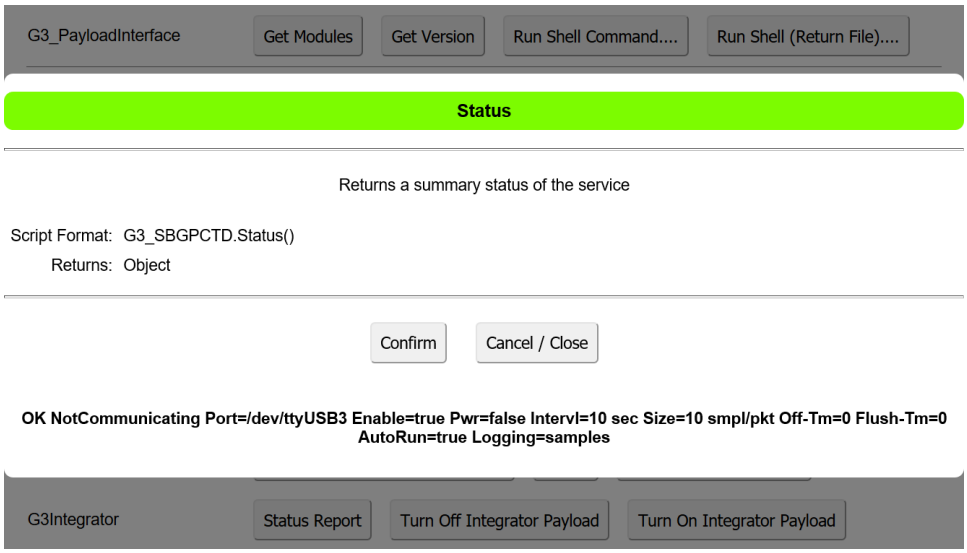
To issue a SBGPCTD command to an expansion port payload from the SV3 vehicle web page open the vehicle web page, click on **Enter script command** in the **Operate** drop-down menu, then follow the instructions in the Readme “How To Use This Page”.



Scroll down to the **G1_SBGPCTD**, **G2_SBGPCTD**, or **G3_SBGPCTD** section of the page. You’ll find buttons corresponding to the commands covered in SBGPCTD Commands (Expansion Port – WGMS SV3 Commands Window).



Click on the button corresponding to the command you wish to send. In the resulting window enter any parameters, if necessary, and click **Confirm** to send the command.

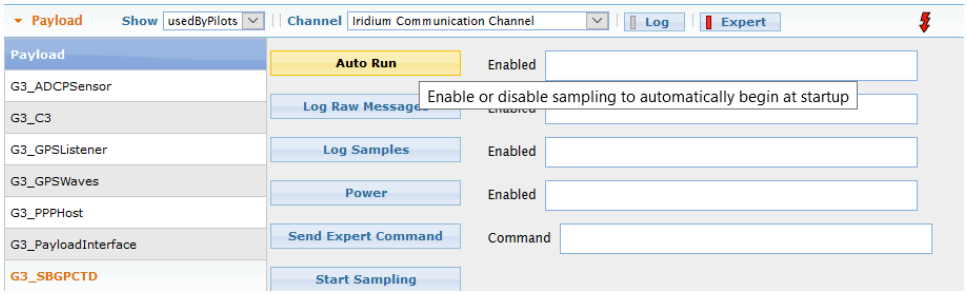


2.2 SBGPCTD Commands (Expansion Port – WGMS SV3 Commands Window)

There are 9 SBGPCTD commands available from the WGMS SV3 Commands Window.

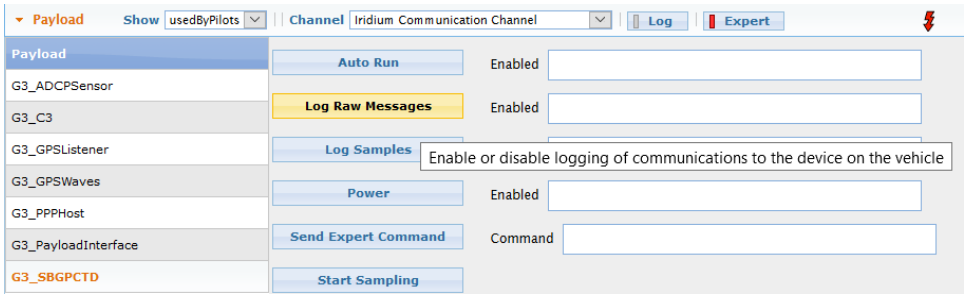
2.2.1 Auto Run

Enter true or false then click **Auto Run** to change the setting. This command enables or disables sampling to automatically begin at startup of the payload.



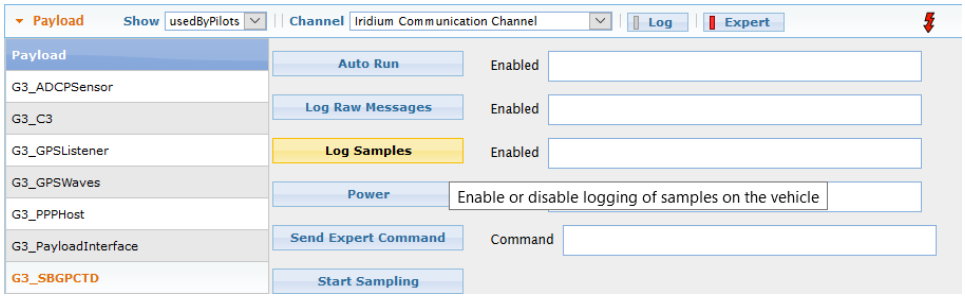
2.2.2 Log Raw Messages

Enter true or false in the field, then click **Log Raw Messages** to enable or disable logging of communications to a timestamped data file on the SMC filesystem.



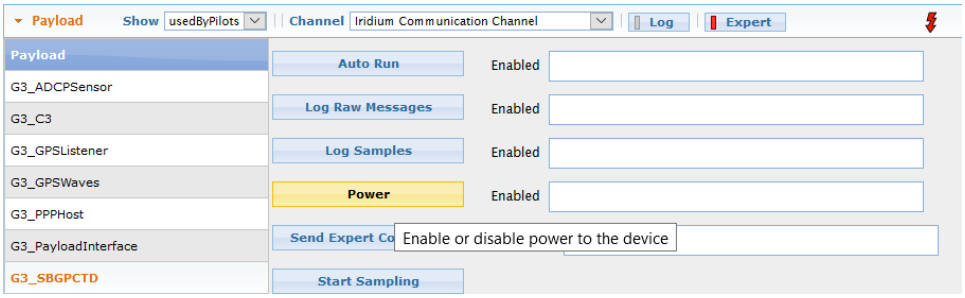
2.2.3 Log Samples

Enter true or false in the field, then click **Log Samples** to enable or disable logging of samples to a timestamped data file on the SMC.



