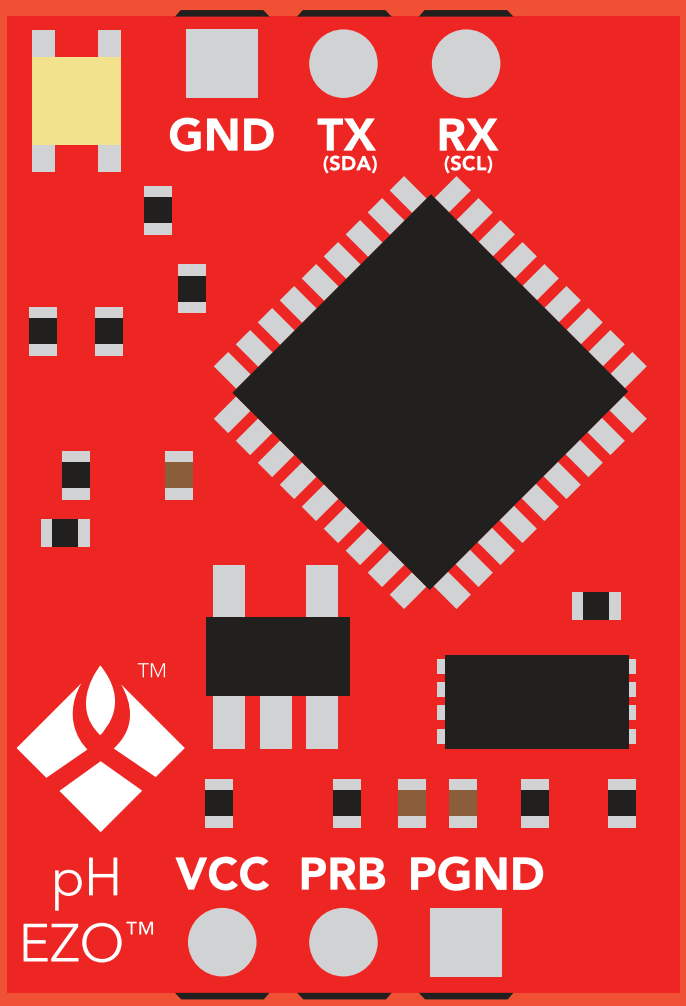
Atlas Scientific™
Environmental Robotics
V 6.1
Revised 5/23

# EZO-pH™

**Embedded pH Circuit**
**ISO 10523 Compliant**
(determination of pH)

| | |
|---|---|
| Reads | **pH** |
| Range | **.001 – 14.000** |
| Resolution | **.001** |
| Accuracy | **+/– 0.002** |
| pH reading time | **800ms** |
| Supported probes | **Any type & brand** |
| Calibration | **1, 2, 3 point** |
| Temp compensation | **Yes** |
| Data protocol | **UART & I²C** |
| Default I²C address | **99 (0x63)** |
| Operating voltage | **3.3V – 5V** |
| Data format | **ASCII** |

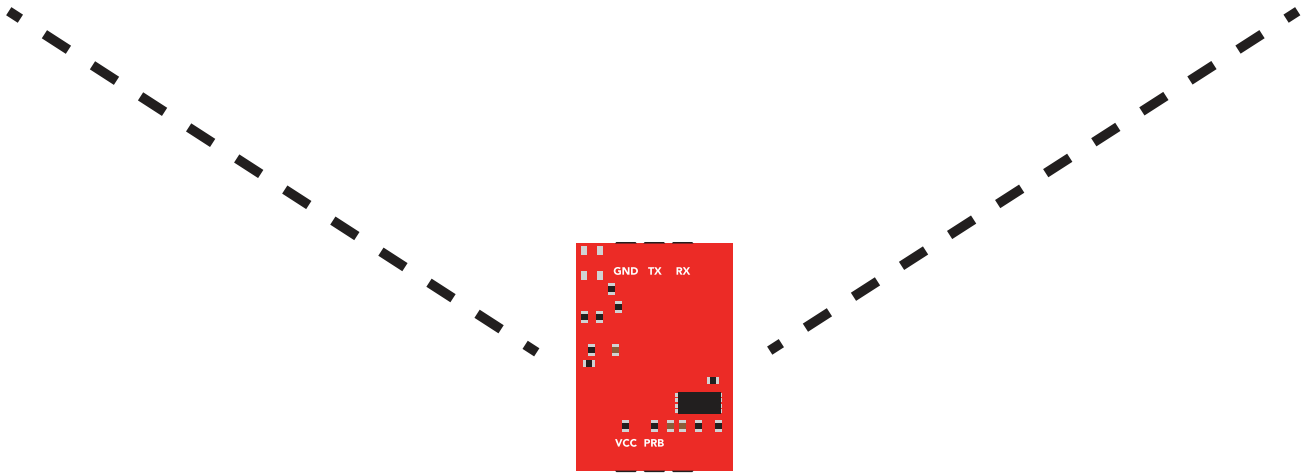



RoHS COMPLIANT

PATENT PROTECTED

Written by Jordan Press
Designed by Noah Press

This is an evolving document, check back for updates.


