Atlas Scientific™
Environmental Robotics

V 2.1

Revised 10/23

# EZO-PRS™

**Embedded Pressure Sensor**

| | |
|---|---|
| Reads | **Pressure (Gauge)** |
| | **psi (0 - 74.000) Default** |
| | **atm (0 - 5.03)** |
| | **bar (0 - 5.102)** |
| | **kPa (0 - 510.212)** |
| | **inches of water (0 - 2,050.36”)** |
| | **cm of water (0 - 5,202.71 cm)** |
| Response time | **1 reading per second** |
| Resolution | **0.001** |
| Accuracy | **+/- 1.35% (1psi)** |
| Threaded connection | **1/4" NPT** |
| Cable | **1 meter / 5 lead** |
| Data protocol | **UART & I²C** |
| Default I²C address | **106 (0x6A)** |
| Data format | **ASCII** |
| Operating voltage | **3.3V – 5V** |
| Water resistant/dust proof | **IP67** |


Written by Jordan Press
Designed by Noah Press

This is an evolving document, check back for updates.

# Table of contents

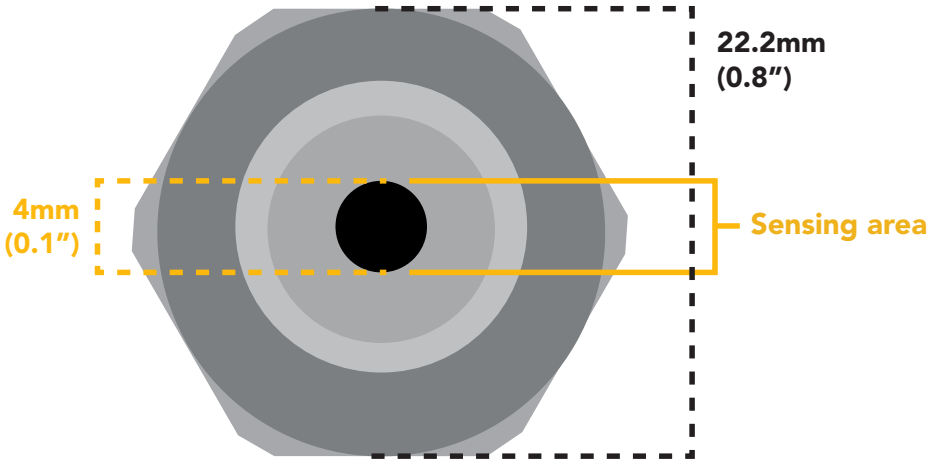


## UART

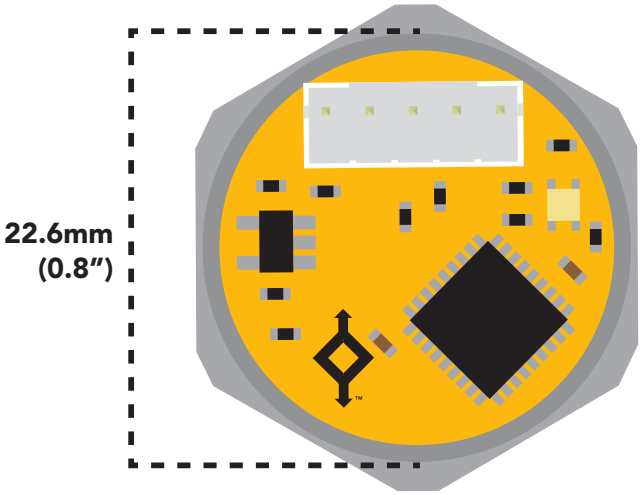


## I²C

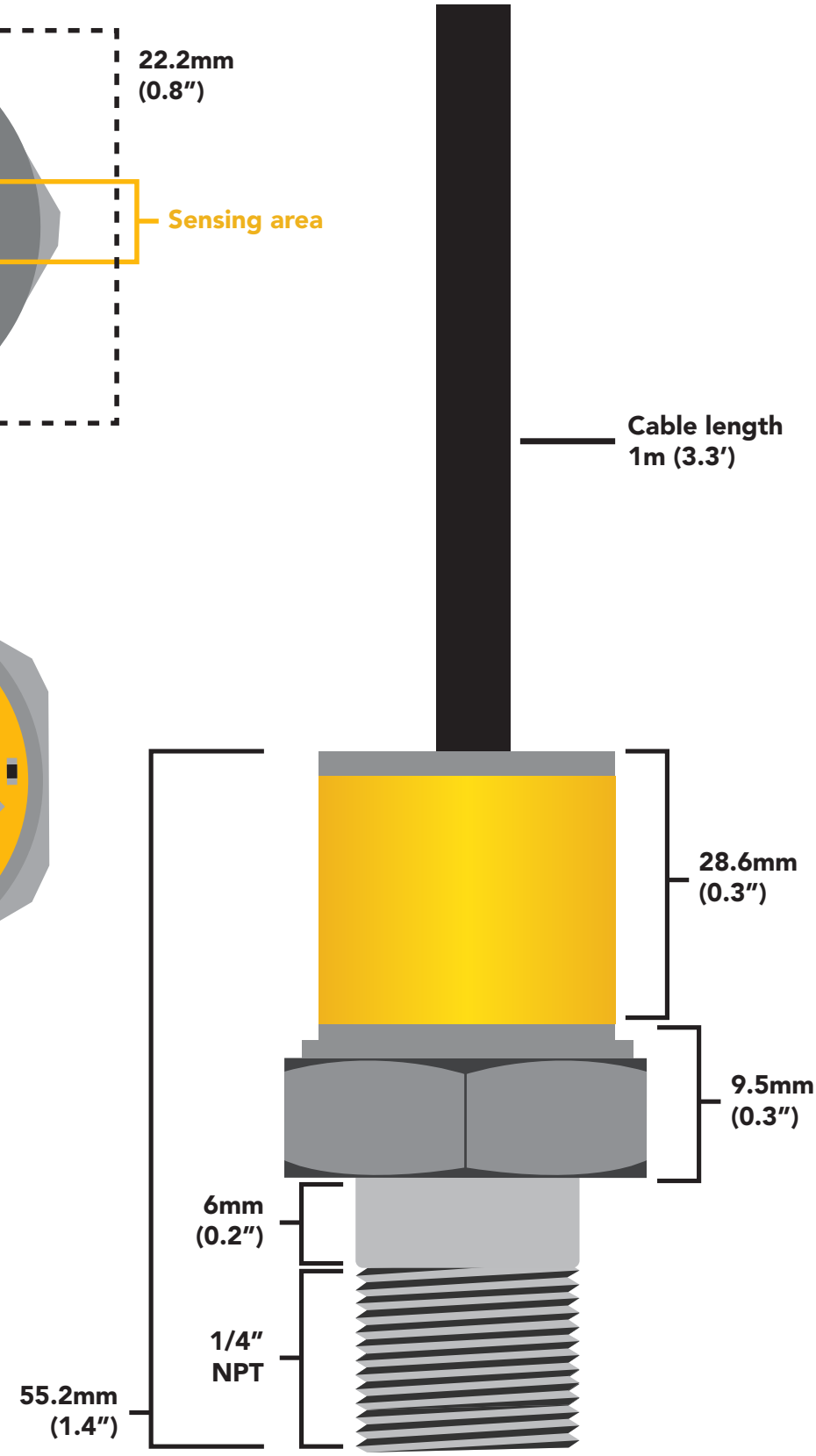
# EZO-PRS™ dimensions




Front




Back




**Weight** 108g

**Body** 304L Stainless Steel

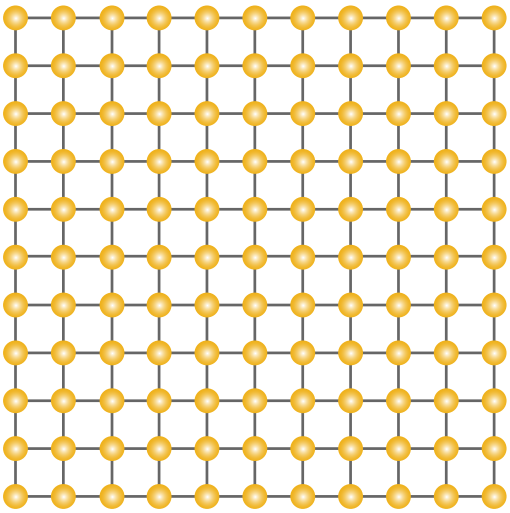
**Durability** IP67



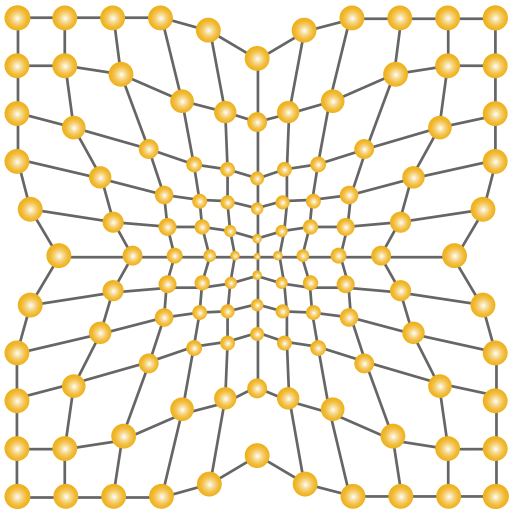

AtlasScientific™
Environmental Robotics

# Operating principle

Internally the pressure sensor uses a piezoresistive semiconducting element. The semiconducting element *(a silicon wafer)* changes its resistance in proportion to pressure. As the pressure increases the atomic spacing of the silicon atoms decreases, this in turn lowers the resistance of the silicon wafer.



**Atmospheric pressure**



**50 PSI**

## Chemical compatibly

Any gas, liquid or oil compatible with 304L Stainless Steel.

## Power consumption

| | LED | MAX | STANDBY | SLEEP |
|---|---|---|---|---|
| **5V** | ON | 14.25 mA | 14.25 mA | 1.66 mA |
| | OFF | 14.00 mA | 14.00 mA | |
| **3.3V** | ON | 13.21 mA | 13.21 mA | 0.85 mA |
| | OFF | 12.95 mA | 12.95 mA | |

## Absolute max ratings

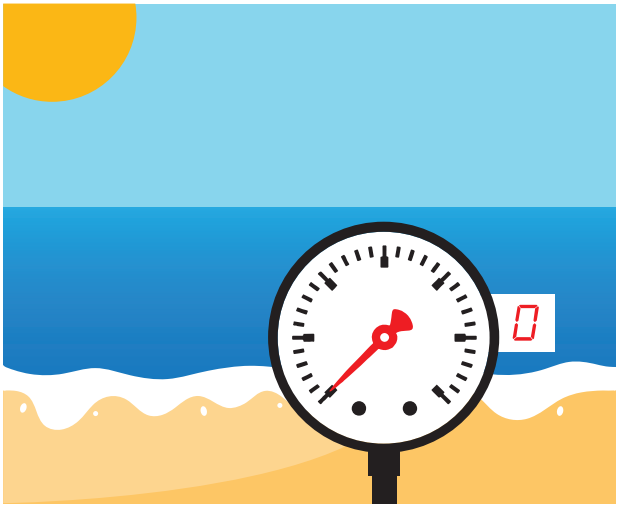
| Parameter | MIN | TYP | MAX |
|---|---|---|---|
| Storage temperature (EZO-Pressure) | -65 °C | | 125 °C |
| Operational temperature (EZO-Pressure) | -40 °C | 25 °C | 105 °C |
| VCC | 3.3V | 5V | 5.5V |
| Pressure limit (sensor damage) | | | ~150 psi |
| Burst Pressure | | | 7,500 psi |

# Gauge pressure vs Absolute pressure

The EZO-PRS™ reads ***gauge pressure*** only.

## Gauge pressure

A gauge pressure sensor reads pressure relative to atmospheric pressure.



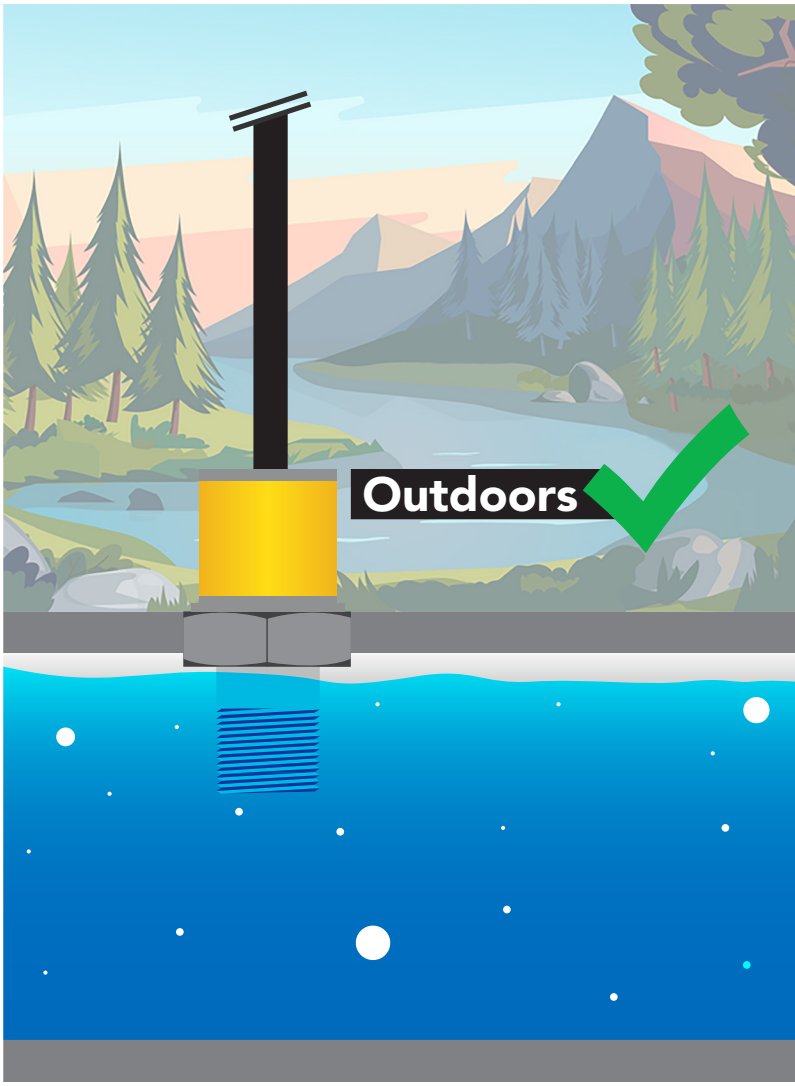


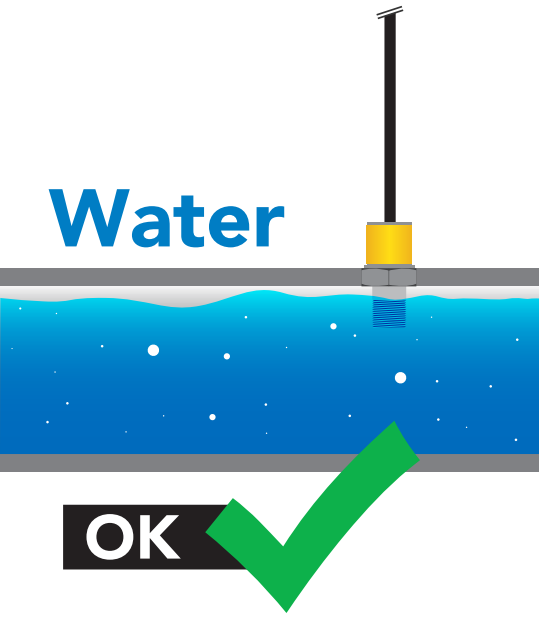
A gauge pressure sensor will always read atmospheric pressure as 0.