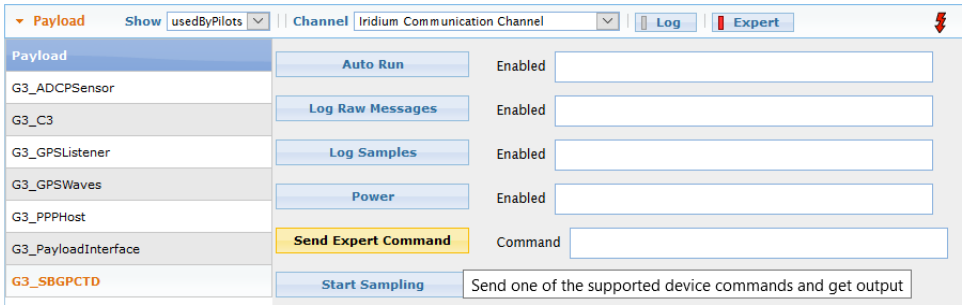
2.2.4 Power

Enter true or false then click **Power** to change the setting. This command enables or disables power to the device.



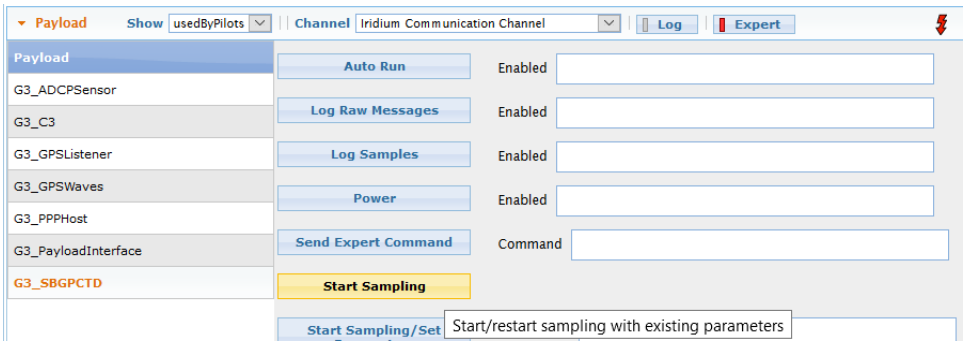
2.2.5 Send Expert Command

Click **Send Expert Command** to send one of the supported device commands and get the output. See B.5 Send Expert Command for the supported device commands.



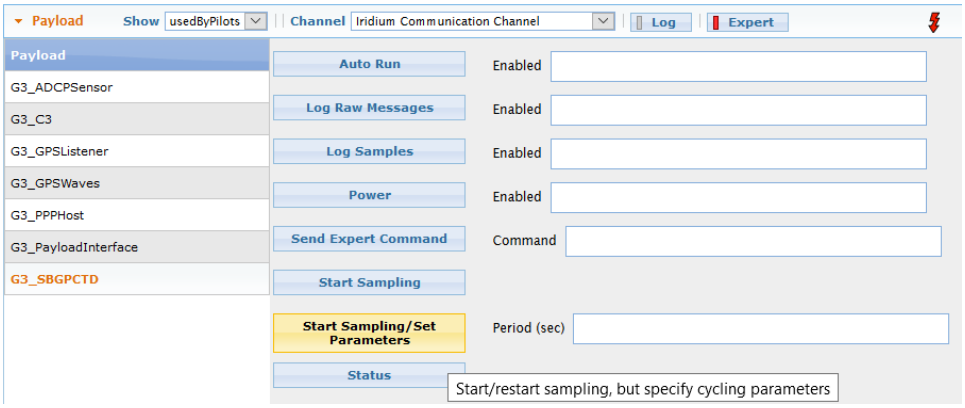
2.2.6 Start Sampling

Click **Start Sampling** to start/restart sampling with existing parameters.



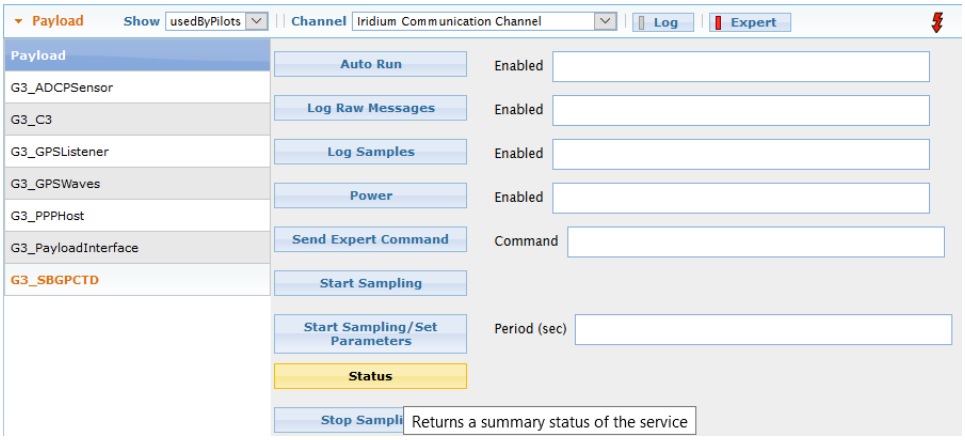
2.2.7 Start Sampling/Set Parameters

Click **Start Sampling/Set Parameters** to start/restart sampling but specify cycling parameters.