GND TX RX
VCC PRB

# ✓ Available data protocols

## UART

Default

## I²C

# X Unavailable data protocols

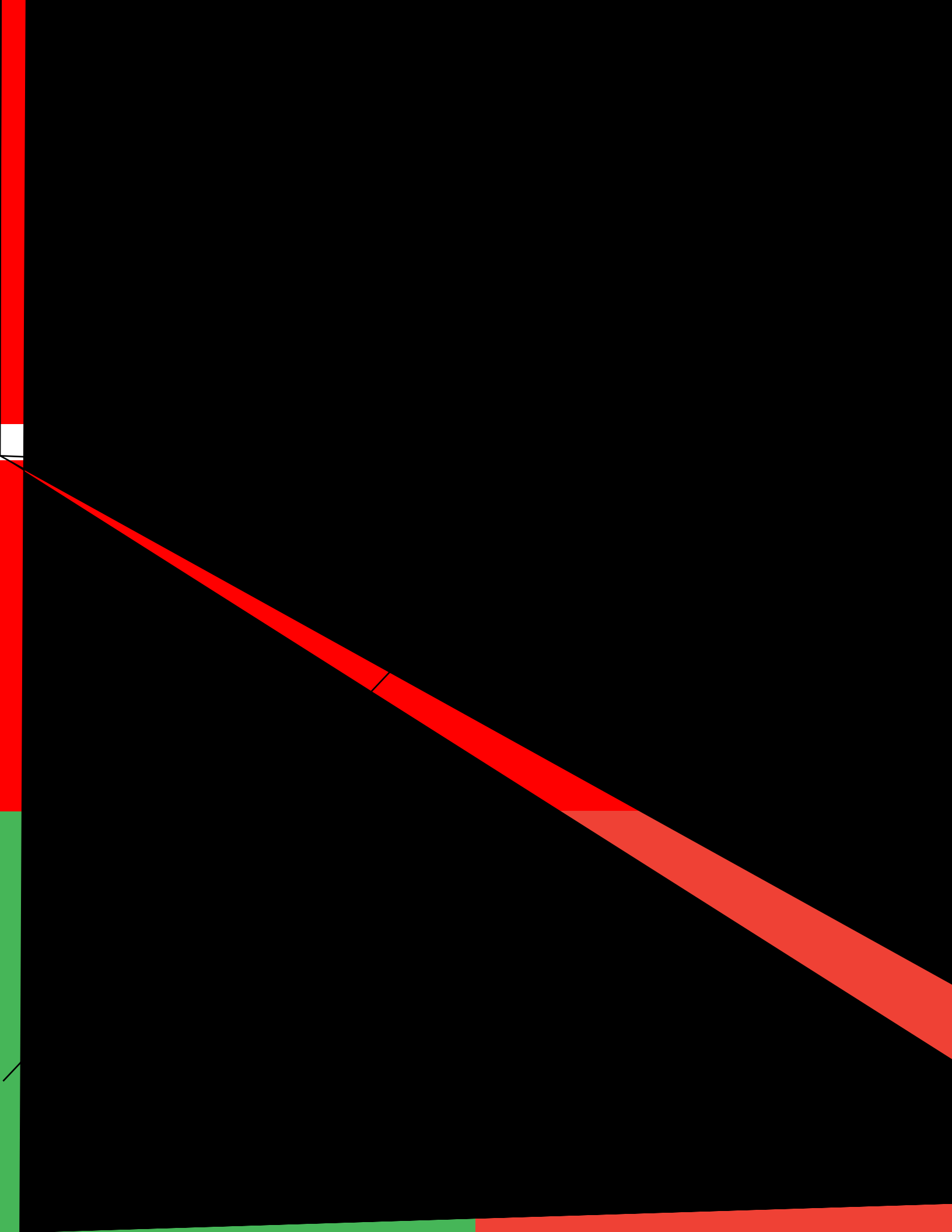
## SPI

## Analog

## RS-485

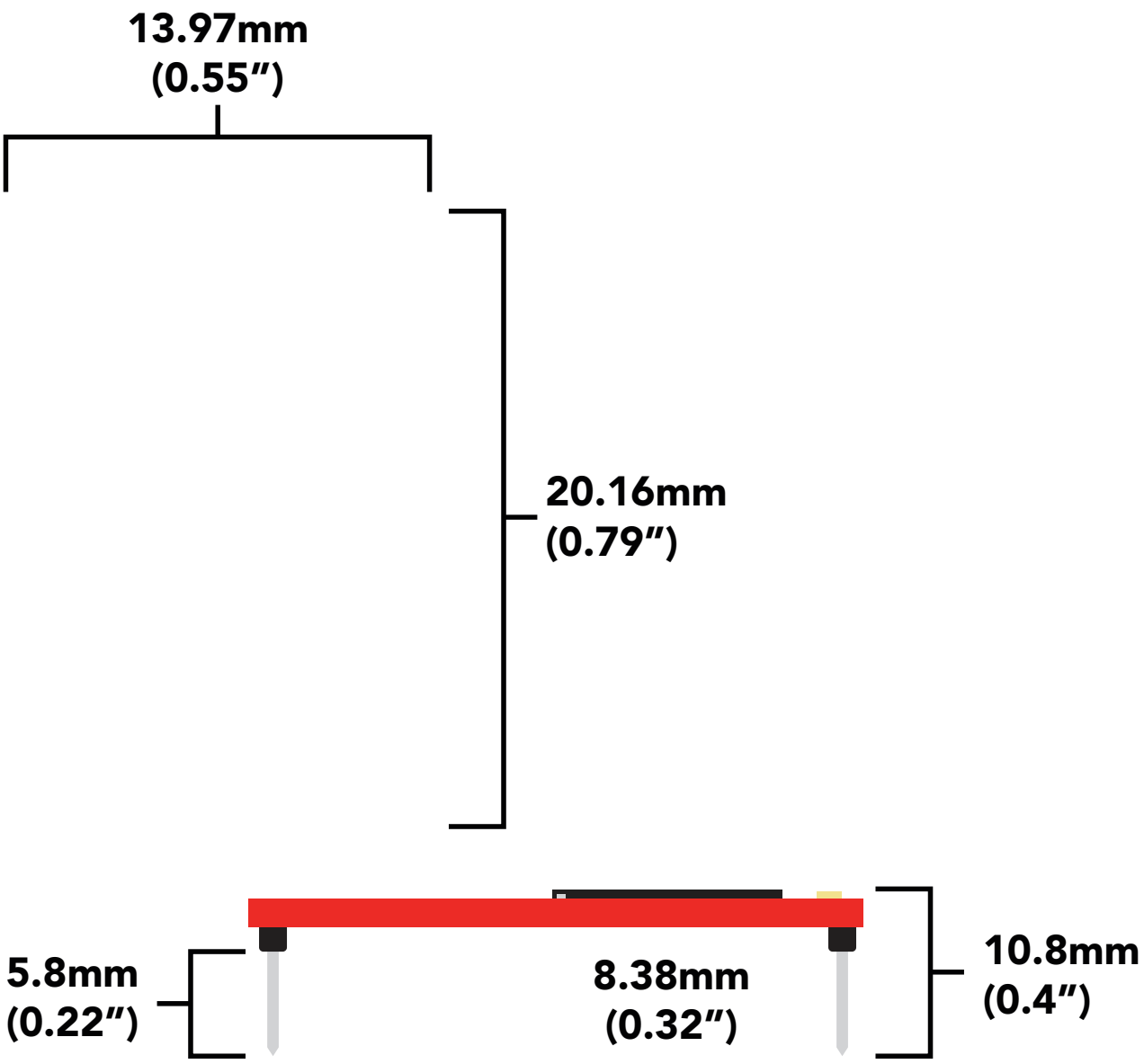
## Mod Bus

## 4–20mA



# UART

# EZO™ circuit dimensions




## Power consumption

| | LED | MAX | STANDBY | SLEEP |
|---|---|---|---|---|
| 5V | ON | 18.3 mA | 16 mA | 1.16 mA |
| | OFF | 13.8 mA | 13.8 mA | |
| 3.3V | ON | 14.5 mA | 13.9 mA | 0.995 mA |
| | OFF | 13.3 mA | 13.3 mA | |

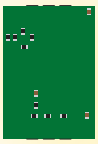
## Absolute max ratings

| Parameter | MIN | TYP | MAX |
|---|---|---|---|
| Storage temperature (EZO™ pH) | -65 °C | | 125 °C |
| Operational temperature (EZO™ pH) | -40 °C | 25 °C | 85 °C |
| VCC | 3.3V | 5V | 5.5V |

# Electrical isolation

The Atlas Scientific EZO™ pH circuit is a very sensitive device. This sensitivity is what gives the pH circuit its accuracy. It also means that the pH circuit can read micro-voltages bleeding into the water from unnatural sources such as pumps, solenoid valves, or other probes/sensors.

When electrical noise interferes with the pH readings, it is common to see rapidly fluctuating readings or readings that are pinned to 14 or 0. To verify that electrical noise is causing inaccurate readings, place the pH probe in a cup of water by itself. The readings should stabilize quickly, confirming that electrical noise was the issue.



## Advice:

1. When reading pH along with other sensors, electrical isolation is strongly recommended.
2. Never build a commercial product without electrical isolation.

Atlas Scientific offers several different electrical isolation products that can be used in your design. Select the electrical isolation product that works best for your design.

Basic EZO™
Inline Voltage Isolator

Vertical Isolator

Electrically Isolated
EZO™ Carrier Board


Gen 2 Electrically Isolated
USB EZO™ Carrier Board

Whitebox T1

Whitebox T3

Whitebox T3

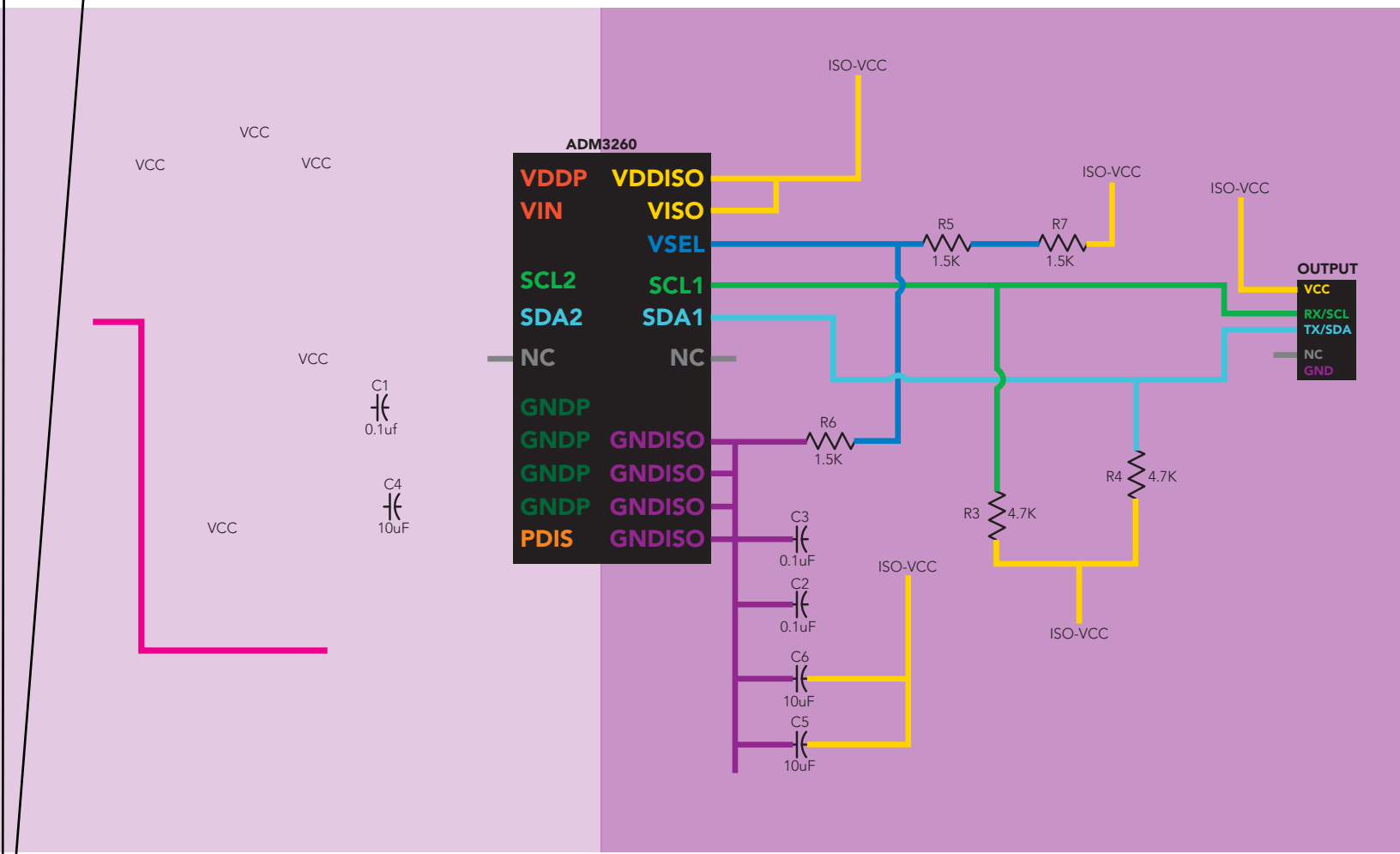
Electrically Isolated EZO™
Carrier Board (old style)