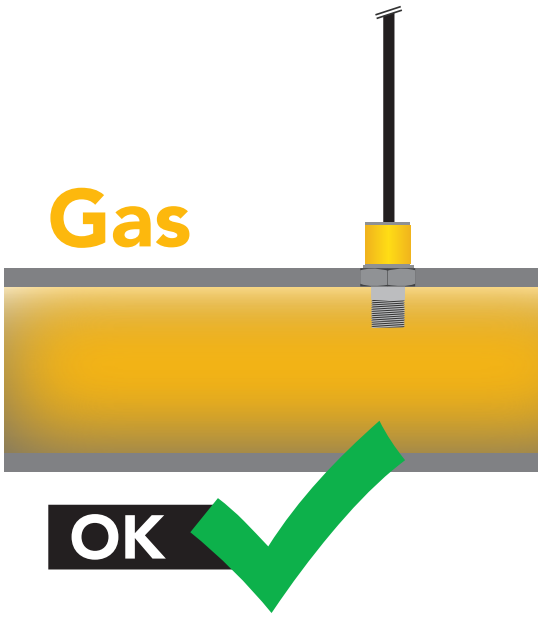
## Absolute pressure

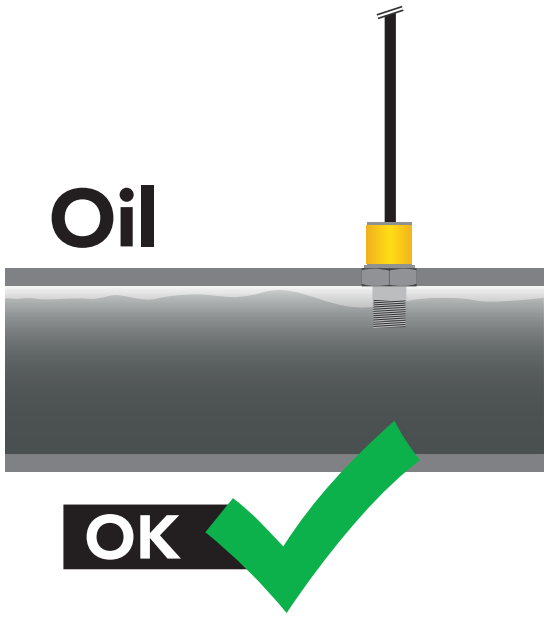
An absolute pressure sensor reads pressure relative to the vacuum of space.

# Typical applications

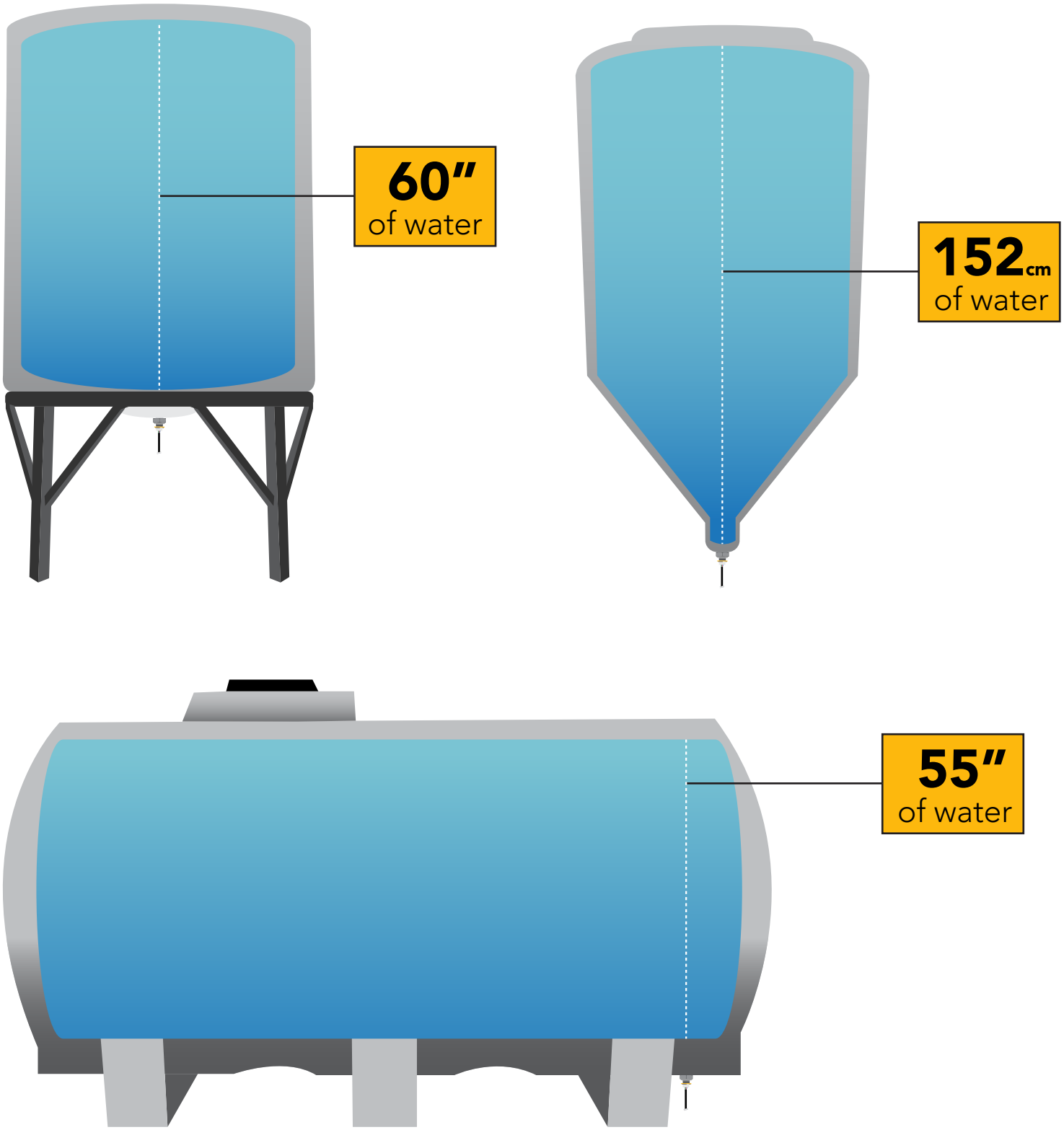



# Typical applications

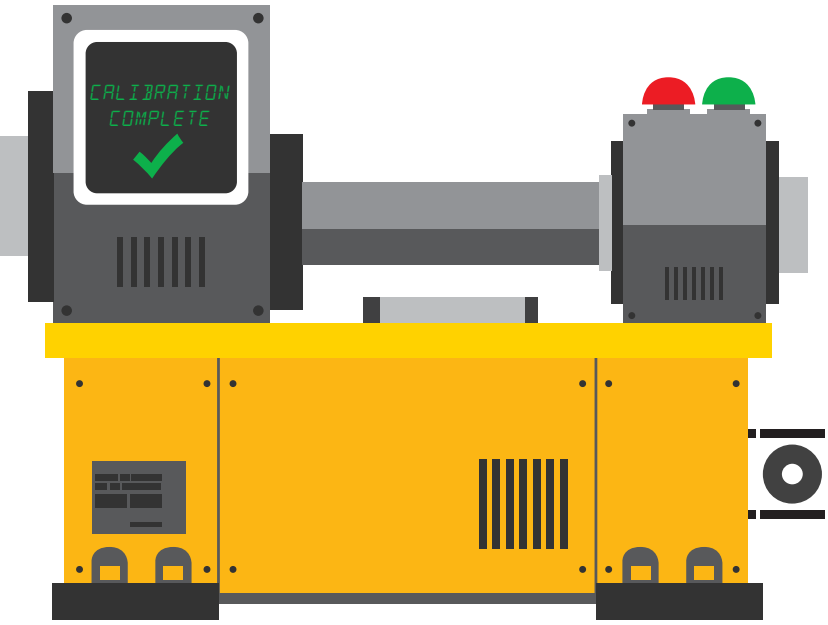
## Measuring the water level in a tank

Setting the EZO-PRS™ to measure the hight of water is a great way to measure the volume of a tank. See page **24** or **48** for more info.



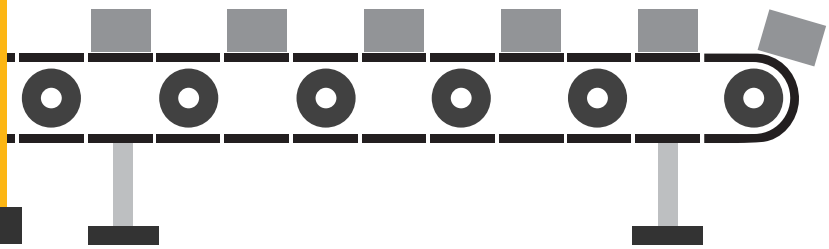
# Calibration theory

The Atlas Scientific EZO-PRS™ Embedded Pressure Sensor comes half-calibrated. When using the sensor for the first time, it is common to see a small pressure reading even though it is not connected to a pressure source. Issue the "Cal,0" command to complete the calibration process.




**Factory calibration points**

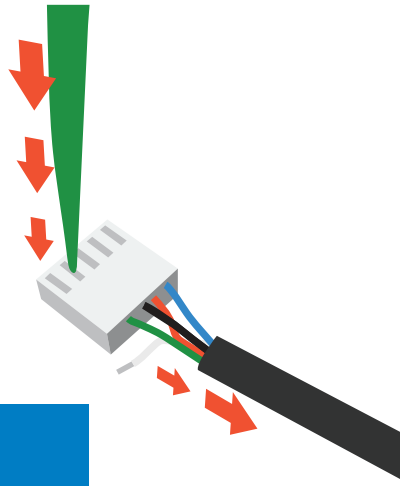
User calibration point = **0 psi**
High point calibration = **74 psi**



# Pin out

**Data and power cable pinout**

| | | |
|---|---|---|
| White | – | RX/SCL |
| Green | – | TX/SDA |
| Black | – | GND |
| Red | – | VCC |
| Blue | – | ALM |



Should you need to remove this connector from the data cable, follow the provided illustration.

The alarm pin will go high when a set pressure level has been crossed.


If unused leave **ALM** floating. Do not connect **ALM** to **VCC** or **GND**.

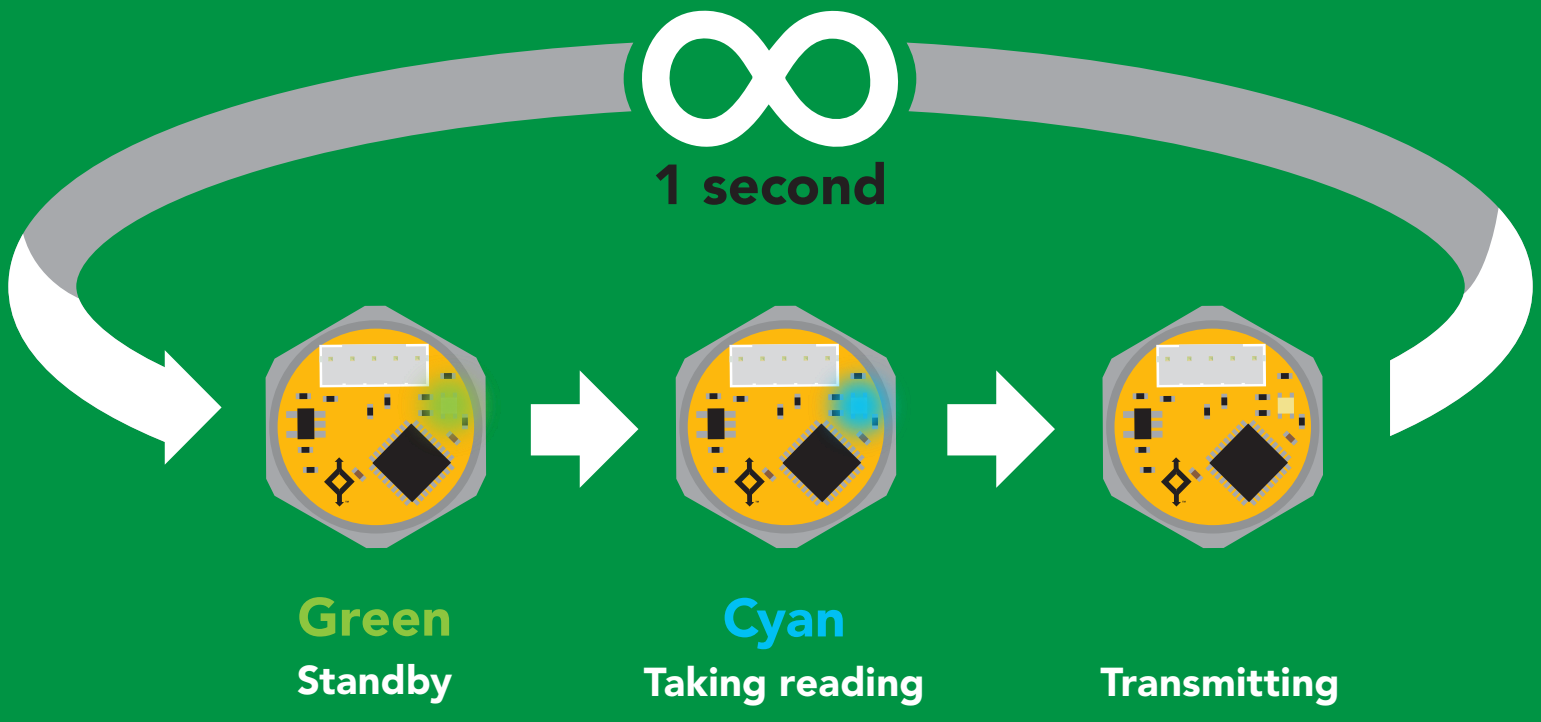
See page **21** or **45** to enable pressure level alarm.