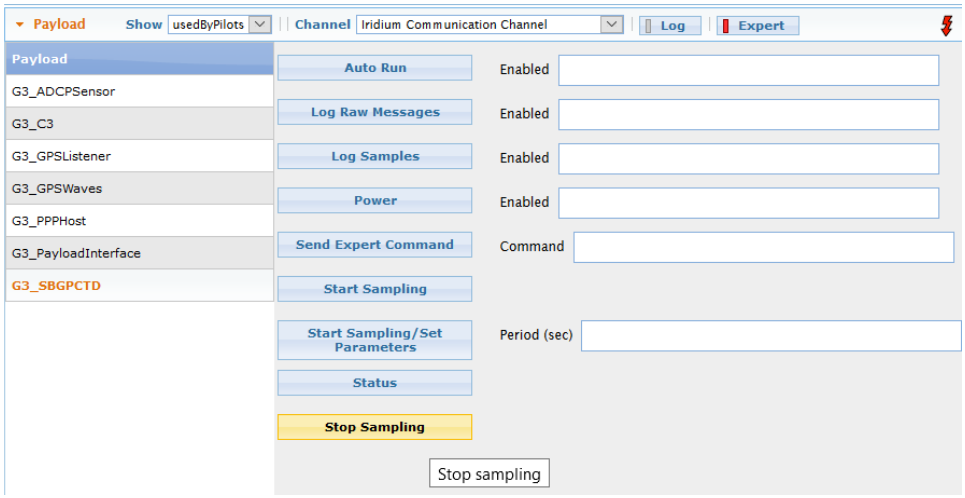
2.2.8 Status

Click **Status** to return a summary status of the service.



2.2.9 Stop Sampling

Click **Stop Sampling** to stop sampling the sensor. It is recommended to send a Power 'false' command after this to ensure the sensor is off and not sampling.



2.3 Using a Write Console Script to Issue SBGPCTD Commands

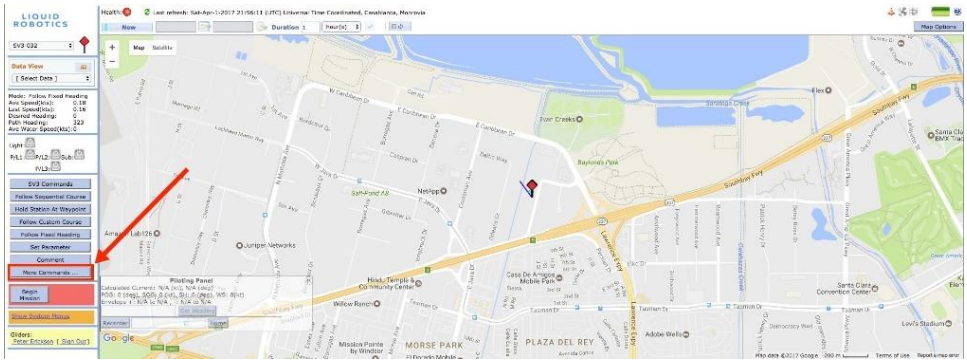
You can issue SBGPCTD commands by entering command syntax into a Write Console Script. You can issue a Write Console Script from WGMS or from the Wave Glider SV3 vehicle web page.

- If the SMC payload is connected to an expansion port use the command syntax described in Appendix B: SBGPCTD Command Syntax for Expansion Port Payloads.
- If the SMC payload is connected to a sensor port use the command syntax described in Appendix C: SBGPCTD Command Syntax for Sensor Port Payloads.

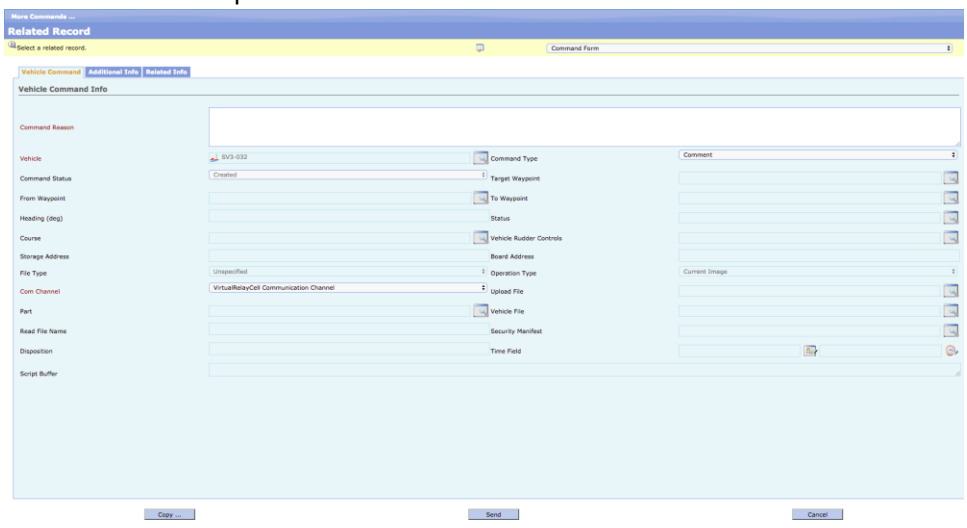
2.3.1 From WGMS

To issue a SBGPCTD command from WGMS using a Write Console Script.

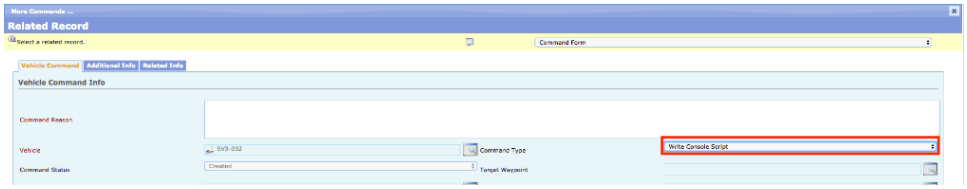
- From the WGMS Home Page, click on the **More Commands…** button.



- A command form will open.

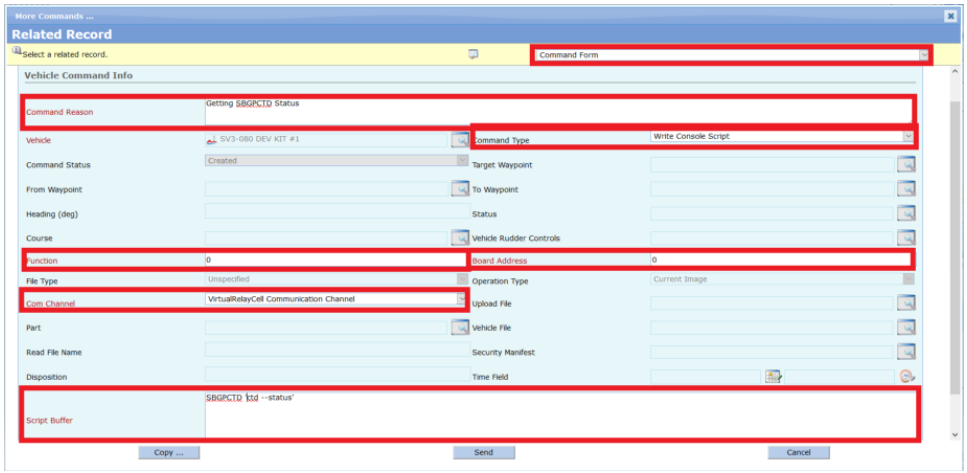


- Select **Write Console Script** in the **Command Type** drop-down menu.