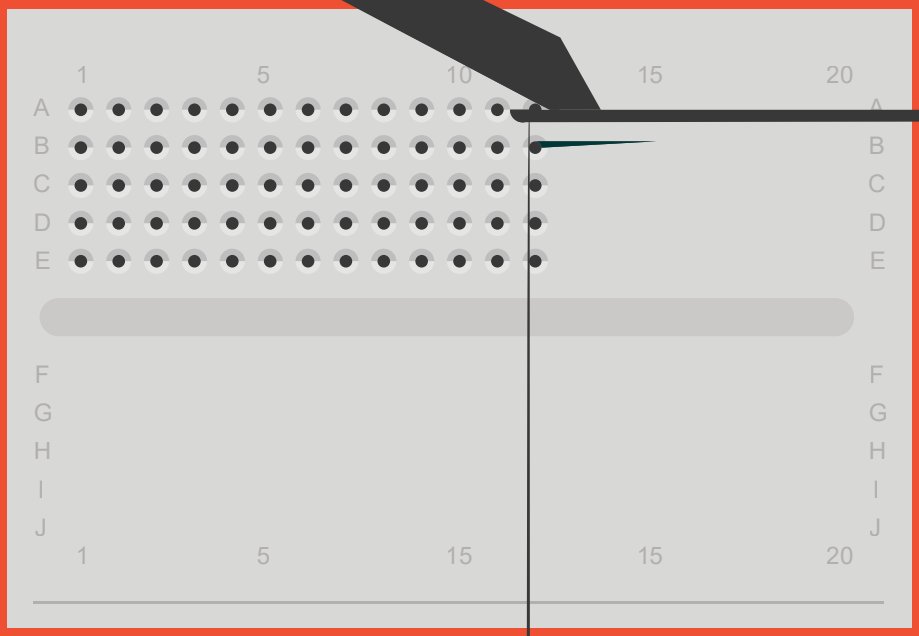
For various reasons, you may need to build your own electrical isolator. Because electrical isolation is so important, we have published our isolation schematic for anyone to use.

This isolation schematic is based on the ADM3260, which can output up to 150 mW of isolated power. PCB layout requires special attention for EMI/EMC and RF Control. Having good ground planes and keeping the capacitors as close to the chip as possible are crucial for proper performance.

The two data channels have a 4.7kΩ pull-up resistor on both the isolated and non-isolated lines (R1, R2, R3, and R4). The output voltage is set using a voltage divider (R5, R6, and R7). This produces a voltage of 3.9V regardless of your input voltage.

Isolated ground is different from non-isolated ground, these two lines should not be connected together.





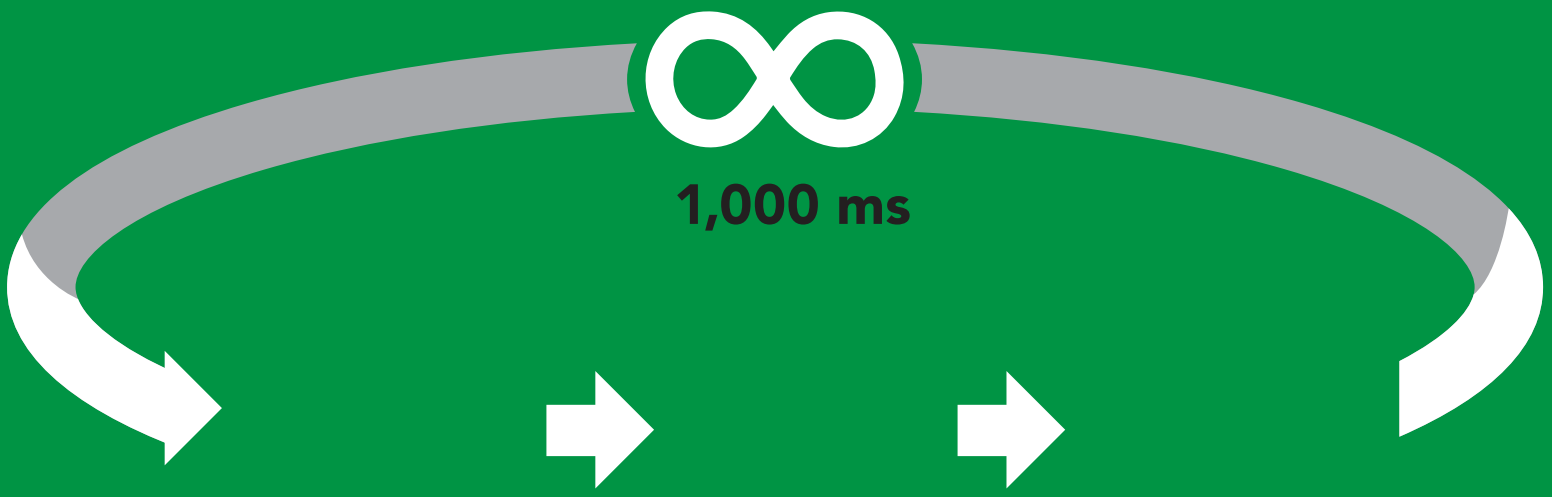

DO NOT CUT THE PROBE CABLE WITHOUT REFERING TO ***THIS DOCUMENT!***

ONLY USE SHIELDED CABLES.

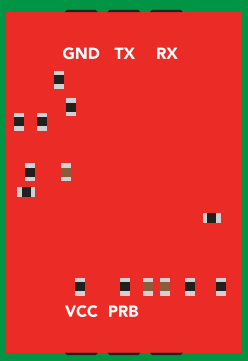
# Default state
# UART mode




**Green**
Standby

**Cyan**
Taking reading

Transmitting



# UART mode

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

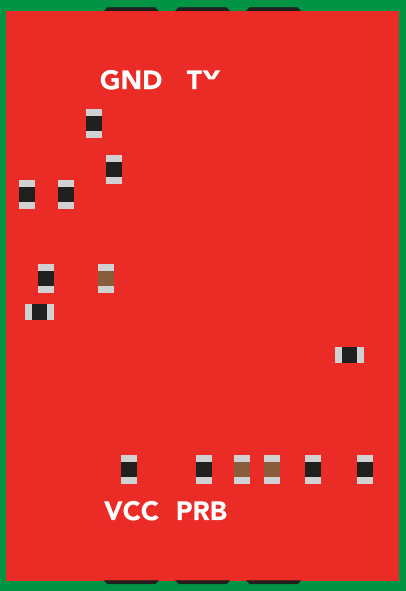
**Baud**

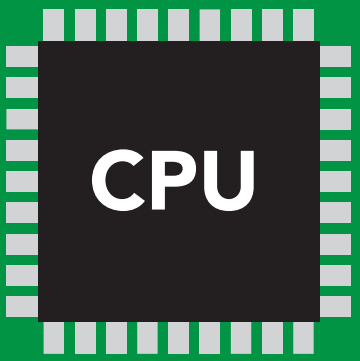
- 300
- 1,200
- 2,400
- **9,600 default**
- 19,200
- 38,400
- 57,600
- 115,200

**RX**
Data in

**TX**
Data out

**Vcc** 3.3V – 5.5V


# Data format

| | |
|---|---|
| Reading | pH |

| | |
|---|---|
| Data type | floating point |
| Decimal places | 3 |
| Smallest string | 4 characters |
| Largest string | 40 characters |