Atlas Scientific™ Environmental Robotics

# Default state

# UART mode

| | |
|---|---|
| **Baud** | **9,600** |
| **Readings** | **continuous** |
| **Units** | **PSI** |
| **Speed** | **1 reading per second** |
| **LED** | **on** |



# ✓ Available data protocols

## UART

Default

## I²C

# X Unavailable data protocols

## SPI

## Analog

## RS-485

## Mod Bus

## 4–20mA




Atlas Scientific™ Environmental Robotics

# UART mode

**Settings that are retained if power is cut**

Baud rate
Calibration
Continuous mode
Custom calibration
Device name
Enable/disable response codes
Hardware switch to $I^2C$ mode
LED control
Protocol lock
Software switch to $I^2C$ mode

**Settings that are *NOT* retained if power is cut**

Find
Sleep mode

# UART mode

| | |
|---|---|
| 8 data bits | no parity |
| 1 stop bit | no flow control |

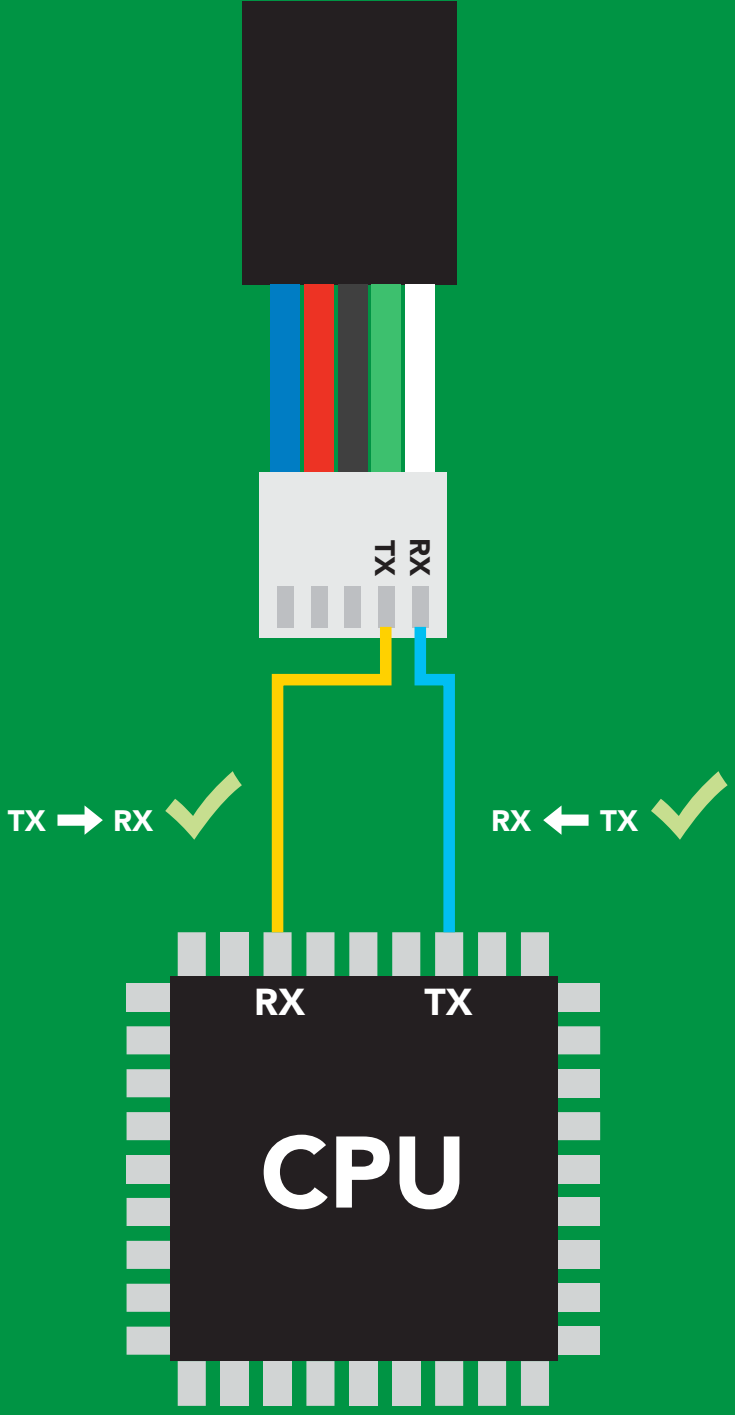
**Baud**

300
1,200
2,400
**9,600 default**
19,200
38,400
57,600
115,200

**RX**
Data in

**TX**
Data out

**Vcc** 3.3V – 5.5V


# Data format

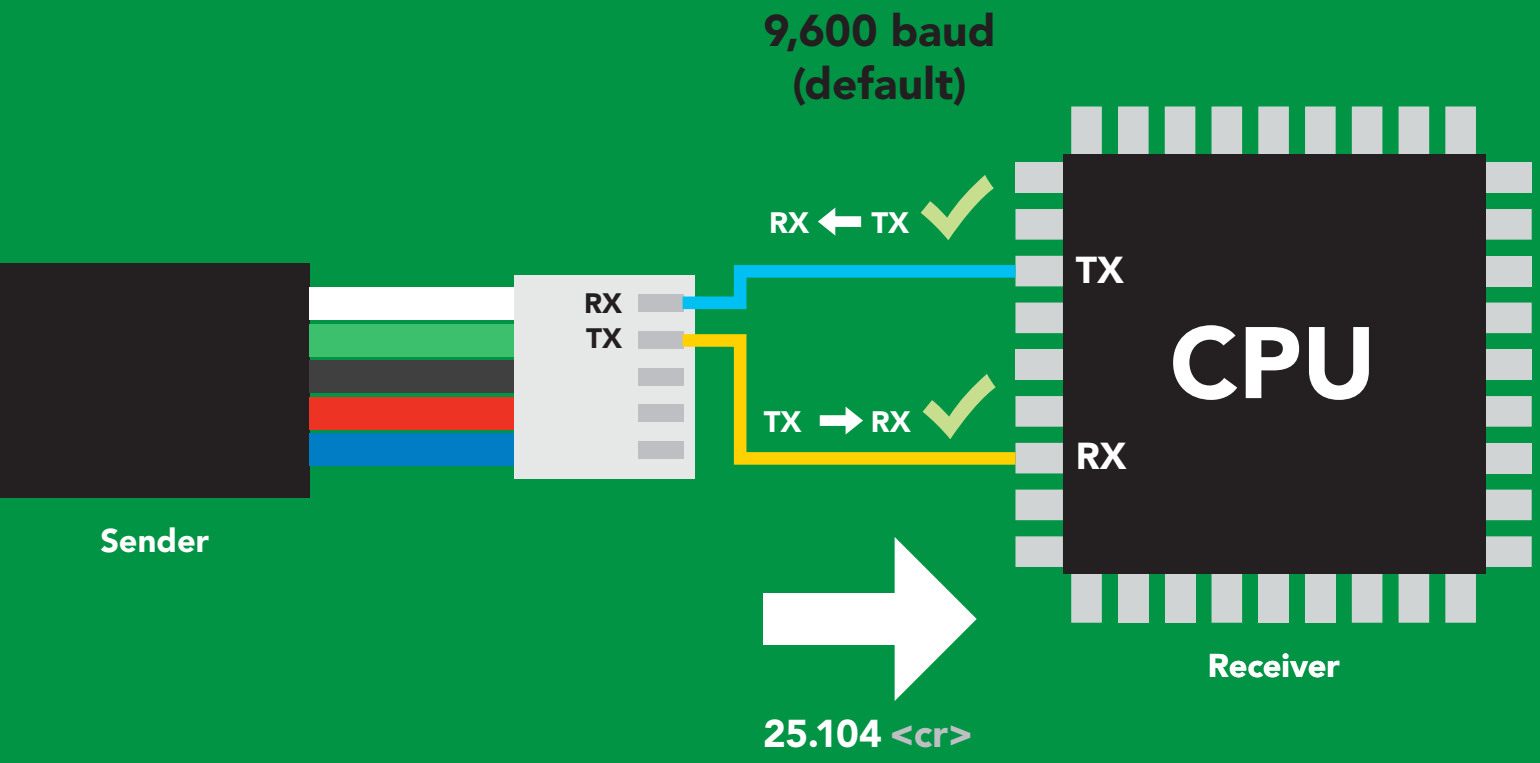
| | |
|---|---|
| **Output** | pressure |
| **Units** | PSI **default** (ATM, kPa, bar, inch's of water cm of water) |
| **Encoding** | ASCII |
| **Format** | string |
| **Terminator** | carriage return |
| **Data type** | floating point |
| **Decimal places** | 3 |
| **Smallest string** | 3 characters |
| **Largest string** | 39 characters |

# Receiving data from device

## 2 parts

**ASCII data string**
Command

**Carriage return <cr>**
Terminator




## Advanced

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ASCII: | 2 | 5 | . | 1 | 0 | 4 | <cr> |
| Hex: | 32 | 35 | 2E | 31 | 30 | 34 | 0D |
| Dec: | 50 | 53 | 46 | 49 | 48 | 52 | 13 |




Atlas Scientific™ Environmental Robotics
r 0.1

# Sending commands to device

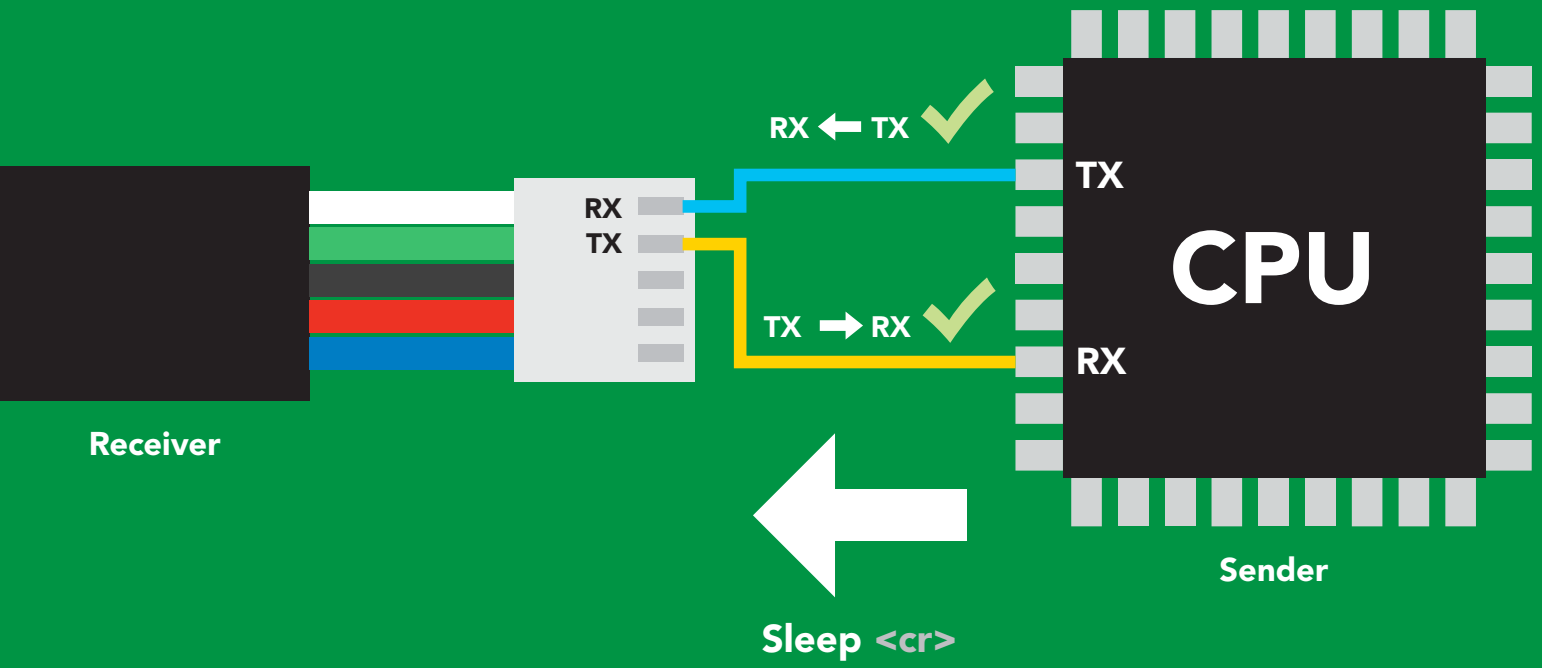
**2 parts**

**Command (not case sensitive)**

**ASCII data string**

**Carriage return <cr>**

**Terminator**



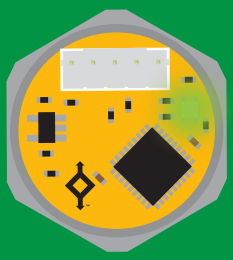

## Advanced

| | | | | | | |
|---|---|---|---|---|---|---|
| ASCII: | S | l | e | e | p | <cr> |
| Hex: | 53 | 6C | 65 | 65 | 70 | 0D |
| Dec: | 83 | 108 | 101 | 101 | 112 | 13 |

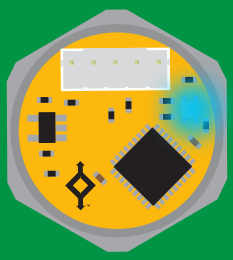



Atlas Scientific™ Environmental Robotics

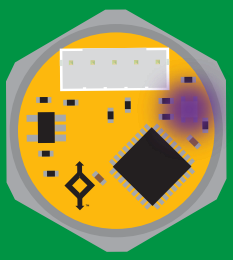
# LED color definition



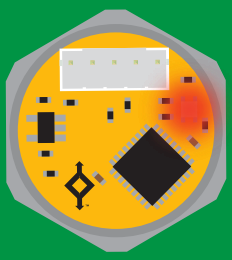
**Green**
**UART standby**



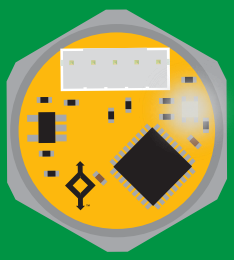
**Cyan**
**Taking reading**



**Purple**
**Changing baud rate**



**Red**
**Command not understood**



**White**
**Find**

| | |
|---|---|
| **5V** | LED ON **+.25 mA** |
| **3.3V** | **+.26 mA** |

# UART mode
## command quick reference

**All commands are ASCII strings or single ASCII characters.**

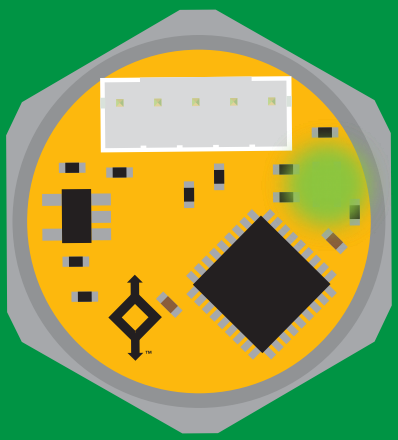
| Command | Function | | Default state |
|---|---|---|---|
| Alarm | enable/disable alarm | pg. 21 | n/a |
| Baud | change baud rate | pg. 30 | 9,600 |
| C | enable/disable continuous mode | pg. 19 | enabled |
| Cal | performs custom calibration | pg. 22 | n/a |
| Dec | add/remove decimal places | pg. 23 | n/a |
| Factory | enable factory reset | pg. 32 | n/a |
| Find | finds device with blinking white LED | pg. 18 | n/a |
| i | device information | pg. 26 | n/a |
| I2C | change to I²C mode | pg. 33 | not set |
| L | enable/disable LED | pg. 17 | enabled |
| Name | set/show name of device | pg. 25 | not set |
| Plock | enable/disable protocol lock | pg. 31 | disabled |
| R | returns a single reading | pg. 20 | n/a |
| Sleep | enter sleep mode/low power | pg. 29 | n/a |
| Status | retrieve status information | pg. 28 | enable |
| U | pressure units | pg. 24 | psi |
| *OK | enable/disable response codes | pg. 27 | enable |

# LED control

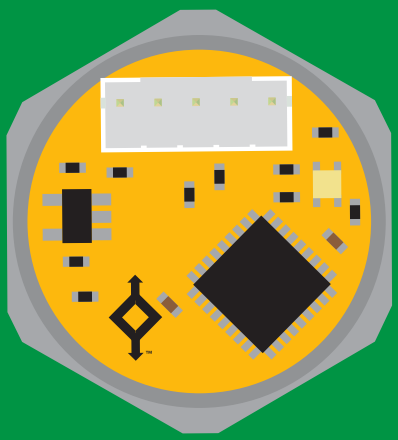
## Command syntax

L,1 <cr> LED on **default**
L,0 <cr> LED off
L,? <cr> LED state on/off?

| Example | Response |
|---|---|
| L,1 <cr> | *OK <cr> |
| L,0 <cr> | *OK <cr> |
| L,? <cr> | ?L,1 <cr> or ?L,0 <cr><br>*OK <cr> |