- Complete the required fields that appear in **red** text, then click **Send**.

- **Use Function 0 and Board Address 8230 for Sensor Port payloads**. Also see Appendix C: SBGPCTD Command Syntax for Sensor Port Payloads.
- **Use Function 0 and Board Address 0 for Expansion Port payloads**. Also see Appendix B: SBGPCTD Command Syntax for Expansion Port Payloads.



2.3.2 From the Vehicle Web Page

To issue a SBGPCTD command from the SV3 vehicle web page:

- Open the vehicle web page, click on **Write Console Script** in the **Operate** drop-down menu.
- Complete the required fields then click Send.
  - **Use Function 0 and Board Address 8230 for Sensor Port payloads**. Also see Appendix C: SBGPCTD Command Syntax for Sensor Port Payloads.
  - **Use Function 0 and Board Address 0 for Expansion Port payloads**. Also see Appendix B: SBGPCTD Command Syntax for Expansion Port Payloads.

2.4 Issuing SBGPCTD Commands on the SMC

Log into the SMC then use ServerExec to issue SBGPCTD commands. Refer to "Connecting to the Sensor Management Computer" for instructions on how to log into a payload.

The command syntax when issuing SBGPCTD commands on the SMC is the same as the sensor port command syntax, except that each command is preceded by **`ServerExec`**. This is true for both expansion port payloads and sensor port payloads. For example, the command set for issuing commands to the SMC on a payload is:

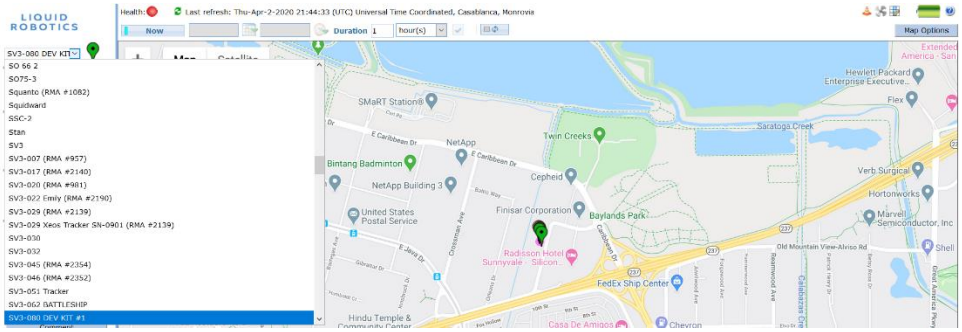
```
ServerExec SBGPCTD "ctd -autorun [true/false]"
ServerExec SBGPCTD "ctd -log-raw [true/false]"
ServerExec SBGPCTD "ctd --start"
ServerExec SBGPCTD "ctd --stop"
```

For more information on these commands see Appendix C: SBGPCTD Command Syntax for Sensor Port Payloads.

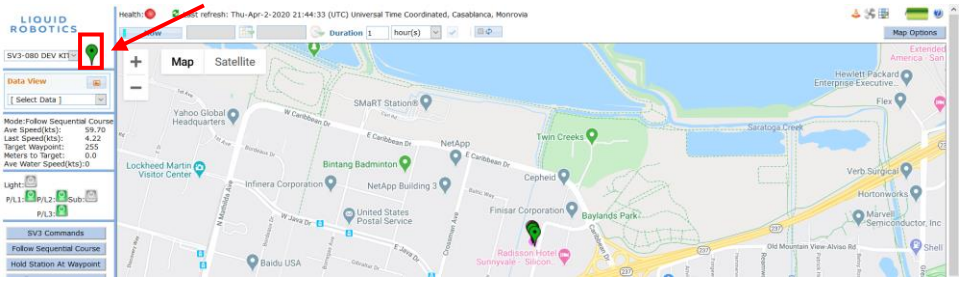
3 VIEWING THE COMMAND RESPONSE

WGMS records the command response sent when the command you submit is received and acknowledged. You can view this response to confirm that the command has been received and to review the state of the sensor at the time the command was received.

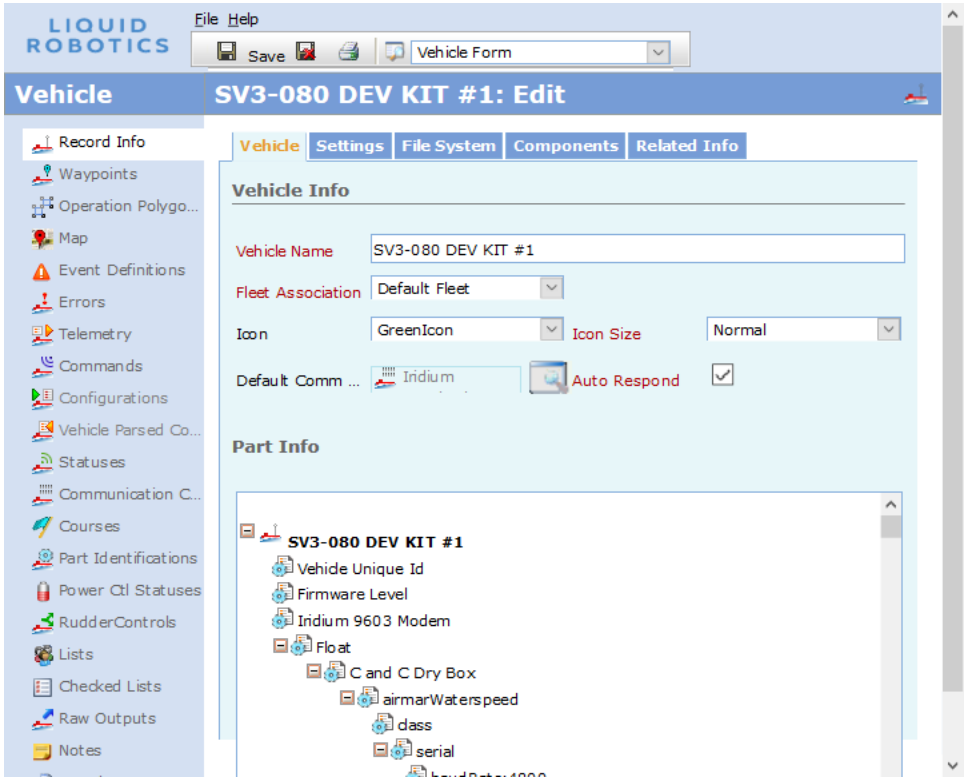
1. On the WGMS page, select the vehicle you are working with in the drop down.