# LED color definition

| | LED ON |
|---|---|
| 5V | +2.2 mA |
| 3.3V | +0.6 mA |

# Receiving data from device

2 parts

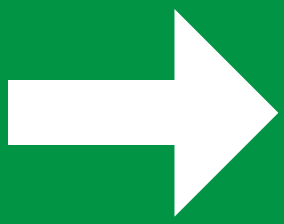
ASCII data string

Command

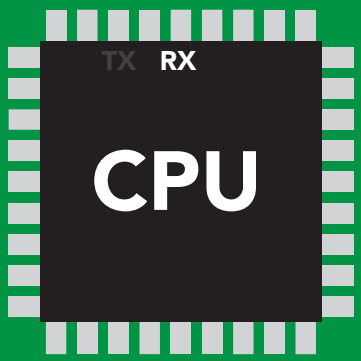
Carriage return <cr>

Terminator

9,600 baud
(default)



9.560 <cr>




Receiver

Sender

## Advanced

| | | | |
|---|---|---|---|
| ASCII: | 9 | 5 | <cr> |
| Hex: | 39 | 35 | 0D |
| Dec: | 57 | 53 | 13 |

# Sending commands to device

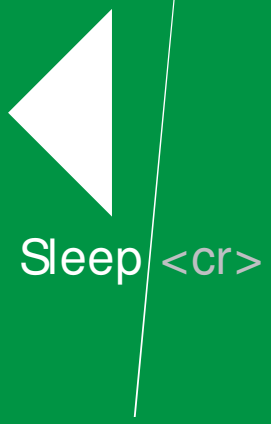
2 parts

Command (not case sensitive)

ASCII data string

Carriage return <cr>

Terminator



Sleep <cr>

Sender

Receiver

| | | | | | | |
|---|---|---|---|---|---|---|
| ASCII: | S | l | e | e | p | |
| Hex: | 53 | 6C | | | | |
| Dec: | 83 | 108 | 101 | | | |

# UART mode

## command quick reference

All commands are ASCII strings or single ASCII characters.

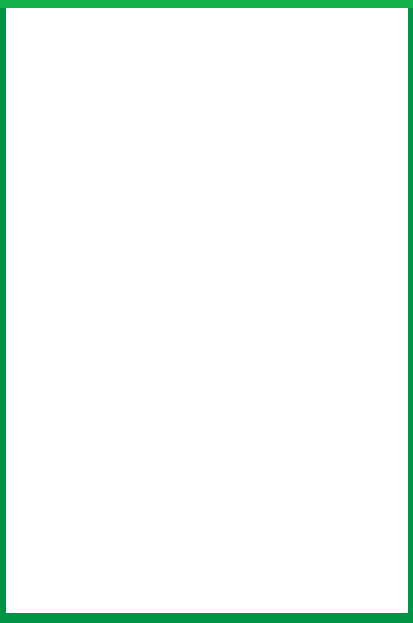
| Command | Function | | Default state |
|---|---|---|---|
| Baud | change baud rate | pg. 33 | 9,600 |
| C | enable/disable continuous reading | pg. 20 | enabled |
| Cal | performs calibration | pg. 22 | n/a |
| Export | export calibration | pg. 23 | n/a |
| Factory | enable factory reset | pg. 35 | n/a |
| Find | finds device with blinking white LED | pg. 19 | n/a |
| i | device information | pg. 29 | n/a |
| I2C | change to I²C mode | pg. 36 | not set |
| Import | import calibration | pg. 24 | n/a |
| L | enable/disable LED | pg. 18 | enabled |
| Name | set/show name of device | pg. 28 | not set |
| pHext | enable/disable extended pH scale | pg. 26 | disabled |
| Plock | enable/disable protocol lock | pg. 34 | disabled |
| R | returns a single reading | pg. 21 | n/a |
| Sleep | enter sleep mode/low power | pg. 32 | n/a |
| Slope | returns the slope of the pH probe | pg. 25 | n/a |
| Status | retrieve status information | pg. 31 | enable |
| T | temperature compensation | pg. 27 | 25˚C |
| *OK | enable/disable response codes | pg. 30 | 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

Atlas

# 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 reading 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

| Example | Response |
|---|---|
| C,1 <cr> | *OK <cr><br>pH (1 sec) <cr><br>pH (2 sec) <cr><br>pH (n sec) <cr> |
| C,30 <cr> | *OK <cr><br>pH (30 sec) <cr><br>pH (60 sec) <cr><br>pH (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

A single reading takes 800ms

R <cr> takes single reading

| Example | Response |
| --- | --- |
| R <cr> | 9.560 <cr><br>*OK <cr> |

**Green**
**Standby**

**Cyan**
**Taking reading**

**Transmitting**

800 ms

# Calibration

## Command syntax

Issuing the cal,mid command after the EZO™ pH circuit has been calibrated, will clear the other calibration points. Full calibration will have to be redone.

| | | |
|---|---|---|
| Cal,mid,n | <cr> | single point calibration at midpoint |
| Cal,low,n | <cr> | two point calibration at lowpoint |
| Cal,high,n | <cr> | three point calibration at highpoint |
| Cal,clear | <cr> | delete calibration data |
| Cal,? | <cr> | device calibrated? |

| Example | Response |
|---|---|
| Cal,mid,7.00 <cr> | *OK <cr> |
| Cal,low,4.00 <cr> | *OK <cr> |
| Cal,high,10.00 <cr> | *OK <cr> |
| Cal,clear <cr> | *OK <cr> |
| Cal,? <cr> | ?Cal,0 <cr> or ?Cal,1 <cr> (one point) or ?Cal,2 <cr> (two point) or ?Cal,3 <cr> (three point) *OK <cr> |

# Export calibration

## Command syntax

| | | |
|---|---|---|
| Export,? | <cr> | calibration string info |
| Export | <cr> | export calibration string from calibrated device |

## Example

Export,? <cr>

## Response

10,120 <cr>

### Response breakdown

# of strings to export

Export <cr>
Export <cr>
(7 more)
Export <cr>
Export <cr>

59 6F 75 20 61 72 <cr> (1 of 10)
65 20 61 20 63 6F <cr> (2 of 10)
⋮
6F 6C 20 67 75 79 <cr> (10 of 10)
*DONE

Disabling *OK simplifies this process

Export <cr>

# Import calibration

## Command syntax

Import: Use this command to upload calibration settings to one or more devices.

Import,n <cr> import calibration string to new device

| Example | | Response |
|---|---|---|
| Import, 59 6F 75 20 61 72 <cr> | (1 of 10) | *OK <cr> |
| Import, 65 20 61 20 63 6F <cr> | (2 of 10) | *OK <cr> |
| ⋮ | | ⋮ |
| Import, 6F 6C 20 67 75 79 <cr> | (10 of 10) | *OK <cr> |

* If one of the imported strings is not correctly entered, the device will not accept the import, respond with *ER

# Slope

After calibrating a pH probe issuing the

## Command syntax

Slope,? <cr> returns the slope of the pH probe

## Example

Slope,? <cr>

## Response

?Slope,99.7,100.3, -0.89 <cr>
*OK <cr>

## Response breakdown

?Slope, 99.7 100.3 -0.89

99.7% is how closely the slope of the acid calibration line matched the "ideal" pH probe.

100.3% is how closely the slope of the base calibration matches the "ideal" pH probe.

This is how many millivolts the zero point is off from true 0.

# Extended pH scale

Very strong acids and basses can exceed the traditional pH scale. This command extends the pH scale to show below 0 and above 14.