L,1



L,0

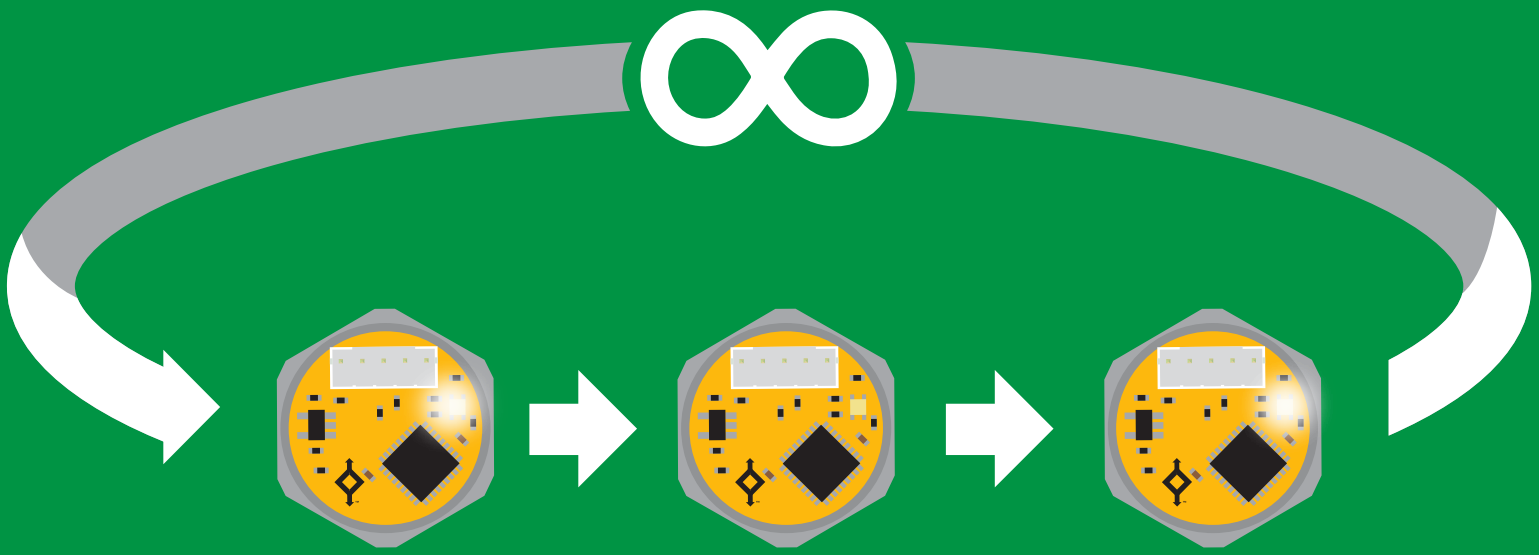
# Find

## Command syntax

**This command will disable continuous mode**
**Send any character or command to terminate find.**

**Find** <cr> **LED rapidly blinks white, used to help find device**

| Example | Response |
| --- | --- |
| **Find** <cr> | ***OK** <cr> |



# Continuous mode

## Command syntax

C,1 <cr> enable continuous readings once per second **default**

C,n <cr> continuous readings every n seconds (n = 2 to 99 sec)

C,0 <cr> disable continuous readings

C,? <cr> continuous reading mode on/off?

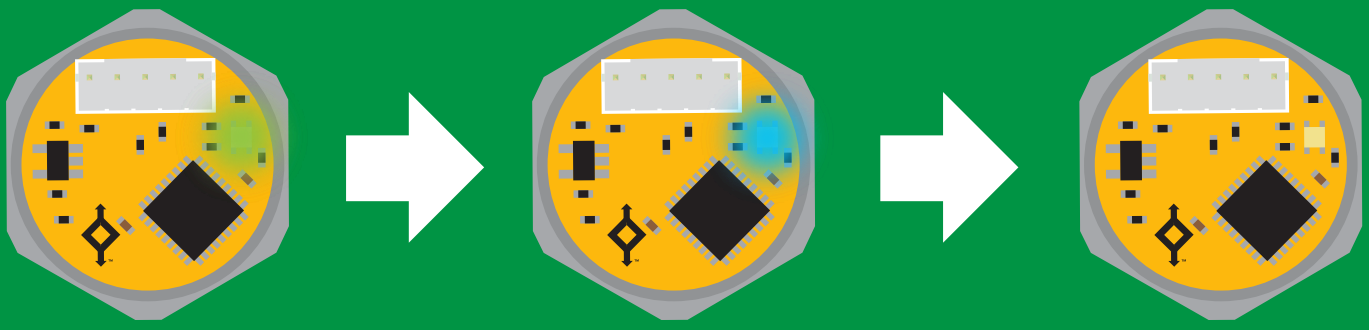
| Example | Response |
|---|---|
| C,1 <cr> | *OK <cr><br>pressure (1 sec) <cr><br>pressure (2 sec) <cr><br>pressure (n sec) <cr> |
| C,30 <cr> | *OK <cr><br>pressure (30 sec) <cr><br>pressure (60 sec) <cr><br>pressure (90 sec) <cr> |
| C,0 <cr> | *OK <cr> |
| C,? <cr> | ?C,1 <cr> or ?C,0 <cr> or ?C,30 <cr><br>*OK <cr> |

# Single reading mode

## Command syntax

R <cr> takes single reading

| Example | Response |
| --- | --- |
| R <cr> | 38.462 <cr><br>*OK <cr> |



**Green**
Standby

**Cyan**
Taking reading

Transmitting

1 second




Atlas Scientific™ Environmental Robotics

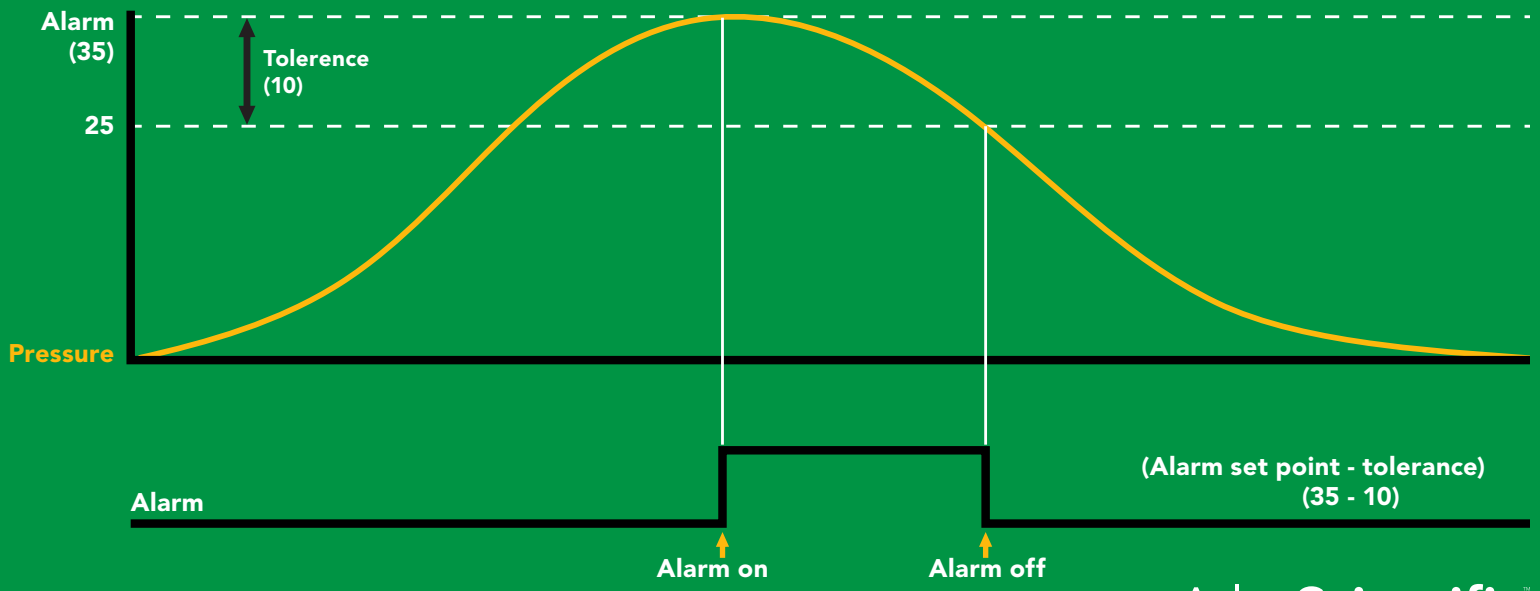
# Alarm

## Command syntax

The alarm pin will = 1 when pressure levels are > alarm set point. Alarm tolerance sets how far below the set point pressure levels need to drop before the pin will = 0 again.

| | | |
|---|---|---|
| Alarm,en,[1,0] | <cr> | enable / disable alarm |
| Alarm,n | <cr> | sets alarm |
| Alarm,tol,n | <cr> | sets alarm tolerance |
| Alarm,? | <cr> | alarm set? |

| Example | Response |
|---|---|
| Alarm,en,1 <cr> | *OK <cr> Enable alarm |
| Alarm,35 <cr> | *OK <cr> |
| Alarm,tol,10 <cr> | *OK <cr> Pressure level must fall 10 units below set point for alarm to reset. |
| Alarm,? <cr> | ?,alarm,35,10,1 <cr> if all are enabled |







AtlasScientific™
Environmental Robotics

# Custom calibration

## Command syntax

**Although calibration is not required, it may be necessary to adjust your 0 point or perform a custom calibration.**

| | | |
|---|---|---|
| Cal,n | <cr> | calibrates the high point |
| Cal,0 | <cr> | calibrates the zero point |
| Cal,clear | <cr> | restores calibration to factory settings |
| Cal,? | <cr> | device calibrated? |

| Example | Response |
|---|---|
| Cal,50 <cr> | *OK <cr> high point calibration in psi |
| Cal,0 <cr> | *OK <cr> low point calibration in psi |
| Cal,clear <cr> | *OK <cr> |
| Cal,? <cr> | ?Cal,0 <cr> (no calibration) or ?Cal,1 <cr> (only zero point calibration) or ?Cal,2 <cr> (only high point calibration) or ?Cal,3 <cr> (zero and high point calibration) *OK <cr> |

**Calibration should be done using the pressure scale you have set the sensor to.**

**Example**

**Readings are set to bar.**
**High point calibration = 3.44**
**(3.44 bar = 50 psi)**




AtlasScientific™ Environmental Robotics

# Add/remove decimal places

## Command syntax

**Change how many decimal points the reading outputs.**

| | | |
|---|---|---|
| Dec,n | <cr> | n = number of decimal points between 0 and 3 |
| Dec,? | <cr> | number of decimal points the output is set to |

| Example | Response |
|---|---|
| R <cr> | 38.462 <cr> |
| Dec,1 <cr> | *OK <cr> |
| R <cr> | 38.4 <cr> |
| Dec,? <cr> | ?Dec,1 <cr> |

# Pressure units

(psi, atm, bar, kPa, inch's of water, cm of water)

## Command syntax

| | | | |
|---|---|---|---|
| U,[1/0] | <cr> | 1 will add a unit identifier to the output | |
| U,psi | <cr> | output will be in psi **default** | |
| U,atm | <cr> | output will be in atm | |
| U,bar | <cr> | output will be in bar | |
| U,kPa | <cr> | output will be in kPa | |
| U,inh2o | <cr> | output will be in inches of water | (Resolution: 0.027") |
| U,cmh2o | <cr> | output will be in cm of water | (Resolution: 0.7mm) |
| U,? | <cr> | pressure units? | |

| Example | Response |
|---|---|
| U,bar <cr> | *OK <cr> |
| U,1 <cr> | *OK <cr><br>1.228,bar <cr> |
| U,? <cr> | ?U,bar <cr> |




Atlas Scientific™ Environmental Robotics

# Naming device

## Command syntax

Do not use spaces in the name

| | | |
|---|---|---|
| Name,n | <cr> | set name |
| Name, | <cr> | clears name |
| Name,? | <cr> | show name |

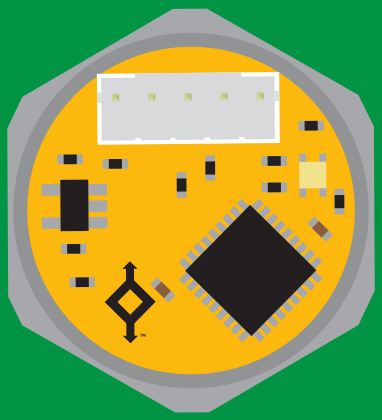
n =

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

Up to 16 ASCII characters

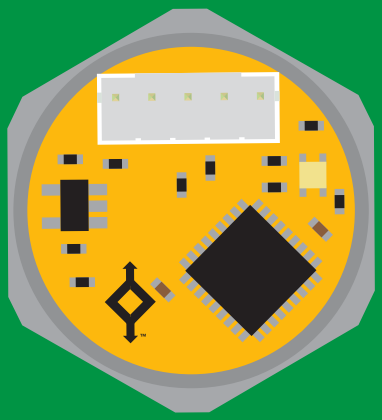
| Example | Response |
|---|---|
| Name, <cr> | *OK <cr> name has been cleared |
| Name,zzt <cr> | *OK <cr> |
| Name,? <cr> | ?Name,zzt <cr><br>*OK <cr> |

Name,zzt



*OK <cr>

Name,?



?Name,zzt <cr>
*OK <cr>




AtlasScientific™ Environmental Robotics

# Device information

## Command syntax

**i** <cr> **device information**

| Example | Response |
| --- | --- |
| **i** <cr> | **?i,PRS,1.0** <cr><br>***OK** <cr> |

## Response breakdown

| ?i, | PRS | 1.0 |
| --- | --- | --- |
| | ↑ Device | ↑ Firmware |

# Response codes

## Command syntax

*OK,1 <cr> enable response **default**
*OK,0 <cr> disable response
*OK,? <cr> response on/off?

| Example | Response |
|---|---|
| R <cr> | 38.462 <cr><br>*OK <cr> |
| *OK,0 <cr> | no response, *OK disabled |
| R <cr> | 38.462 <cr> *OK disabled |
| *OK,? <cr> | ?*OK,1 <cr> or ?*OK,0 <cr> |

### Other response codes

| | |
|---|---|
| *ER | unknown command |
| *OV | over volt (VCC>=5.5V) |
| *UV | under volt (VCC<=3.1V) |
| *RS | reset |
| *RE | boot up complete, ready |
| *SL | entering sleep mode |
| *WA | wake up |

**These response codes cannot be disabled**

# Reading device status

## Command syntax

**Status** <cr> **voltage at Vcc pin and reason for last restart**

| Example | Response |
|---|---|
| **Status** <cr> | **?Status,P,5.038** <cr><br>***OK** <cr> |

## Response breakdown

**?Status, P, 5.038**

- **P** — Reason for restart
- **5.038** — Voltage at Vcc

### Restart codes

| | |
|---|---|
| P | powered off |
| S | software reset |
| B | brown out |
| W | watchdog |
| U | unknown |

# Sleep mode/low power

## Command syntax

**Send any character or command to awaken device.**

**Sleep <cr>** enter sleep mode/low power

| Example | Response |
|---|---|
| Sleep <cr> | *OK <cr><br>*SL <cr> |
| Any command | *WA <cr> wakes up device |

| | STANDBY | SLEEP |
|---|---|---|
| 5V | 13.4 mA | 0.415 mA |
| 3.3V | 12.4 mA | 0.13 mA |


Standby
13.4 mA

Sleep
0.415 mA




AtlasScientific™
Environmental Robotics

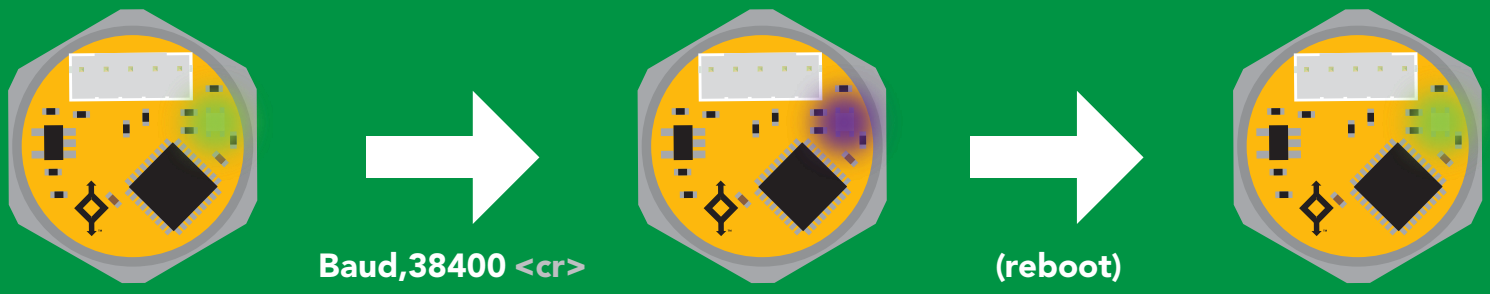
# Change baud rate

## Command syntax

**Baud,n <cr> change baud rate**

| Example | Response |
|---|---|
| Baud,38400 <cr> | *OK <cr> |
| Baud,? <cr> | ?Baud,38400 <cr><br>*OK <cr> |

n =
- 300
- 1200
- 2400
- **9600 default**
- 19200
- 38400
- 57600
- 115200




Standby

Changing baud rate

*OK <cr>

Standby




AtlasScientific™ Environmental Robotics

# Protocol lock

## Command syntax

Locks device to UART mode.