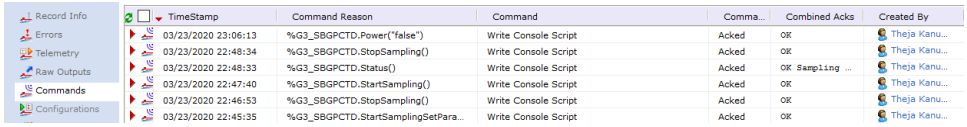
2. Click on the vehicle icon.



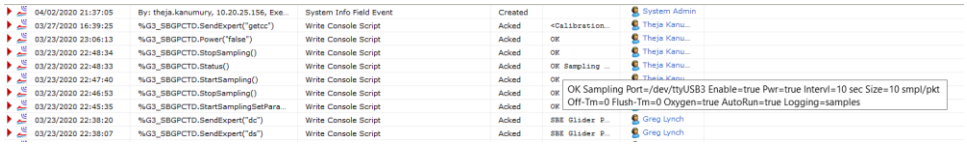
3. A vehicle window should pop-up.



4. Click **Commands**.



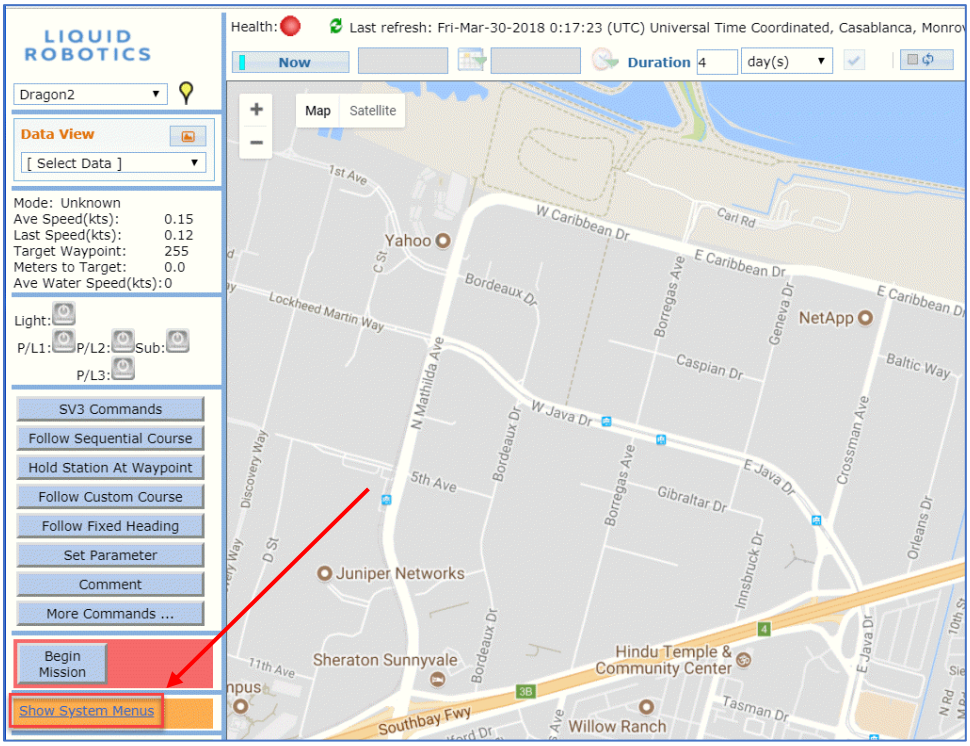
5. Enter SBGPCTD in the search field to filter out non-SBGPCTD records.
6. Hover over the **Combined Acks** to view the full response.



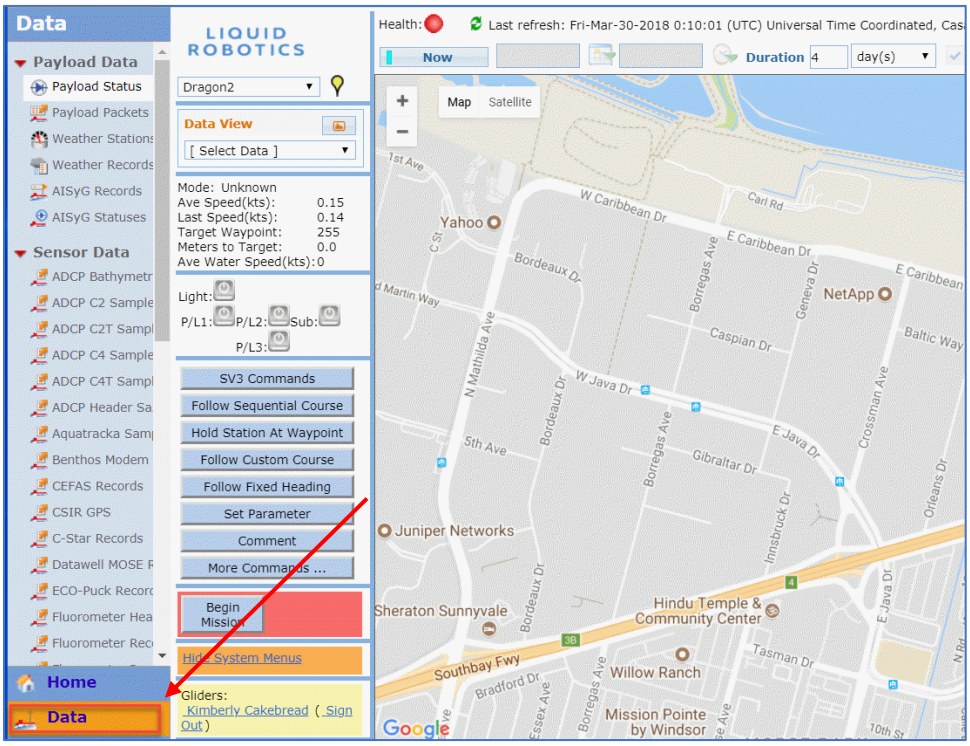
4 VIEWING THE DATA

To view the data being collected and sent to shore via Wave Glider communication paths:

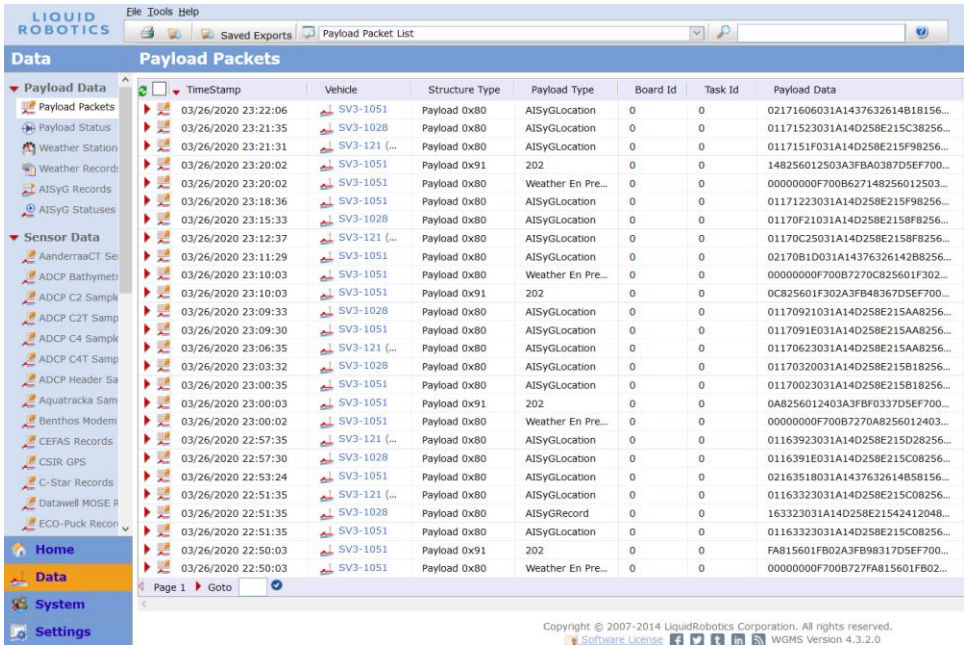
1. Go to the vehicle page.



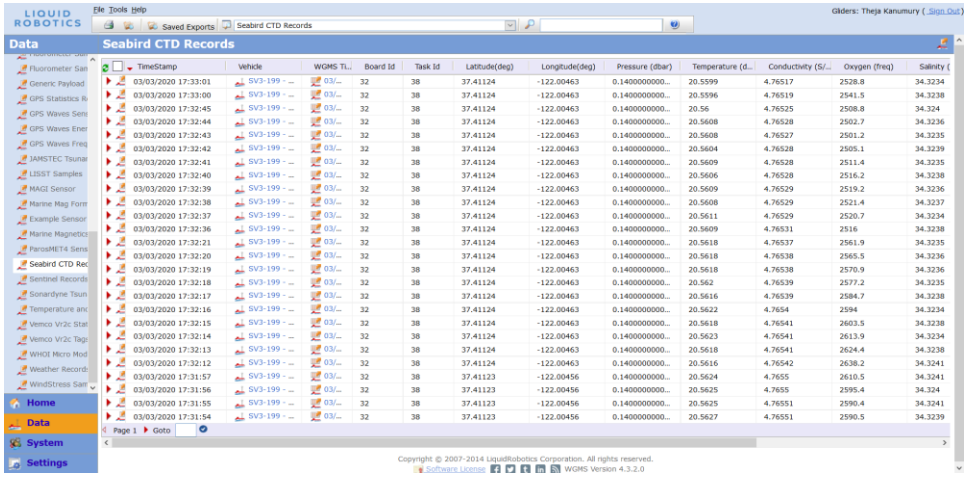
2. If System Menus are not displayed, click **Show System Menus**.



3. Click the **Data** panel.



4. Click **Seabird CTD Records**.



APPENDIX A: SBGPCTD CONFGURATION FILE

The SBGPCTD configuration file is located at:

`/home/smc/roots/200_SBGPCTD/SeabirdGPCTD-config.xml`

The salient configuration settings are:

- The CORBA server name and type. As with every service, the server name must be unique, and the name is used whenever executing the command via ServerExec:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!DOCTYPE boost_serialization>
<boost_serialization signature="serialization::archive" version="10">
<rootlevel class_id="0" tracking_level="0" version="1">
    <sensorProgramSettings class_id="1" tracking_level="0" version="1">
            <serverName>SBGPCTD</serverName>
            <serverKind></serverKind>
```

- The board ID and task ID, which make up the board address used externally (from WGMS or the SV3 web page) to send commands to the process, and the address stamped onto telemetry samples. It is recommended to set the board ID to 255, which means that the board ID configured in PayloadManager is used. The readable name should match the server name.

```
<payloadManagerClientSettings class_id="11" tracking_level="0" version="0">
    <serviceName>PayloadManager</serviceName>
    <serviceKind></serviceKind>
    <deviceAddedEventName>PayloadDeviceAdded</deviceAddedEventName>
    <deviceChangedEventName>PayloadDeviceChanged</deviceChangedEventName>
    <deviceRemovedEventName>PayloadDeviceRemoved</deviceRemovedEventName>
    <commonEventType>PayloadRegistrationEvent</commonEventType>
    <commonEventDomain>LiquidR-SMC</commonEventDomain>
    <payloadDescription class_id="12" tracking_level="0" version="2">
            <boardID>255</boardID>
            <taskID>38</taskID>
            <description>Seabird GPCTD interface</description>
            <ior></ior>
            <readableName>SBGPCTD</readableName>
            <descriptiveJSON></descriptiveJSON>
    </payloadDescription>
</payloadManagerClientSettings>
```

- The sensor is powered on and off using the powerOnVerb and the powerOffVerb. Use `sudo SetPowerSwitch <switch-number> <new-state>` (SMCv1) `SetAMPSPowerSwitch <module-number> <slot-number> <new-state>` (SMCv2)