Lowest possible reading: -1.6
Highest possible reading: 15.6

## Command syntax

| | | |
|---|---|---|
| pHext,0 <cr> | extended pH scale off (0–14) | default |
| pHext,1 <cr> | extended pH scale on (-1.6–15.6) | |
| pHext,? <cr> | extended pH scale on/off? | |

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

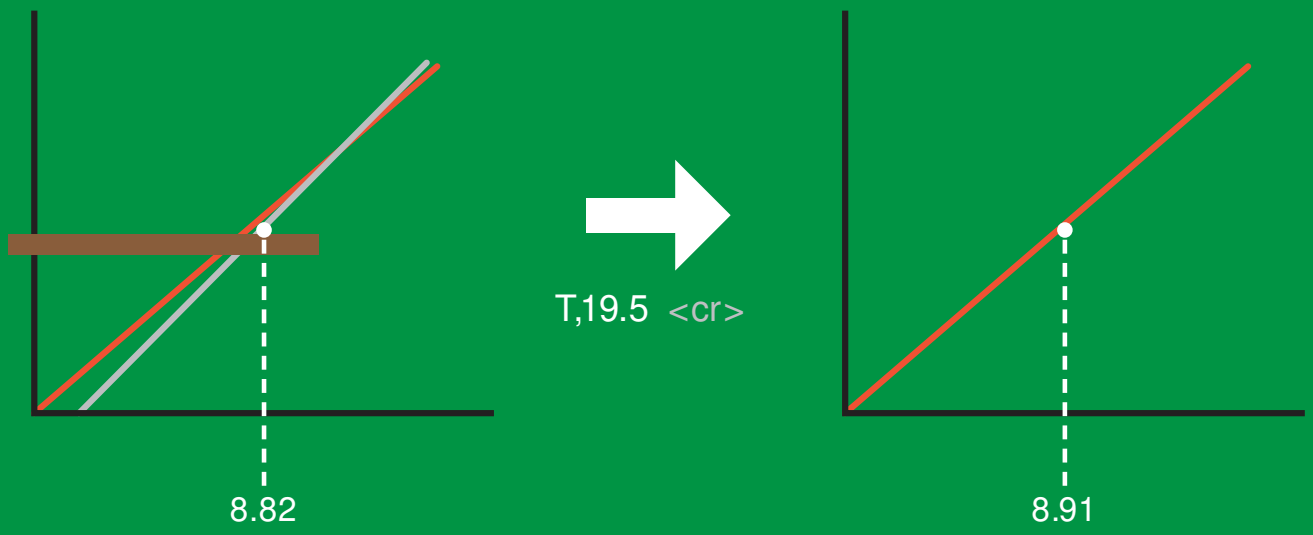
# Temperature compensation

## Command syntax

Default temperature = 25°C
Temperature is always in Celsius
Temperature is not retained if power is cut

| | | |
|---|---|---|
| T,n | <cr> | n = any value; floating point or int |
| T,? | <cr> | compensated temperature value? |
| RT,n | <cr> | set temperature compensation and take a reading |

| Example | Response |
|---|---|
| T,19.5 <cr> | *OK <cr> |
| RT,19.5 <cr> | *OK <cr><br>8.91 <cr> |
| T,? <cr> | ?T,19.5 <cr><br>*OK <cr> |



# 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>

# Device information

## Command syntax

i <cr> device information

| Example | Response |
| --- | --- |
| i <cr> | ?i,pH,2.16<cr><br>*OK <cr> |

## Response breakdown

?i, pH, 2.16

pH, ↑ Device

2.16 ↑ 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> | 9.560 <cr><br>*OK <cr> |
| *OK,0 <cr> | no response, *OK disabled |
| R <cr> | 9.560 <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

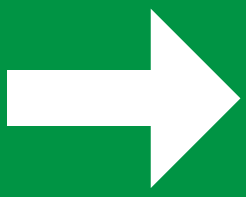
## Command syntax

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 | 16 mA | 1.16 mA |
| 3.3V | 13.9 mA | 0.995 mA |



Sleep <cr>

Standby
16 mA

Sleep
1.16 mA

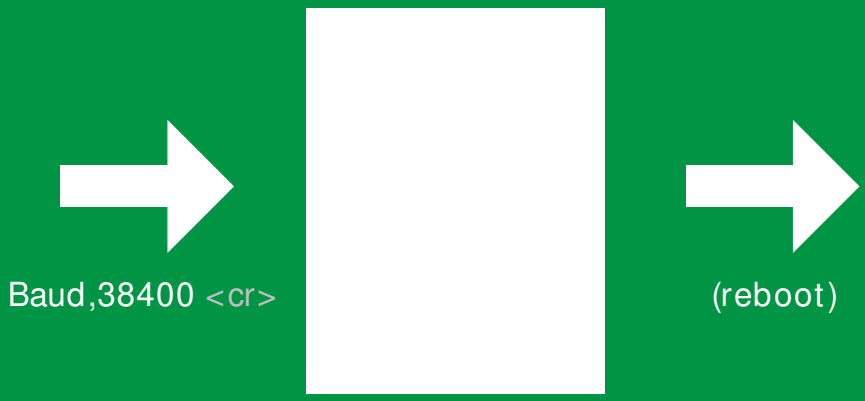
# Change baud rate

## 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



# P ck

## Command syntax

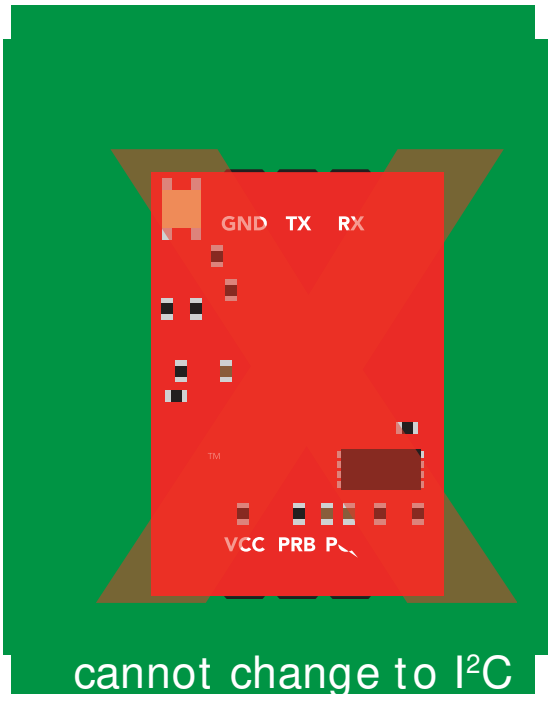
Locks device to UART mode.

Plock,1 <cr> enable Plock

Plock,0 <cr> disable Plock default

Plock,? <cr> Plock on/off?

| Example | Response |
|---|---|
| <cr> | <cr> |
| <cr> | <cr> |
| <cr> | <cr> or <cr> |




cannot change to I²C

# Factory reset

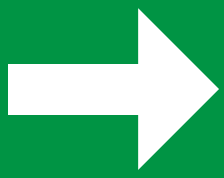
## Command syntax

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

# Change to I²C mode

## Command syntax

Default I²C address 99 (0x63)

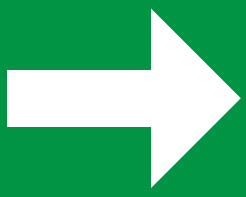
I2C,n <cr> sets I²C address and reboots into I²C mode

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

Manually switching to I²C will set the I²C address to 99 (0x63)

Disconnect RX line

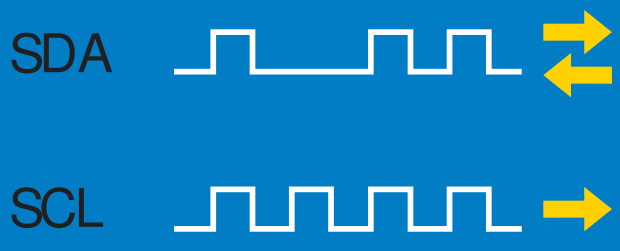



Atl

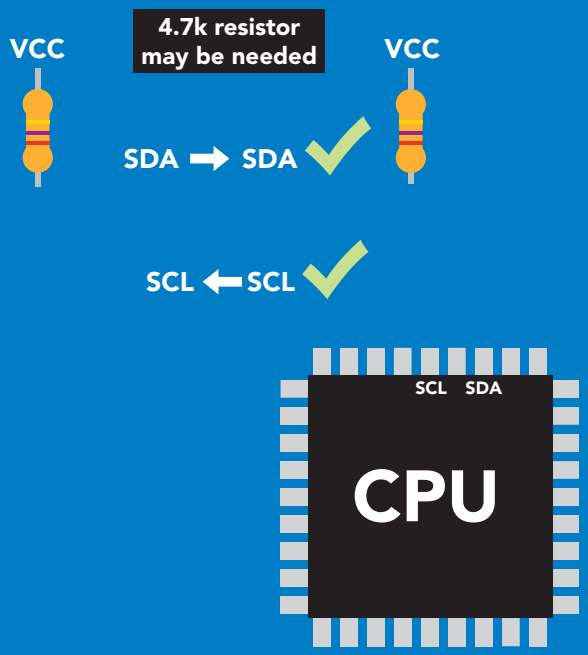
The I²C protocol is ***considerably more complex*** than the UART (RS–232) protocol. Atlas Scientific assumes the embedded systems engineer understands this protocol.