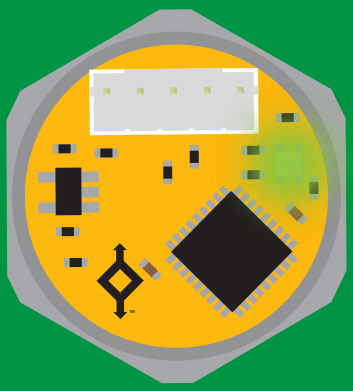
Plock,1 <cr> enable Plock

Plock,0 <cr> disable Plock **default**

Plock,? <cr> Plock on/off?

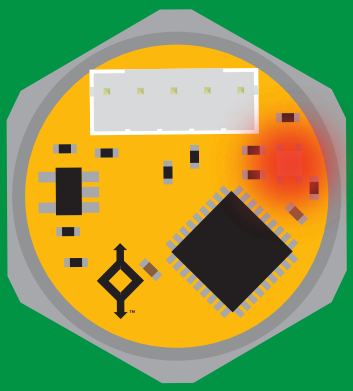
| Example | Response |
|---|---|
| Plock,1 <cr> | *OK <cr> |
| Plock,0 <cr> | *OK <cr> |
| Plock,? <cr> | ?Plock,1 <cr> or ?Plock,0 <cr> |

Plock,1



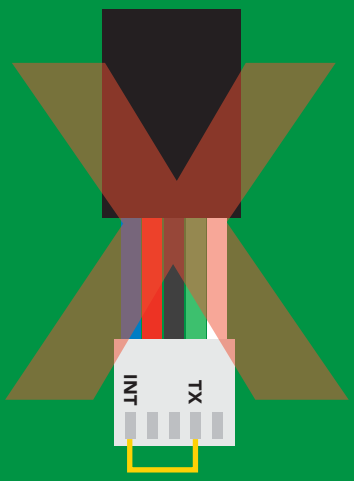
*OK <cr>

I2C,100



cannot change to I²C

*ER <cr>




cannot change to I²C




AtlasScientific™ Environmental Robotics

# Factory reset

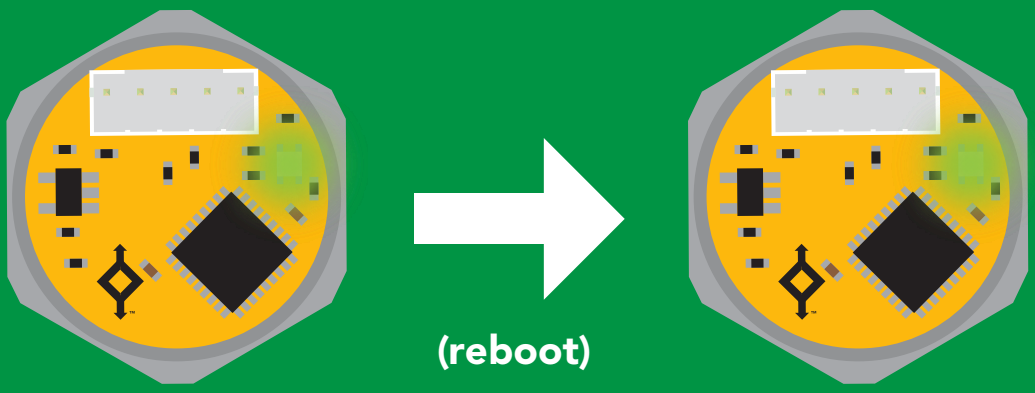
## Command syntax

Clears calibration LED on "*OK" enabled

**Factory** <cr> enable factory reset

| Example | Response |
|---|---|
| **Factory** <cr> | ***OK** <cr> |

**Factory** <cr>



(reboot)

***OK** <cr>

***RS** <cr>
***RE** <cr>

**Baud rate will not change**




Atlas Scientific Environmental Robotics

# Change to I²C mode

## Command syntax

Default I²C address 106 (0x6A)

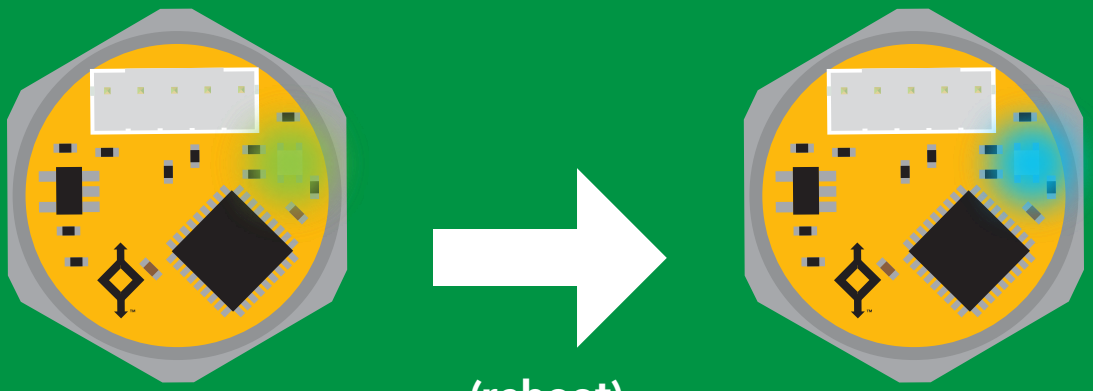
I2C,n <cr> sets I²C address and reboots into I²C mode

n = any number 1 – 127

| Example | Response |
|---|---|
| I2C,100 <cr> | *OK (reboot in I²C mode) |

| Wrong example | Response |
|---|---|
| I2C,139 <cr> n ≯ 127 | *ER <cr> |

I2C,100



(reboot)

**Green**
*OK <cr>

**Blue**
now in I²C mode




AtlasScientific™ Environmental Robotics

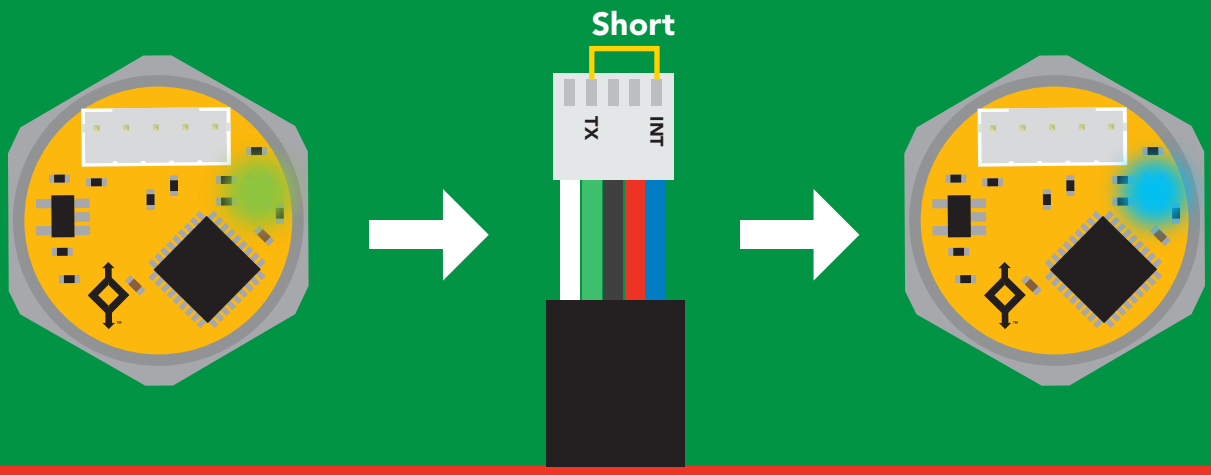
# Manual switching to I²C

- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to INT
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from
- Green to Blue
- Disconnect ground (power off)

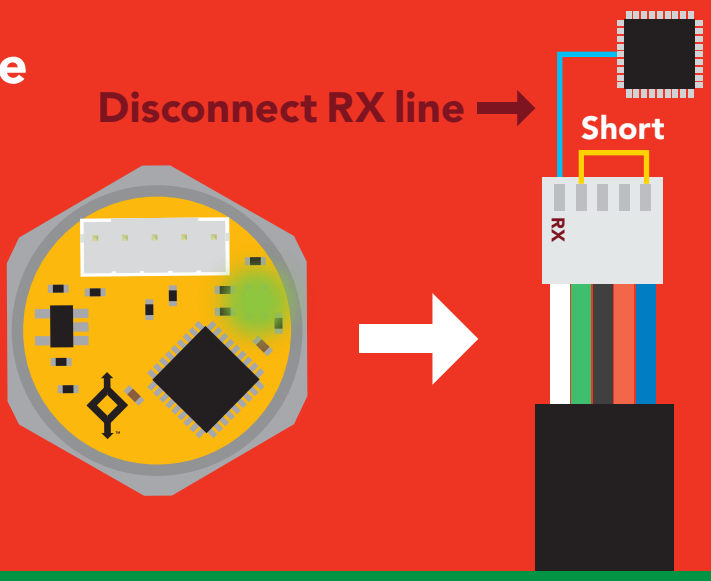
Reconnect all data and power

**Manually switching to I²C will set the I²C address to 106 (0x6A)**

## Example




## Wrong Example







AtlasScientific™ Environmental Robotics

# I²C mode

**The I²C protocol is *considerably more complex* than the UART (RS–232) protocol. Atlas Scientific assumes the embedded systems engineer understands this protocol.**

**To set your EZO-PRS™ into I²C mode click here**

**Settings that are retained if power is cut**

Calibration
Change I²C address
Custom calibration
Hardware switch to UART mode
LED control
Protocol lock
Software switch to UART mode

**Settings that are *NOT* retained if power is cut**

Find
Sleep mode

AtlasScientific™
Environmental Robotics

# I²C mode

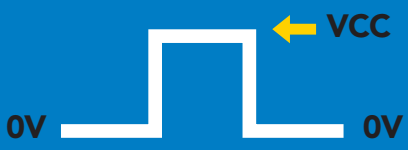
**I²C address** (0x01 – 0x7F)
**106 (0x6A) default**