```
<sensorSettings class_id="14" tracking_level="0" version="1">
    <powerOffVerb>SetAMPSPowerSwitch 1 1 0</powerOffVerb>
    <powerOnVerb>SetAMPSPowerSwitch 1 1 1</powerOnVerb>
    <serialListenerSettings class_id="15" tracking_level="0" version="0">
            <maxReadSize>1024</maxReadSize>
            <intercharacterGapTimeoutMs>50</intercharacterGapTimeoutMs>
    </serialListenerSettings>
    <serialPortSettings class_id="16" tracking_level="0" version="0">
            <portName>/dev/ttyUSB3</portName>
            <baudRate>115200</baudRate>
            <parity>NONE</parity>
            <characterSize>8</characterSize>
```

```
            <stopBits>1</stopBits>
            <flowControl>NONE</flowControl>
            <rts>1</rts>
            <dtr>1</dtr>
            <breakTimeoutMultiplier>1</breakTimeoutMultiplier>
    </serialPortSettings>
</sensorSettings>
```

The serial port device name (portName) and the baud rate to be used (baudRate) can also be set.

- There is a setting for logging data on the SMC that should be modified if a user is working with multiple GPCTDs. Specifically, the directory where the data is stored should be changed to prevent output from different sensor instances from getting intermixed or overwritten:

```
<fileSetSettings class_id="14" tracking_level="0" version="0">
    <archiveCompressionType>BZIP2</archiveCompressionType>
    <archiveDirectory></archiveDirectory>
    <extension>.CSV</extension>
    <keepWritesTogether>1</keepWritesTogether>
    <maxBytesQueued>10000</maxBytesQueued>
    <maxFileSize>1000000</maxFileSize>
    <maxFiles>1000</maxFiles>
    <maxRetainTime>00:00:00</maxRetainTime>
    <prefix>SBGPCTD</prefix>
    <purgeStrategy>OLDEST</purgeStrategy>
    <workingDirectory>/mnt/XDATA/SBGPCTD</workingDirectory>
</fileSetSettings>
```

APPENDIX B: SBGPCTD COMMAND SYNTAX FOR EXPANSION PORT PAYLOADS

The following commands can be issued to the SBGPCTD application running on the SMC in a payload that is connected to an expansion port.

B.1 Auto Run

Method Name	"*AutoRun*"
Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.AutoRun()
Parameter Types	["enabled (true/false)"]
Description	Enable or disable sampling to automatically begin at startup of the payload. Leave blank to return whether enabled or disabled.
Example	%G1_SBGPCTD.AutoRun("true")
Expected Output	`true`

B.2 Log Raw Messages

Method Name	"*LogRawMsgs*"
Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.LogRawMsgs()
Parameter Types	["enabled (true/false)"]
Description	Enable or disable logging of communication to the device on the SMC. Leave blank to return whether enabled or disabled.
Example	%G1_SBGPCTD.LogRawMsgs("false")
Expected Output	`false`

B.3 Log Samples

Method Name	"*LogSamples*"
Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.LogSamples()
Parameter Types	["enabled (true/false)"]
Description	Enable or disable logging of samples on the SMC. Leave blank to return whether enabled or disabled.
Example	%G1_SBGPCTD.LogSamples("true")
Expected Output	`true`

B.4 Power

Method Name	"*Power*"
Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.Power()
Parameter Types	["enabled (true/false)"]
Description	Enable or disable power to the device.
Example	%G1_SBGPCTD.Power("true")
Expected Output	`OK`

B.5 Send Expert Command

Method Name	"*SendExpert*"

Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.SendExpert()
Parameter Types	[java.lang.String]
Description	Send one of the supported device commands and get output: • dc (display calibration coefficients) • ds (display status) • getcc (get calibration coefficients) • getcd (get configuration data: XML) • getec (get event counters: XML) • gethd (get hardware configuration: XML) • getsd (get status data: XML) • sl (send last sample) • slp (send last pressure sample)
Example	%G1_SBGPCTD.SendExpert("ds")
Expected Output	SBE Glider Payload CTD 1.0A SERIAL NO. 0046 23 Mar 2020 22:38:43 vMain = 17.12, vLith = 2.84 autorun = no samplenumber = 123461, free = 435779, profiles = 1500 not logging, never started sample every 10 seconds sample mode is interval, continuous pump data format = converted engineering do not force on RS232 transmitter transmit real time data acquire SBE 43 oxygen minimum conductivity frequency = 3000.0 custom pump mode enabled

B.6 Start Sampling

Method Name	"*StartSampling*"
Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.StartSampling()
Parameter Types	none
Description	Start/restart sampling with existing parameters.
Example	%G1_SBGPCTD.StartSampling()
Expected Output	OK

B.7 Start Sampling/Set Parameters

Method Name	"*StartSamplingSetParam*"
Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.StartSamplingSetParam()
Parameter Types	[java.lang.String, java.lang.String, java.lang.String, java.lang.String]
Description	Start/restart sampling but specify cycling parameters. • period (sec) is the number of seconds between samples. Note that this value cannot be over

	14 seconds if an O2 sensor is installed (see GPCTD manual for further detail). • samples-per-block is the number of samples per telemetry message, and number of samples between imposing an off time. • flush-time (sec) is the number of seconds to run the pump (to get water in the instrument) before sampling. • off-time (sec) is the number of seconds to suspend pumping and sampling between blocks. If this value is zero, sampling is continuous.
Example	%G1_SBGPCTD.StartSamplingSetParam(10,8,110,600)
Expected Output	`OK`

B.8 Status

Method Name	"*Status*"
Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.Status()
Parameter Types	none
Description	Returns a summary status of the service.
Example	%G1_SBGPCTD.Status()
Expected Output	`OK Sampling Port=/dev/ttyUSB3 Enable=true Pwr=true Intervl=10 sec Size=10 smpl/pkt Off-Tm=0 sec Flush-Tm=0 Oxygen=true AutoRun=true Logging=samples`

B.9 Stop Sampling

Method Name	"*StopSampling*"
Expert Command	No
Return Types	java.lang.String
Console Command	%G1_SBGPCTD.StopSampling()
Parameter Types	none
Description	Stop sampling. Recommended to send `%G1_Power("false")` after the stop sampling command to ensure the sensor is off and not sampling.
Example	%G1_SBGPCTD.StopSampling()
Expected Output	`OK`

APPENDIX C: SBGPCTD COMMAND SYNTAX FOR SENSOR PORT PAYLOADS

There are two ways you can issue commands to the SBGPCTD application running on the SMC when the payload is connected to a sensor port.

- From WGMS using the Write Console Script command with the Board Address set to decimal 8230 (hex `0x2026`).
- On the SMC using ServerExec to send commands to the process named SBGPCTD.

C.1 SBGPCTD Commands

C.1.1 ctd --autorun <on-or-off>

Enables (on/true) or disabled (off/false) logging of all messages and responses from the sensor. If left blank, it will reply with the current state.

C.1.2 ctd --expert <command>

Send one of the supported device commands and receive output. The supported commands are:

- dc (display calibration coefficients)
- ds (display status)
- getcc (get calibration coefficients)
- getcd (get configuration data: XML)
- getec (get event counters: XML)
- gethd (get hardware configuration: XML)
- getsd (get status data: XML)
- sl (send last sample)
- slp (send last pressure sample)

C.1.3 ctd --power <on-or-off>

Enables (on/true) or disables (off/false) power to the sensor. Note that power is automatically turned on when sampling is started, see C.1.4, so there's no need to send this command before sending a start command.

C.1.4 ctd –start {<period(sec)>, <samples-per-block>, <flush-time(sec)>, <off-time(sec)>}

Turns on power to the sensor and starts taking readings, returning telemetry for each sample. If included, the parameters overwrite the currently configured values (the new values are persistent).

- period(sec) is the number of seconds between samples. Note that this value cannot be over 14 seconds if an O2 sensor is installed (see GPCTD manual for further detail).
- samples-per-block is the number of samples per telemetry sample, and number of samples between imposing an off time.
- flush-time(sec) is the number of seconds to run the pump (to get water in the instrument) before sampling.
- off-time(sec) is the number of seconds to suspend pumping and sampling between blocks. If this value is zero, sampling is continuous.

C.1.5 ctd --status

Returns a message representing the state of the sensor.

C.1.6 ctd --stop

Stops the sampling loop, but leaves the sensor powered (in low power state). To ensure that the sampling loop is stopped follow the `ctd --stop` command with `ctd --power off`.

APPENDIX D: SBGPCTD CALIBRATION

Seabird Electronics recommends yearly calibration of the GPCTD. For instructions on removing the instrument from your Wave Glider contact Liquid Robotics Customer Support. The SMC software was written to work with GPCTD firmware 1.0A. When returning your instrument to Seabird, instruct them to keep firmware 1.0A and not the current revision of firmware. Beware that a calibration of a GPCTD can take up to 6 months so plan accordingly. After calibration and before reinstalling the instrument you will need to configure it using a terminal program. SeatermV2 is a terminal program distributed by Seabird Electronics that has the advantage of cycling through baud rates to connect. Otherwise it operates like other terminal (e.g. TeraTerm, Putty). To connect your GPCTD to a terminal program on a PC, you will need a cable (Seabird P/N 801944) and an appropriate power supply.

D.1 Connecting to your GPCTD with SeaTermV2

You can download SeaTermV2 software at the following url:
https://www.seabird.com/asset-get.download.jsa?code=251934.

After installing SeaTermV2, connect to the instrument referencing Section 3: Power and Communications Test of the GPCTD manual:
https://www.seabird.com/asset-get.download.jsa?id=54627862520.

D.2 Configuring your GPCTD for Communication with the SMC

Once connected, the instrument will respond with one of two prompts:
`<Executed/>` or `s>`

If the instrument responds with `<Executed/>` you need to execute
`<Executed/>OutputExecutedTag=N`

This will change the prompt from `<Executed/>` to `s>`. Next you will want to execute:
`S>DS`

This command outputs the current configuration of the instrument. Below is an example of the desired configuration for talking to the SMC. Highlighted are the parameters that you may need to change.