# I²C mode

| | |
|---|---|
| I²C address | (0x01 – 0x7F)<br>99 (0x63) default |
| Vcc | 3.3V – 5.5V |
| Clock speed | 100 – 400 kHz |





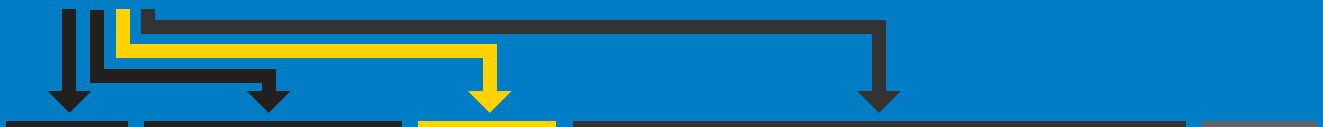


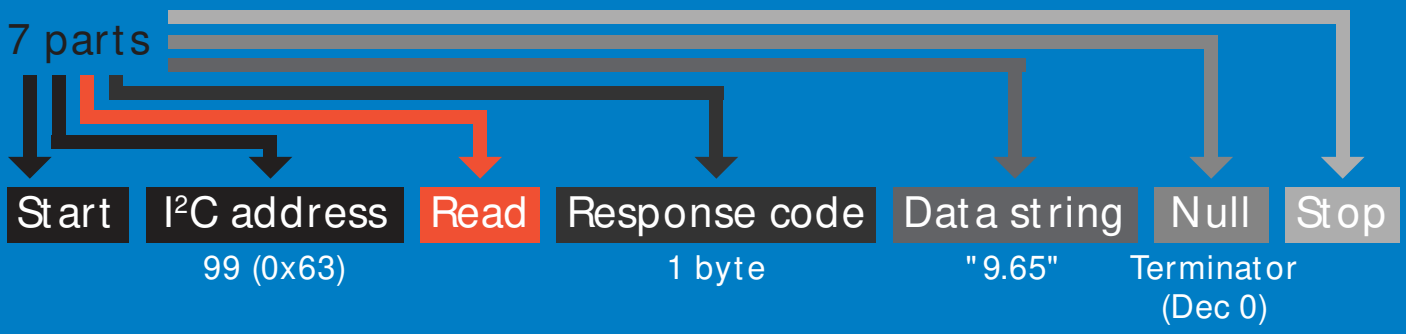

# Data format

| | | | |
|---|---|---|---|
| Reading | pH | Data type | floating point |
| Units | pH | Decimal places | 3 |
| Encoding | ASCII | Smallest string | 4 characters |
| Format | string | Largest string | 40 characters |

# Sending commands to device

5 parts





7 parts
Start
I²C address
99 (0x63)
Read
Response code
1 byte
Data string
"9.65"
Null
Terminator
(Dec 0)
Stop