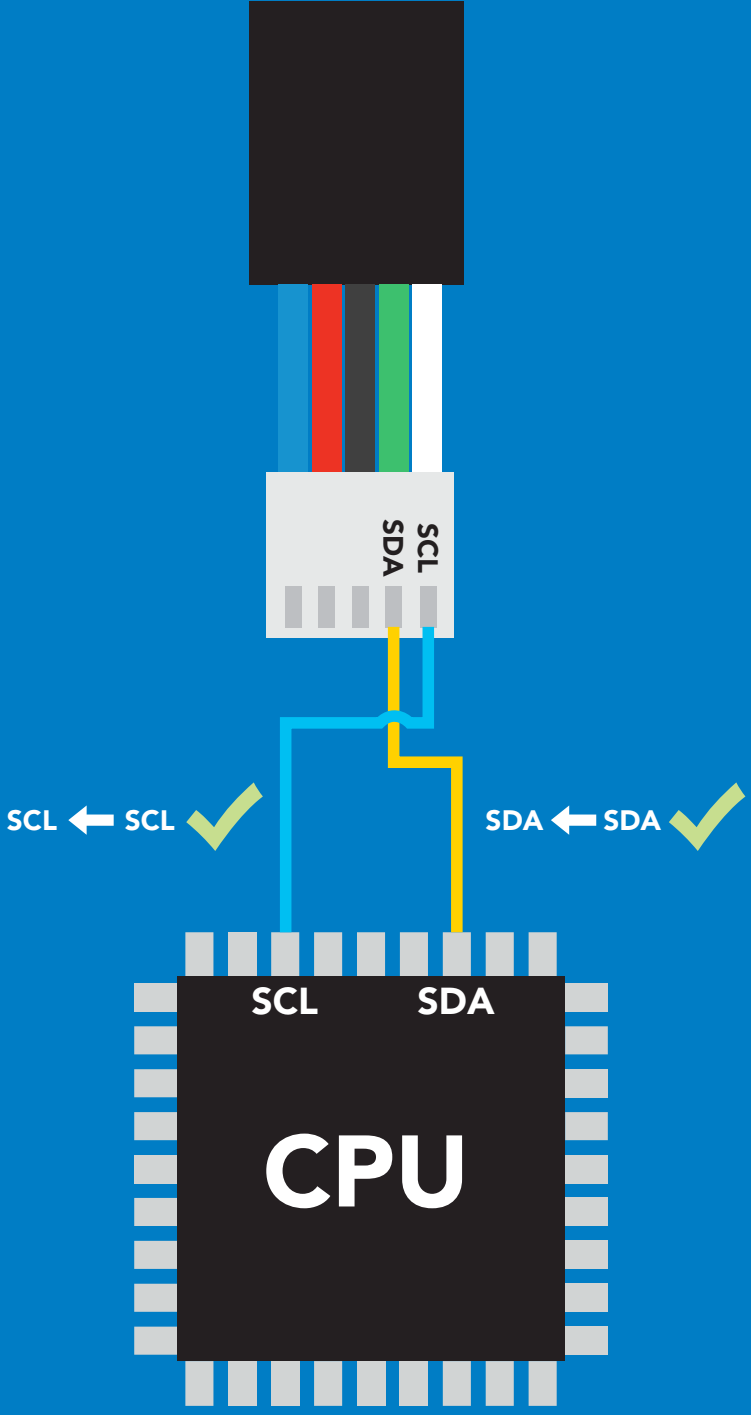
**Vcc** 3.3V – 5.5V

**Clock speed** 100 – 400 kHz

**SDA**

**SCL**



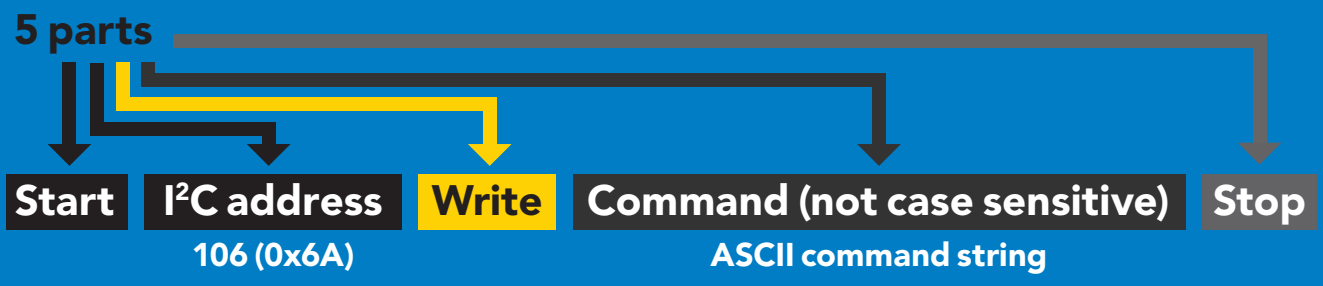




# Data format

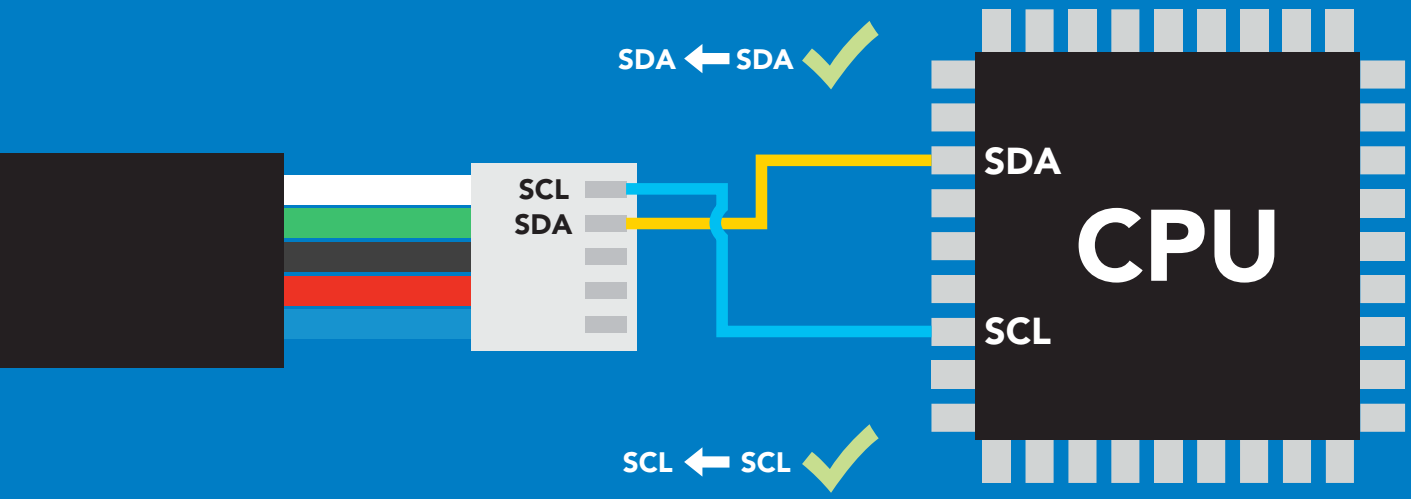
**Output** pressure
**Units** PSI **default**
(ATM, kPa, bar, inch's of water cm of water)
**Encoding** ASCII
**Format** string

**Data type** floating point
**Decimal places** 3
**Smallest string** 3 characters
**Largest string** 39 characters

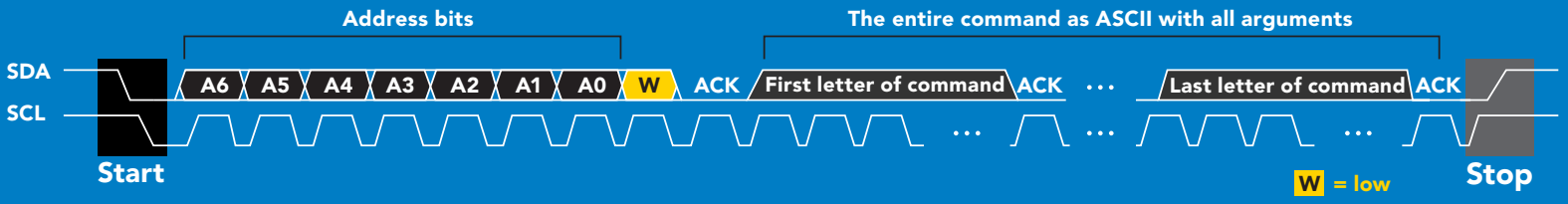
# Sending commands to device




## Example


## Advanced

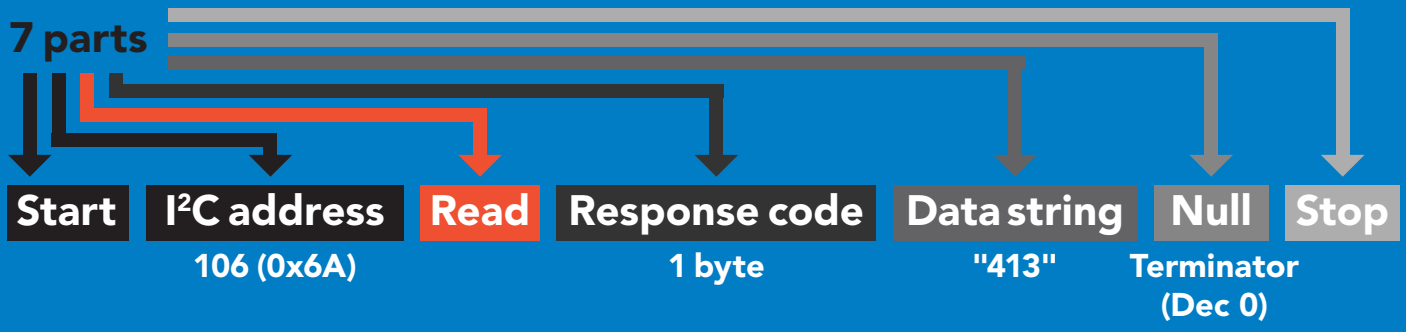


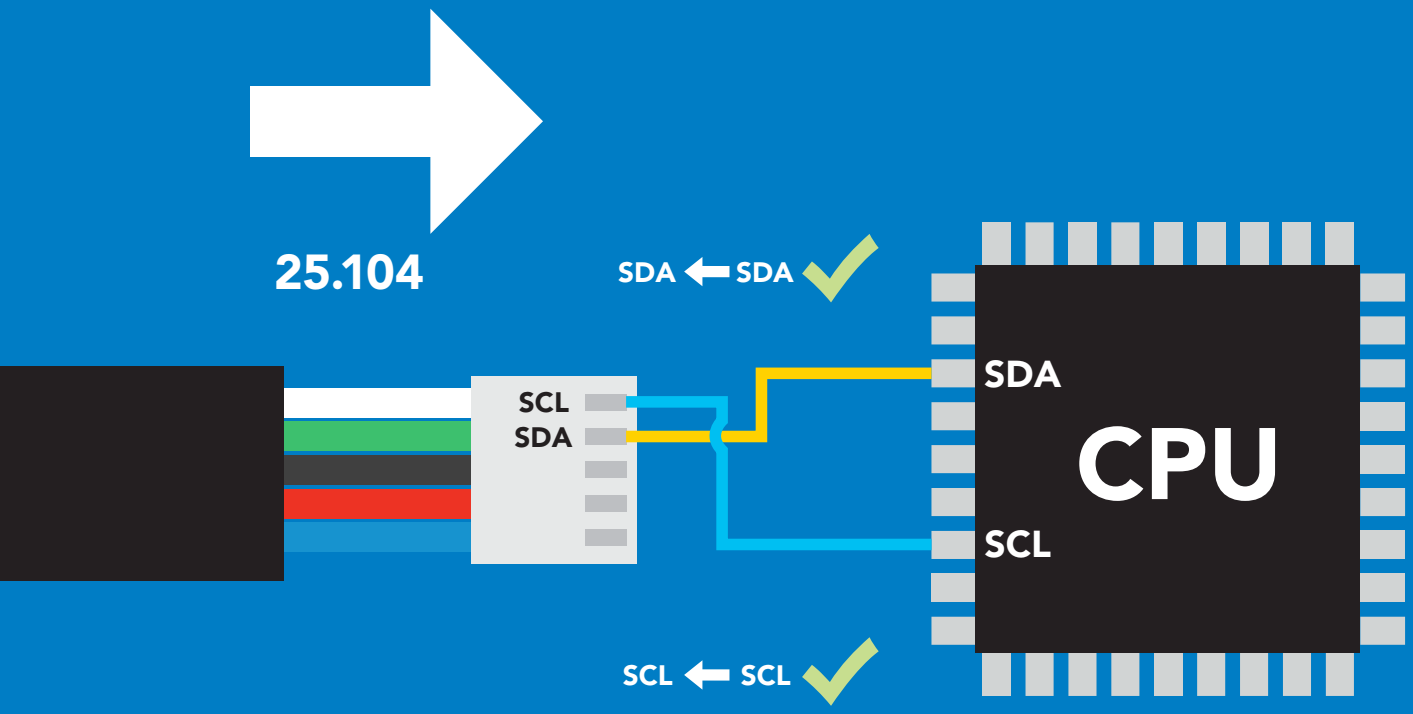




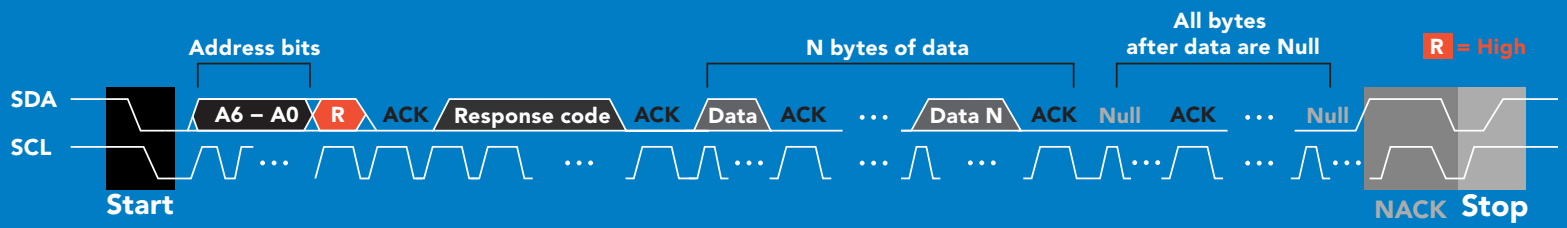
AtlasScientific™
Environmental Robotics

# Requesting data from device




## Advanced





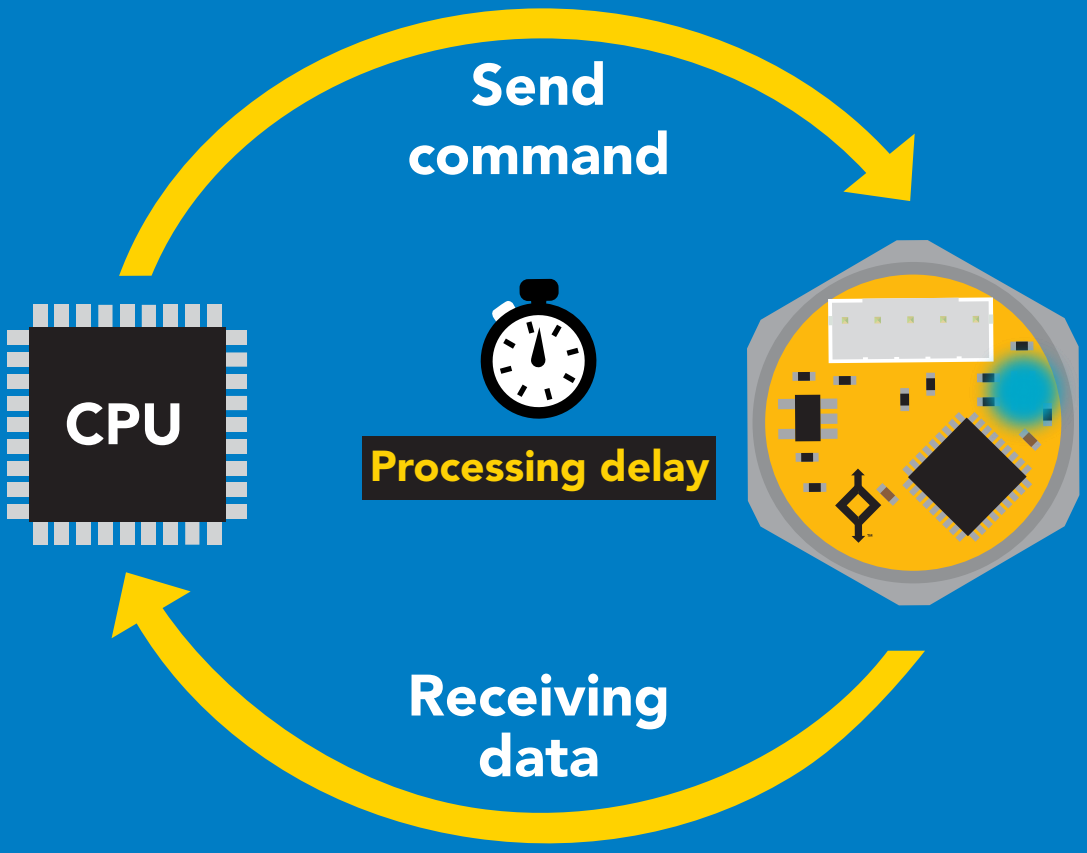



Atlas Scientific™ Environmental Robotics

# Response codes

**After a command has been issued, a 1 byte response code can be read in order to confirm that the command was processed successfully.**

***Reading back the response code is completely optional, and is not required for normal operation.***




## Example

```
I2C_start;
I2C_address;
I2C_write(EZO_command);
I2C_stop;

delay(300);   Processing delay

I2C_start;
I2C_address;
Char[ ] = I2C_read;
I2C_stop;
```

**If there is no processing delay or the processing delay is too short, the response code will always be 254.**

**Response codes**

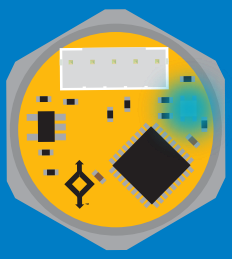
**Single byte, not string**

| | |
|---|---|
| 255 | no data to send |
| 254 | still processing, not ready |
| 2 | syntax error |
| 1 | successful request |

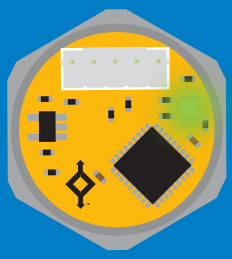



Atlas Scientific Environmental Robotics

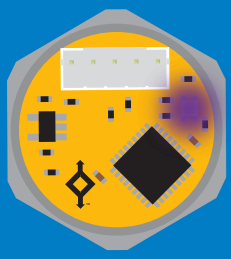
# LED color definition



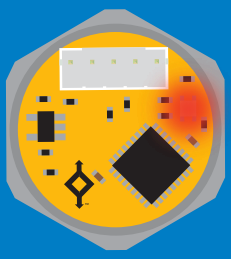
**Blue**
**I²C standby**



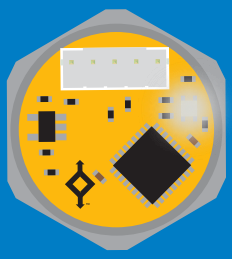
**Green**
**Taking reading**



**Purple**
**Changing I²C address**



**Red**
**Command not understood**



**White**
**Find**

| | LED ON |
|---|---|
| **5V** | **+.25 mA** |
| **3.3V** | **+.26 mA** |

# I²C mode
## command quick reference

**All commands are ASCII strings or single ASCII characters.**

| Command | Function | |
|---|---|---|
| Alarm | enable/disable alarm | pg. 45 |
| Baud | switch back to UART mode | pg. 56 |
| Cal | performs custom calibration | pg. 46 |
| Dec | add/remove decimal places | pg. 47 |
| Factory | enable factory reset | pg. 55 |
| Find | finds device with blinking white LED | pg. 43 |
| i | device information | pg. 50 |
| I2C | change I²C address | pg. 54 |
| L | enable/disable LED | pg. 42 |
| Name | set/show name of device | pg. 49 |
| Plock | enable/disable protocol lock | pg. 53 |
| R | returns a single reading | pg. 44 |
| Sleep | enter sleep mode/low power | pg. 52 |
| Status | retrieve status information | pg. 51 |
| U | pressure units | pg. 48 |