```
SBE Glider Payload CTD 1.0A SERIAL NO. 0065 24 Sep 2012 16:10:04
vMain = 11.66, vLith = 2.82
autorun = no
samplenumber = 0, free = 559240, profiles = 0
not logging, stop command
sample every 60 seconds
sample mode is interval, discontinuous pump
data format = HEX engineering
do not force on RS232 transmitter
transmit real time data
acquire SBE 43 oxygen
minimum conductivity frequency = 3000.0
custom pump mode enabled
```

First thing to confirm is the firmware version is 1.0A. If not, contact Liquid Robotics Technical Support. The next configuration parameter to address is the autorun setting. Confirm it is set to `no` as in the example above. If not, issue the following command:

```
S>AutoRun=N
```

Next, set the sampling interval to 60 seconds if no D.O. sensor is installed, and 10 seconds if a D.O. sensor is installed using one of the two following commands:
`S>Interval=60` or `S>Interval=10`

Next, set the output format to HEX engineering with the following command:
`S>OutputFormat=0`

Next, confirm the GPCTD is set to output real time data (the response to the DS command reads `transmit real time data`). If not, issue the following command:
`S>TxRealTime=Y`

Next, if you have a D.O. sensor installed confirm the GPCTD is set to acquire data from it (the response to the DS command reads `acquire SBE 43 oxygen`), if not, issue the following command:
`S>OxygenInstalled=Y`

If you do not have a D.O. sensor installed confirm the GPCTD is set to not acquire data from one (the response to the DS command reads `do not acquire SBE 43 oxygen`), if not, issue the following command:
`S>OxygenInstalled=N`

Lastly, confirm the DS statement reads `custom pump mode enabled`. At this point your GPCTD should be ready to communicate with the SMC. It is recommended that you save a copy of this configuration (output to the DS command) for troubleshooting should issues arise.

APPENDIX E: CONVERTING DISSOLVED OXYGEN FREQUENCY DATA TO ENGINEERING UNITS

Dissolved oxygen data from the GPCTD are in units of frequency. To convert these data into engineering units of milliliters/liter, reference page 4 in Seabird Application note 64.
https://www.seabird.com/asset-get.download.jsa?code=251036

This document gives an equation with multiple terms. You will need the latest calibration coefficients for your GPCTD to make the calculation. You can get the latest calibration coefficients for your instrument from Seabird Electronics technical support. Alternatively, you can use a `ctd --expert getCC` command described in Section C.1.2. Make note, the formula in Application not 64 references voltage while your dissolved oxygen sensor outputs frequency. Therefore, you will need to substitute the V and Voffset terms in the equation with F and Foffset where F is the reading from the instrument and Foffset is a constant from the calibration sheet.