# 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.***

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

delay(300);
```

Processing delay

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

# LED color definition


**Blue**

**I²C standby**

| | |
|---|---|
| 5V | LED ON<br>+2.2 mA |
| 3.3V | +0.6 mA |

All commands are ASCII strings or single ASCII characters.

| Command | Function | |
|---|---|---|
| | switch back to UART mode | pg. 61 |
| Cal | performs calibration | pg. 48 |
| | export calibration | pg. 49 |
| Factory | enable factory reset | pg. 60 |
| | finds device with blinking white LED | pg. 46 |
| i | device information | pg. 55 |
| | change $I^2C$ address | pg. 59 |
| Import | import calibration | pg. 50 |
| | enable/disable LED | pg. 45 |
| Name | set/show name of device | pg. 54 |
| | enable/disable extended pH scale | pg. 52 |
| Plock | enable/disable protocol lock | pg. 58 |
| | returns a single reading | pg. 47 |
| Sleep | enter sleep mode/low power | pg. 57 |
| | returns the slope of the pH probe | pg. 51 |
| Status | retrieve status information | pg. 56 |
| | temperature compensation | pg. 53 |

# 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 | 1 Dec, 0 Null |
| L,0 | 1 Dec, 0 Null |
| L,? | 1 Dec, ?L,1 ASCII, 0 Null<br>1 Dec, ?L,0 ASCII, 0 Null |

L,1 L,0

# Find

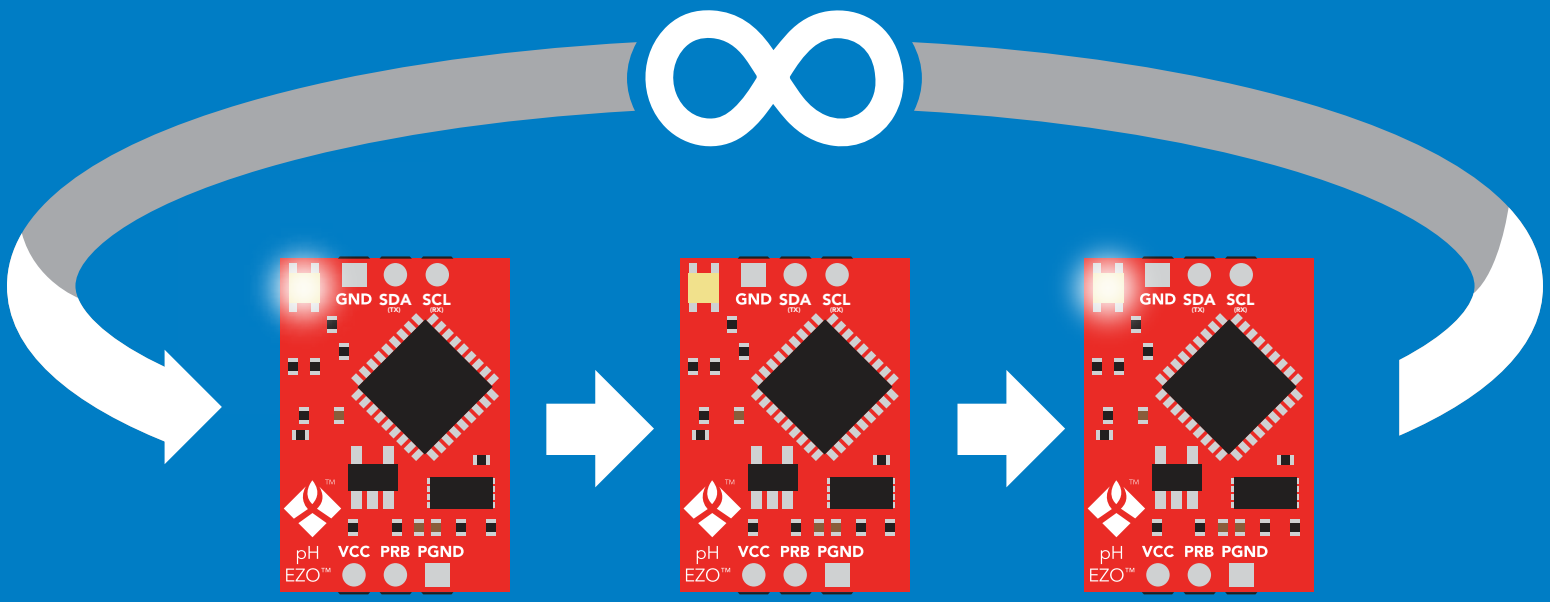
300ms processing delay

## 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<br>1 Dec<br>0 Null |



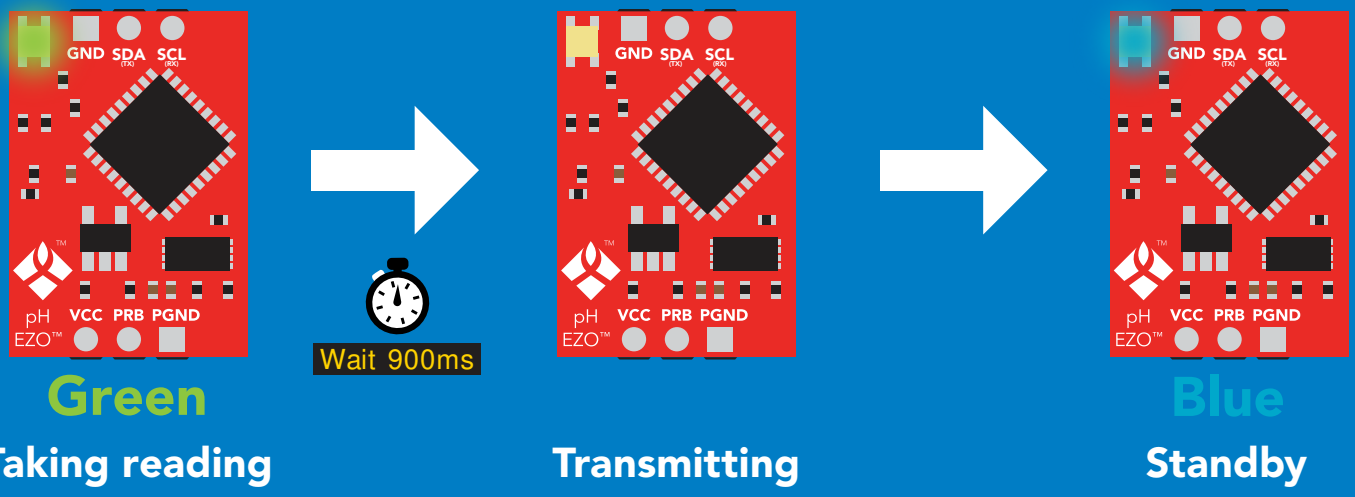
# Taking reading

## Command syntax

900ms processing delay

R return 1 reading

| Example | Response |
|---|---|
| R | Wait 900ms<br>1 Dec<br>9.560 ASCII<br>0 Null |



# Calibration

900ms processing delay

Issuing the cal,mid command after the EZO™ pH circuit has been calibrated, will clear the other calibration points. Full calibration will have to be redone.

## Command syntax

| | |
|---|---|
| Cal,mid,n | single point calibration at midpoint |
| Cal,low,n | two point calibration at lowpoint |
| Cal,high,n | three point calibration at highpoint |
| Cal,clear | delete calibration data |
| Cal,? | device calibrated? |

| Example | Response |
|---|---|
| Cal,mid,7.00 | Wait 900ms → 1 (Dec) 0 (Null) |
| Cal,low,4.00 | Wait 900ms → 1 (Dec) 0 (Null) |
| Cal,high,10.00 | Wait 900ms → 1 (Dec) 0 (Null) |
| Cal,clear | Wait 300ms → 1 (Dec) 0 (Null) |
| Cal,? | Wait 300ms → 1 (Dec) ?Cal,0 (ASCII) 0 (Null) or 1 (Dec) ?Cal,1 (ASCII one point) 0 (Null)<br>1 (Dec) ?Cal,2 (ASCII two point) 0 (Null) or 1 (Dec) ?Cal,3 (ASCII three point) 0 (Null) |




Atlas Scientific™ Environmental Robotics

# Export calibration

300ms processing delay

## Command syntax

Export: Use this command to download calibration settings

| | |
|---|---|
| Export,? | calibration string info |
| Export | export calibration string from calibrated device |

## Example

## Response

### Export,?


| 1 | 10,120 | 0 |
|---|---|---|
| Dec | ASCII | Null |

**Response breakdown**

10, 120

# of strings to export — # of bytes to export

Export strings can be up to 12 characters long

### Export


| 1 | 59 6F 75 20 61 72 | 0 |
|---|---|---|
| Dec | ASCII | Null |

(1 of 10)

### Export


| 1 | 65 20 61 20 63 6F | 0 |
|---|---|---|
| Dec | ASCII | Null |

(2 of 10)

### (7 more)

### Export


| 1 | 6F 6C 20 67 75 79 | 0 |
|---|---|---|
| Dec | ASCII | Null |

(10 of 10)

### Export


| 1 | *DONE | 0 |
|---|---|---|
| Dec | ASCII | Null |




Atlas Scientific Environmental Robotics

# Import calibration

300ms processing delay

## Command syntax

Import: Use this command to upload calibration settings to one or more devices.

Import,n  import calibration string to new device

## Example

Import, 59 6F 75 20 61 72  (1 of 10)

Import, 65 20 61 20 63 6F  (2 of 10)

⋮

Import, 6F 6C 20 67 75 79  (10 of 10)

## Response

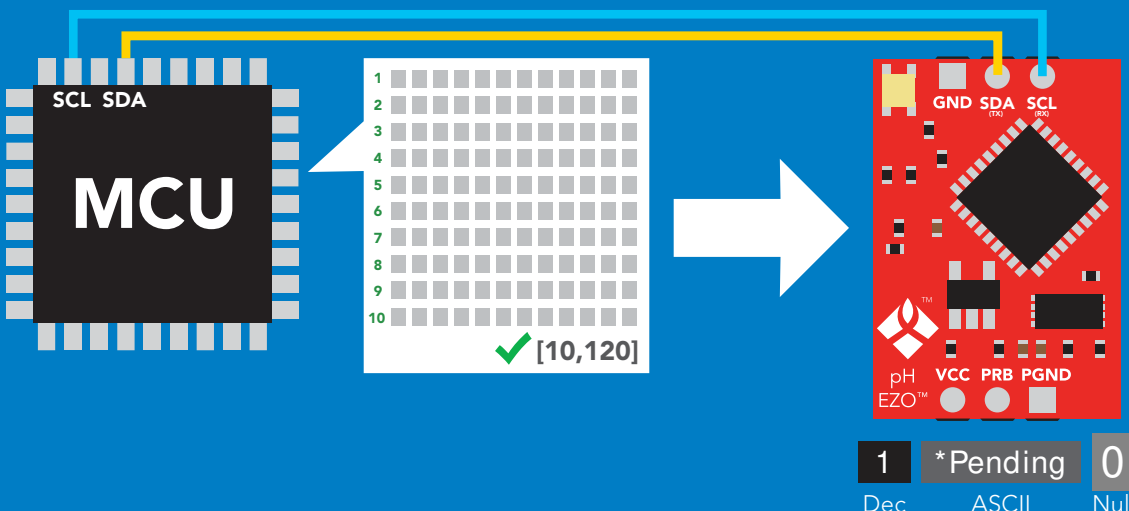
Wait 300ms  1 Dec  0 Null

Wait 300ms  1 Dec  0 Null

⋮

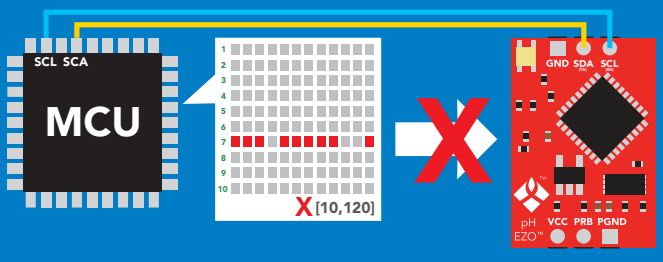