# LED control

## Command syntax

300ms processing delay

| | | |
|---|---|---|
| L,1 | LED on | default |
| L,0 | LED off | |
| L,? | LED state on/off? | |

## Example Response

**L,1**

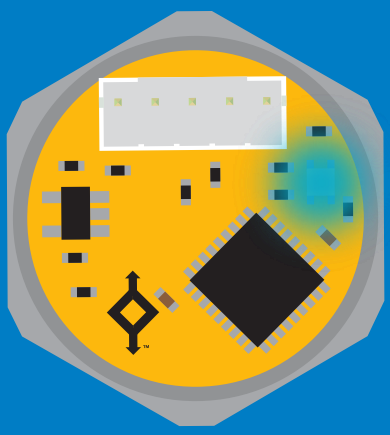
Wait 300ms → 1 (Dec) 0 (Null)

**L,0**

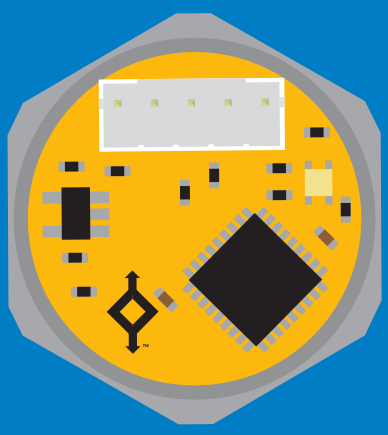
Wait 300ms → 1 (Dec) 0 (Null)

**L,?**

Wait 300ms → 1 (Dec) ?L,1 (ASCII) 0 (Null) or Wait 300ms → 1 (Dec) ?L,0 (ASCII) 0 (Null)



L,1



L,0




AtlasScientific™ Environmental Robotics

# Find

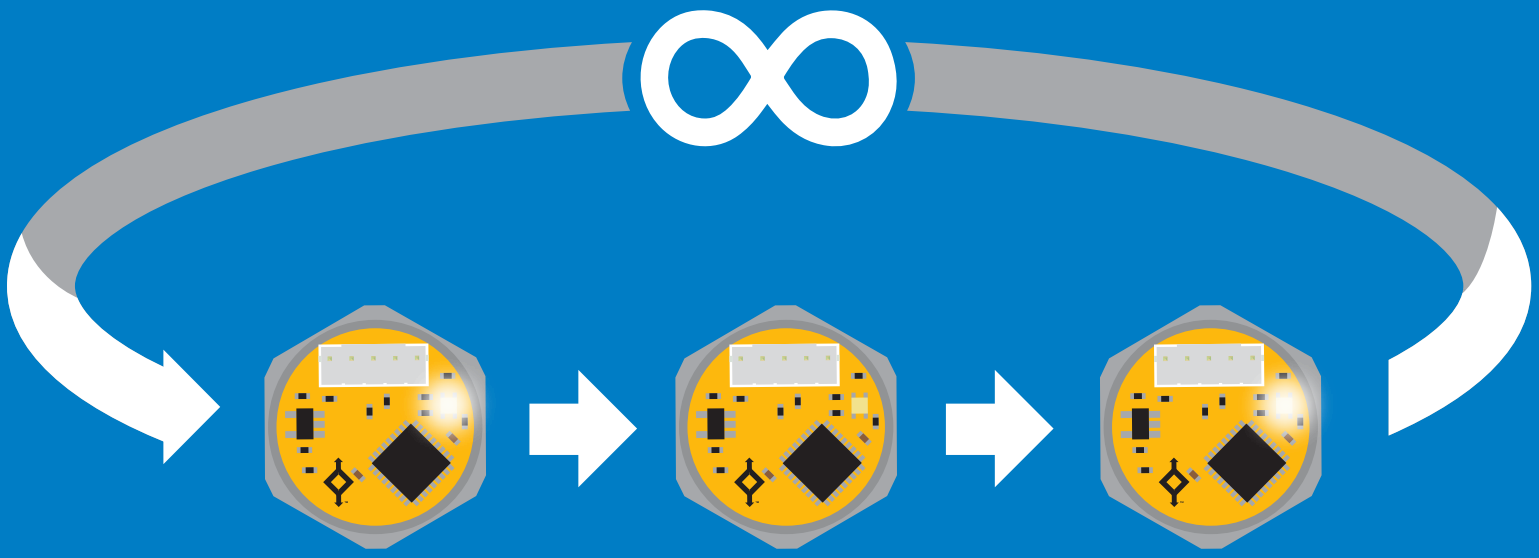
300ms processing delay

## Command syntax

This command will disable continuous mode.
Send any character or command to terminate find.

**Find** LED rapidly blinks white, used to help find device

| Example | Response |
|---|---|
| Find | Wait 300ms; 1 Dec; 0 Null |






Atlas Scientific Environmental Robotics

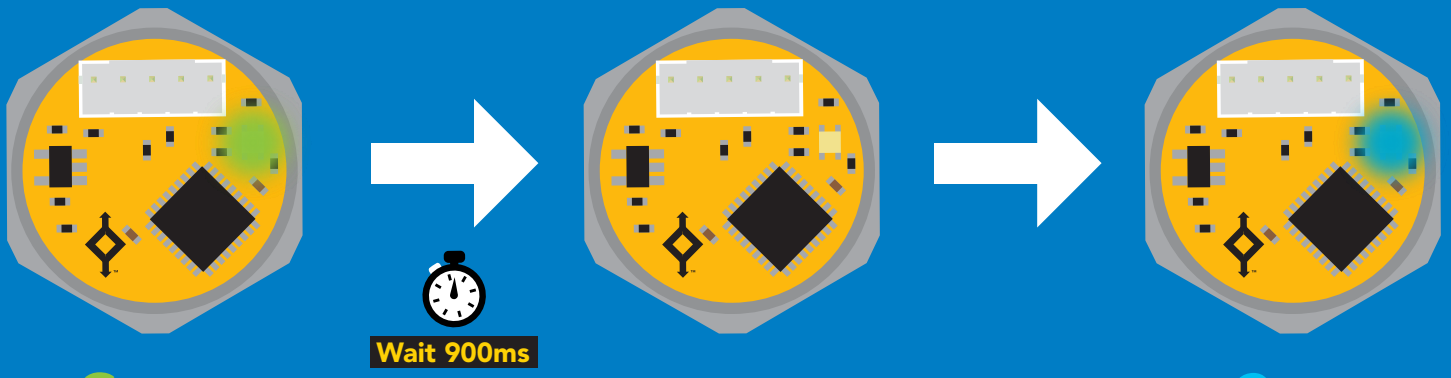
# Taking reading

## Command syntax

900ms processing delay

**R** return 1 reading

| Example | Response |
|---|---|
| R | Wait 900ms · 1 Dec · 38.462 ASCII · 0 Null |




**Green**
Taking reading

Transmitting

**Cyan**
Standby




AtlasScientific™ Environmental Robotics

# Alarm

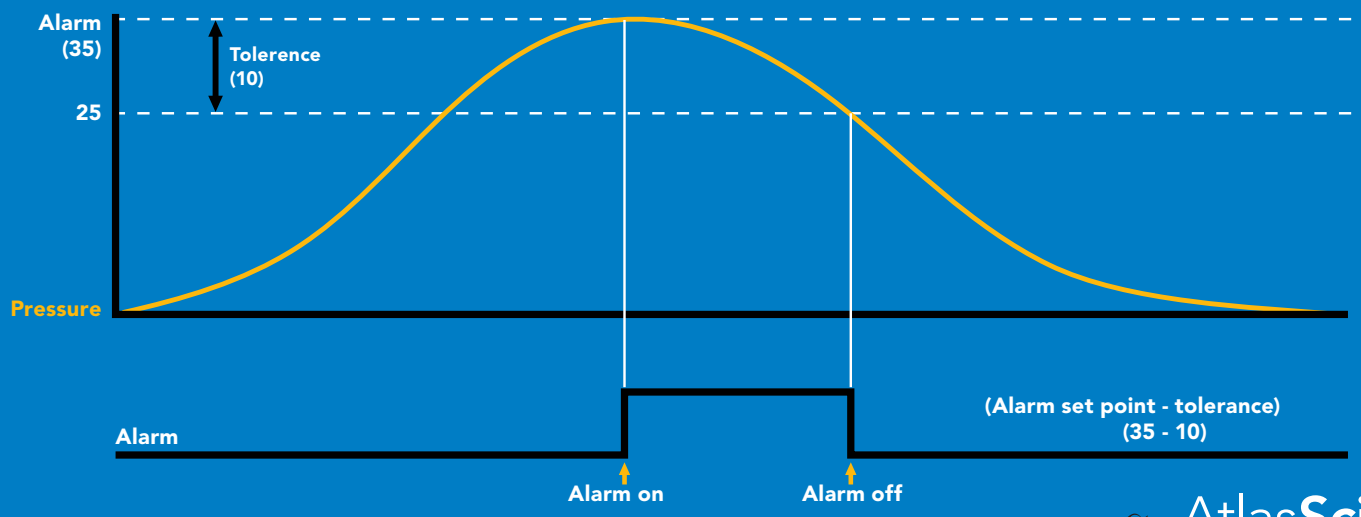
300ms processing delay

## Command syntax

The alarm pin will = 1 when pressure levels are > alarm set point. Alarm tolerance sets how far below the set point pressure levels need to drop before the pin will = 0 again.

| | |
|---|---|
| Alarm,en,[1,0] | enable / disable alarm |
| Alarm,n | sets alarm |
| Alarm,tol,n | sets alarm tolerance |
| Alarm,? | alarm set? |

| Example | Response |
|---|---|
| Alarm,en,1 | Wait 300ms → 1 (Dec) 0 (Null) Enable alarm |
| Alarm,35 | Wait 300ms → 1 (Dec) 0 (Null) |
| Alarm,tol,10 | Wait 300ms → 1 (Dec) 0 (Null) Pressure level must fall 10 units below set point for alarm to reset. |
| Alarm,? | Wait 300ms → 1 (Dec) ?,alarm,35,10,1 (ASCII) 0 (Null) if all are enabled |







AtlasScientific™ Environmental Robotics

# Custom calibration

900ms processing delay

## Command syntax

Although calibration is not required, it may be necessary to adjust your 0 point or perform a custom calibration.

| | |
|---|---|
| Cal,n | calibrates the high point |
| Cal,0 | calibrates the zero point |
| Cal,clear | restores calibration to factory settings |
| Cal,? | device calibrated? |

| Example | Response |
|---|---|
| Cal,50 | Wait 900ms → 1 (Dec) 0 (Null) high point calibration in psi |
| Cal,0 | Wait 900ms → 1 (Dec) 0 (Null) low point calibration in psi |
| Cal,clear | Wait 300ms → 1 (Dec) 0 (Null) |
| Cal,? | Wait 300ms → 1 (Dec) ?Cal,0 (ASCII, no calibration) 0 (Null) or 1 (Dec) ?Cal,1 (ASCII, only zero point calibration) 0 (Null) or 1 (Dec) ?Cal,2 (ASCII, only high point calibration) 0 (Null) or 1 (Dec) ?Cal,3 (ASCII, zero and high point calibration) 0 (Null) |

**Calibration should be done using the pressure scale you have set the sensor to.**

**Example**

**Readings are set to bar.**
**High point calibration = 3.44**
**(3.44 bar = 50 psi)**




Atlas Scientific Environmental Robotics

# Add/remove decimal places

900ms processing delay

## Command syntax

Change how many decimal points the reading outputs.

| | |
|---|---|
| Dec,n | n = number of decimal points between 0 and 3 |
| Dec,? | number of decimal points the output is set to |

| Example | Response |
|---|---|
| R | Wait 900ms — 1 (Dec) — 38.462 (ASCII) — 0 (Null) |
| Dec,1 | Wait 900ms — 1 (Dec) — 0 (Null) |
| R | Wait 900ms — 1 (Dec) — 38.4 (ASCII) — 0 (Null) |
| Dec,? | Wait 900ms — 1 (Dec) — ?Dec,1 (ASCII) — 0 (Null) |

# Pressure units

(psi, atm, bar, kPa, inch's of water, cm of water)

## Command syntax

| | | |
|---|---|---|
| U,[1/0] | 1 will add a unit identifier to the output | |
| U,psi | output will be in psi | |
| U,atm | output will be in atm | |
| U,bar | output will be in bar | |
| U,kPa | output will be in kPa | |
| U,inh2o | output will be in inches of water | (Resolution: 0.027") |
| U,cmh2o | output will be in cm of water | (Resolution: 0.7mm) |
| U,? | pressure units? | |

| Example | Response |
|---|---|
| U,bar |  Wait 300ms — 1 Dec — 0 Null |
| U,1 |  Wait 300ms — 1 Dec — 0 Null<br>  Wait 300ms — 1 Dec — 1.228,bar ASCII — 0 Null |
| U,? |  Wait 300ms — 1 Dec — ?U,bar ASCII — 0 Null |




AtlasScientific™ Environmental Robotics

# Naming device

300ms processing delay

## Command syntax

Do not use spaces in the name

| | |
|---|---|
| Name,n | set name |
| Name, | clears name |
| Name,? | show name |

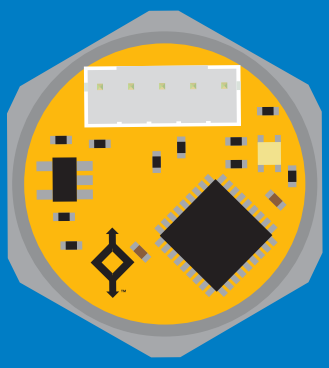
n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Up to 16 ASCII characters

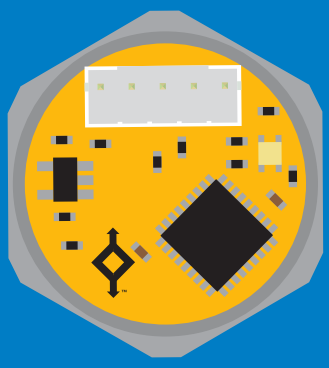
| Example | Response |
|---|---|
| Name, | Wait 300ms — 1 Dec — 0 Null — name has been cleared |
| Name,zzt | Wait 300ms — 1 Dec — 0 Null |
| Name,? | Wait 300ms — 1 Dec — ?Name,zzt ASCII — 0 Null |

Name,zzt



1 0

Name,?



1 ?Name,zzt 0




AtlasScientific™ Environmental Robotics

# Device information

## Command syntax

300ms processing delay

**i** device information

## Example

**i**

## Response


**1** Dec

**?i,PRS, 1.0** ASCII

**0** Null

## Response breakdown

**?i, PRS, 1.0**

PRS, ↑ Device

1.0 ↑ Firmware




Atlas Scientific™ Environmental Robotics

# Reading device status

## Command syntax

300ms processing delay

**Status** voltage at Vcc pin and reason for last restart

## Example Response

**Status**

Wait 300ms

| 1 | ?Status,P,5.038 | 0 |
|---|---|---|
| Dec | ASCII | Null |

## Response breakdown

**?Status, P, 5.038**

P, ↑ Reason for restart

5.038 ↑ Voltage at Vcc

### Restart codes

| | |
|---|---|
| P | powered off |
| S | software reset |
| B | brown out |
| W | watchdog |
| U | unknown |




Atlas Scientific Environmental Robotics

# Sleep mode/low power

## Command syntax

| | |
|---|---|
| Sleep | enter sleep mode/low power |

Send any character or command to awaken device.

| Example | Response |
|---|---|
| Sleep | no response |
| Any command | wakes up device |

Do not read status byte after issuing sleep command.

| | STANDBY | SLEEP |
|---|---|---|
| 5V | 13.4 mA | 0.415 mA |
| 3.3V | 12.4 mA | 0.13 mA |


Standby

Sleep

# Protocol lock

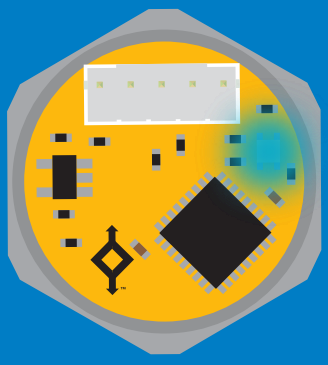
## Command syntax

300ms processing delay

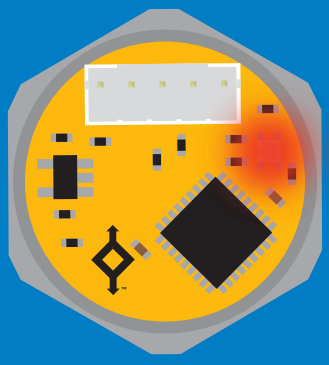
| | | |
|---|---|---|
| Plock,1 | enable Plock | Locks device to I²C mode. |
| Plock,0 | disable Plock | default |
| Plock,? | Plock on/off? | |

| Example | Response |
|---|---|
| Plock,1 | Wait 300ms → 1 (Dec) 0 (Null) |
| Plock,0 | Wait 300ms → 1 (Dec) 0 (Null) |
| Plock,? | Wait 300ms → 1 (Dec) ?Plock,1 (ASCII) 0 (Null) |

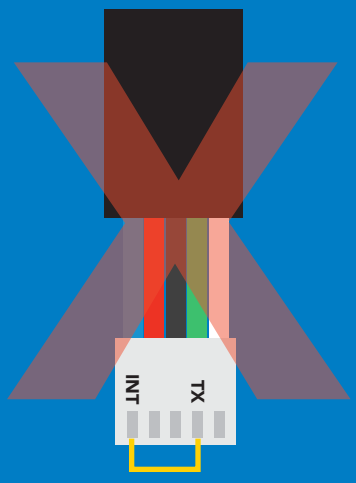
Plock,1



Baud, 9600



cannot change to UART




cannot change to UART




Atlas Scientific Environmental Robotics

# I²C address change

## Command syntax

300ms processing delay

I2C,n  sets I²C address and reboots into I²C mode

| Example | Response |
|---|---|
| I2C,101 | device reboot<br>(no response given) |

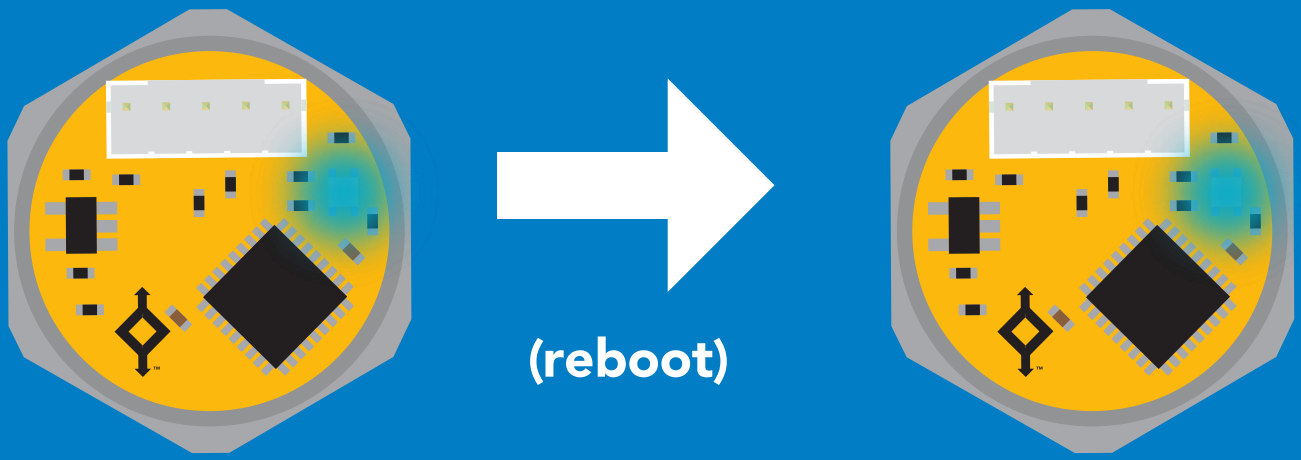
**Warning!**

Changing the I²C address will prevent communication between the device and the CPU until the CPU is updated with the new I²C address.

Default I²C address is 106 (0x6A).

n = any number 1 – 127

I2C,101







Atlas Scientific Environmental Robotics

# Factory reset

## Command syntax

Factory reset will not take the device out of I²C mode.

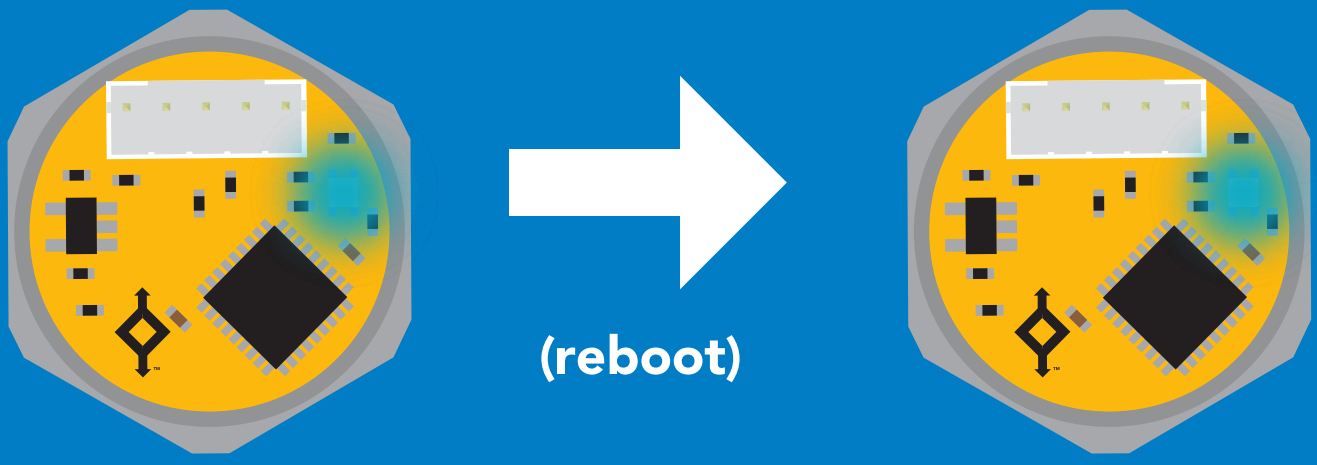
Factory enable factory reset

I²C address will not change

| Example | Response |
| --- | --- |
| Factory | device reboot<br>(no response given) |

Clears calibration
LED on
Response codes enabled

### Factory



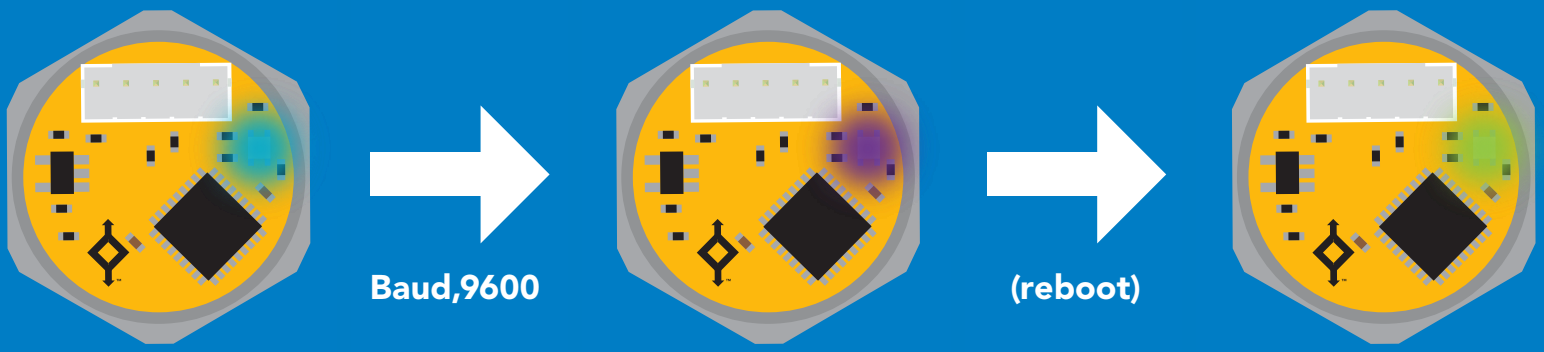
# Change to UART mode

## Command syntax

**Baud,n** switch from I²C to UART

| Example | Response |
|---|---|
| Baud,9600 | reboot in UART mode<br>(no response given) |

n =
- 300
- 1200
- 2400
- 9600
- 19200
- 38400
- 57600
- 115200

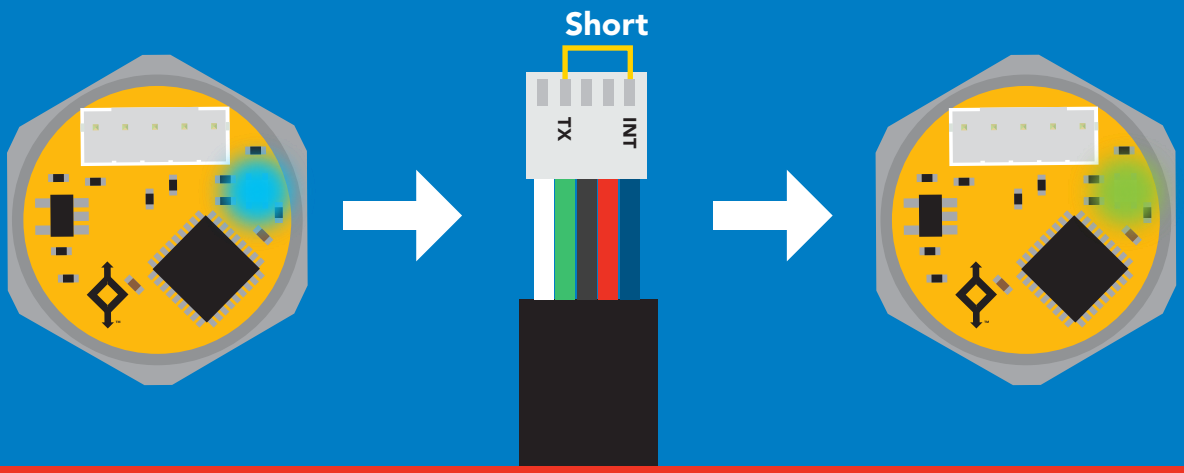


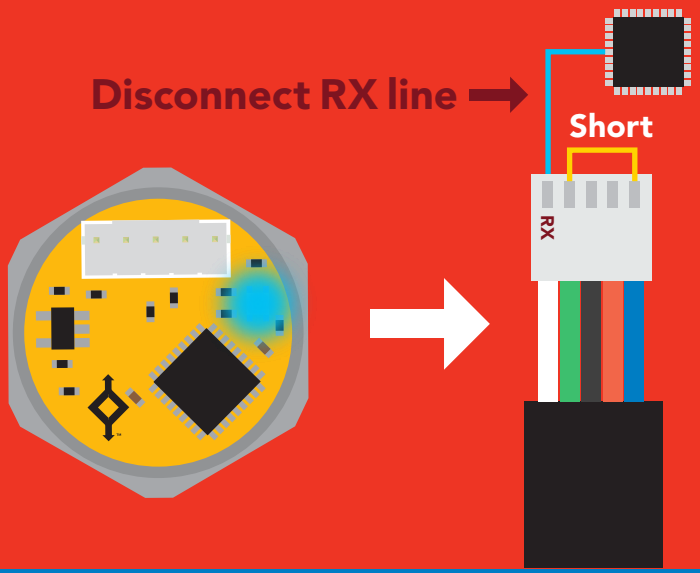

Changing to UART mode

# Manual switching to UART

- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to INT
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from Blue to Green
- Disconnect ground (power off)
- Reconnect all data and power

## Example



# Datasheet change log

### Datasheet V 2.1

Revised artwork on page 8.

### Datasheet V 2.0

Revised design of EZO-PRS throughout document.

### Datasheet V 1.6

Revised naming device info on pages 25 & 49.

### Datasheet V 1.5

Added the custom calibartion pages on pages 22 & 46.

### Datasheet V 1.4

Revised accuracy value on cover page.

### Datasheet V 1.3

Revised pressure output in PSI to 50.000 on cover pg.
added inches of water & cm of water resolution info on pages 23 and 46.

### Datasheet V 1.2

Updated firmware info on pg 55.

### Datasheet V 1.1

Moved Default state to pg 9.

### Datasheet V 1.0

Initial release - New datasheet

# Firmware updates

V1.0 – Initial release (Aug, 7 2019)

V1.01 – (Nov, 5 2019)
- Fixed glitch where the alarm was not initially set correctly.

V1.02 – (April, 9 2021)
- Added custom calibration

# Warranty

Atlas Scientific™ Warranties the EZO-PRS™ Embedded Pressure Sensor to be free of defect during the debugging phase of device implementation, or 30 days after receiving the EZO-PRS™ Embedded Pressure Sensor (which ever comes first).

# The debugging phase

The debugging phase as defined by Atlas Scientific™ is the time period when the EZO-PRS™ Embedded Pressure Sensor is inserted into a bread board, or shield. If the EZO-PRS™ is being debugged in a bread board, the bread board must be devoid of other components. If the EZO-PRS™ Embedded Pressure Sensor is being connected to a microcontroller, the microcontroller must be running code that has been designed to drive the EZO-PRS™ Embedded Pressure Sensor exclusively and output the EZO-PRS™ data as a serial string.

**It is important for the embedded systems engineer to keep in mind that the following activities will void the EZO-PRS™ Embedded Pressure Sensor warranty:**

- **Soldering any part of the EZO-PRS™ Embedded Pressure Sensor.**
- **Running any code, that does not exclusively drive the EZO-PRS™ Embedded Dosing Pump and output its data in a serial string.**
- **Embedding the EZO-PRS™ Embedded Pressure Sensor into a custom made device.**
- **Removing any potting compound.**

# Reasoning behind this warranty

Because Atlas Scientific™ does not sell consumer electronics; once the device has been embedded into a custom made system, Atlas Scientific™ cannot possibly warranty the EZO-PRS™ Embedded Pressure Sensor, against the thousands of possible variables that may cause the EZO-PRS™ Embedded Pressure Sensor to no longer function properly.

## Please keep this in mind:

1. **All Atlas Scientific™ devices have been designed to be embedded into a custom made system by you, the embedded systems engineer.**

2. **All Atlas Scientific™ devices have been designed to run indefinitely without failure in the field.**

3. **All Atlas Scientific™ devices can be soldered into place, however you do so at your own risk.**

Atlas Scientific™ is simply stating that once the device is being used in your application, Atlas Scientific can no longer take responsibility for the EZO-PRS™ Embedded Pressure Sensors continued operation. This is because that would be equivalent to Atlas Scientific™ taking responsibility over the correct operation of your entire device.