Wait 300ms  1 Dec  0 Null

Import,n




1 Dec  *Pending ASCII  0 Null

system will reboot




reboot

* If one of the imported strings is not correctly entered, the device will not accept the import and reboot.




Atlas Scientific Environmental Robotics

# Slope

300ms processing delay

After calibrating a pH probe issuing the slope command will show how closely (in percentage) the calibrated pH probe is working compared to the "ideal" pH probe.

## Command syntax

Slope,? returns the slope of the pH probe

## Example

Slope,?

## Response

Wait 300ms

| 1 | ?Slope,99.7,100.3, -0.89 | 0 |
|---|---|---|
| Dec | ASCII | Null |

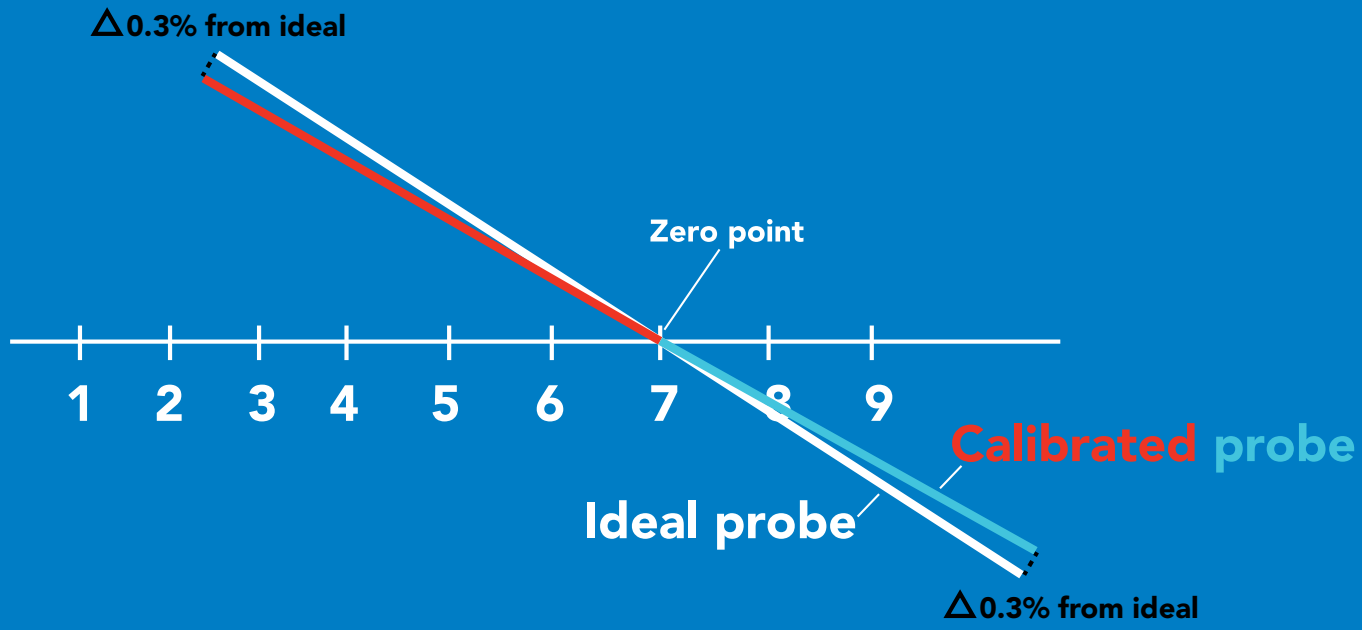
## Response breakdown

?Slope, 99.7 100.3 -0.89

99.7% is how closely the slope of the acid calibration line matched the "ideal" pH probe.

100.3% is how closely the slope of the base calibration matches the "ideal" pH probe.

This is how many millivolts the zero point is off from true 0.







Atlas Scientific Environmental Robotics

# Extended pH scale

300ms processing delay

Very strong acids and basses can exceed the traditional pH scale. This command extends the pH scale to show below 0 and above 14.

Lowest possible reading: -1.6
Highest possible reading: 15.6

## Command syntax

| | | |
|---|---|---|
| pHext,0 | extended pH scale off (0–14) | default |
| pHext,1 | extended pH scale on (-1.6–15.6) | |
| pHext,? | extended pH scale on/off? | |

| Example | Response |
|---|---|
| pHext,1 | Wait 300ms · 1 (Dec) · 0 (Null) |
| pHext,0 | Wait 300ms · 1 (Dec) · 0 (Null) |
| pHext,? | Wait 300ms · 1 (Dec) · ?pHext,1 (ASCII) · 0 (Null) or 1 (Dec) · ?pHext,0 (ASCII) · 0 (Null) |

pHext,0


pH = 0.000

pHext,1


pH = -1.220




Atlas Scientific Environmental Robotics

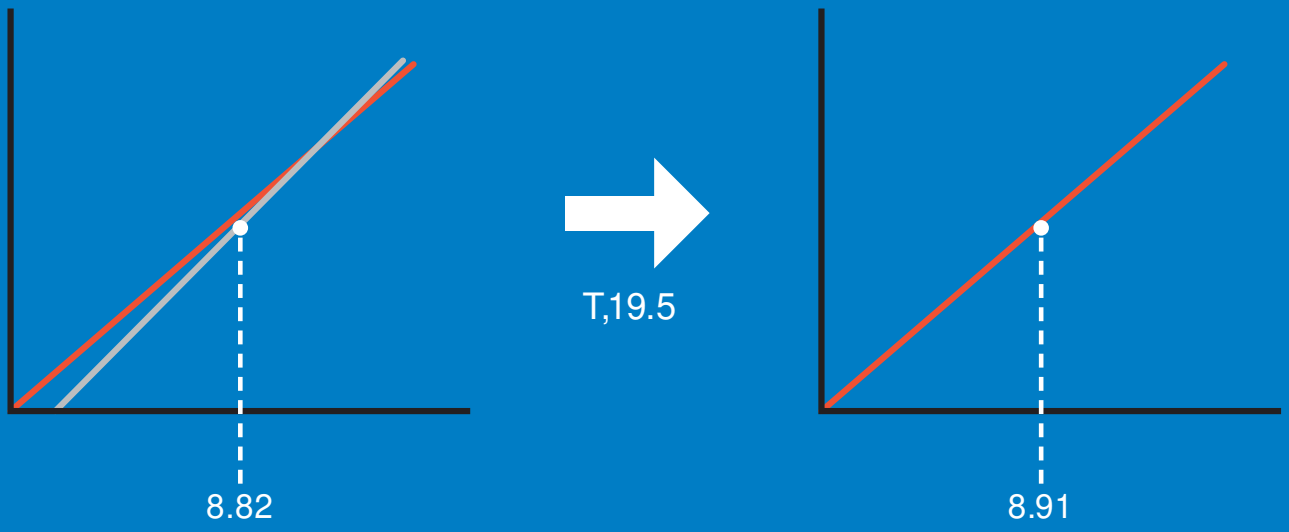
# Temperature compensation

## Command syntax

Default temperature = 25°C
Temperature is always in Celsius
Temperature is not retained if power is cut

| | | |
|---|---|---|
| T,n | n = any value; floating point or int | 300ms processing delay |
| T,? | compensated temperature value? | |
| RT,n | set temperature compensation and take a reading | |

| Example | Response |
|---|---|
| T,19.5 | Wait 300ms · 1 Dec · 0 Null |
| RT,19.5 | Wait 900ms · 1 Dec · 8.91 ASCII · 0 Null |
| T,? | Wait 300ms · 1 Dec · ?T,19.5 ASCII · 0 Null |



# Naming device

300ms processing delay

## Command syntax

Do not use spaces in the name

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

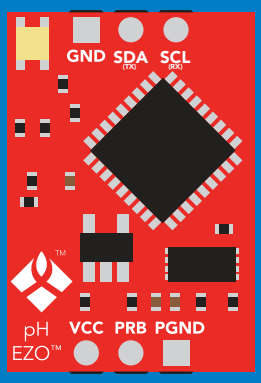
n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

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

Up to 16 ASCII characters

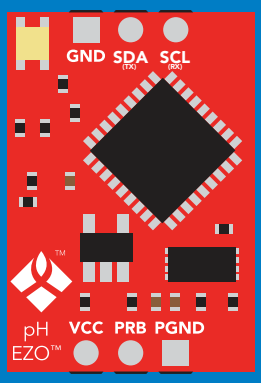
| Example | Response |
|---|---|
| Name, | Wait 300ms → 1 (Dec) 0 (Null) — name has been cleared |
| Name,zzt | Wait 300ms → 1 (Dec) 0 (Null) |
| Name,? | Wait 300ms → 1 (Dec) ?Name,zzt (ASCII) 0 (Null) |

Name,zzt




1 0

Name,?




1 ?Name,zzt 0




Atlas Scientific Environmental Robotics

# Device information

## Command syntax

300ms processing delay

i device information

## Example

i

## Response


1 Dec

?i,pH,1.98 ASCII

0 Null

## Response breakdown

?i, pH, 1.98

pH, Device

1.98 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


1 Dec

?Status,P,5.038 ASCII

0 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 | 16 mA | 1.16 mA |
| 3.3V | 13.9 mA | 0.995 mA |


Standby

Sleep




Atlas Scientific Environmental Robotics

# Protocol lock

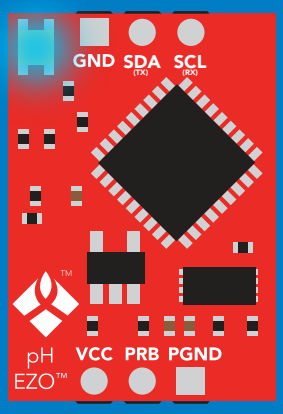
## Command syntax

300ms processing delay

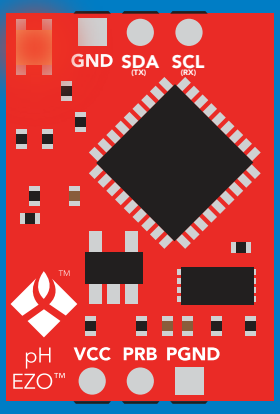
| | | |
|---|---|---|
| Plock,1 | enable Plock | Locks device to I²C mode. |
| Plock,0 | disable Plock | default |
| Plock,? | Plock on/off? | |

| 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) |

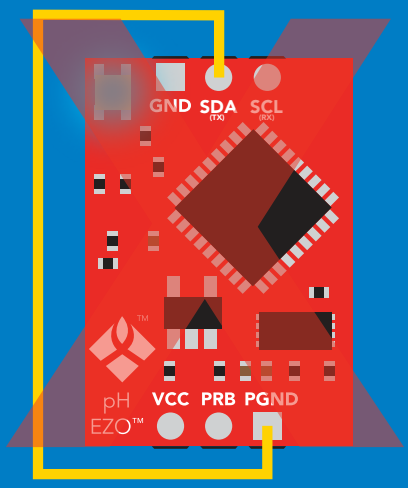
Plock,1




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,100 | device reboot<br>(no response given) |

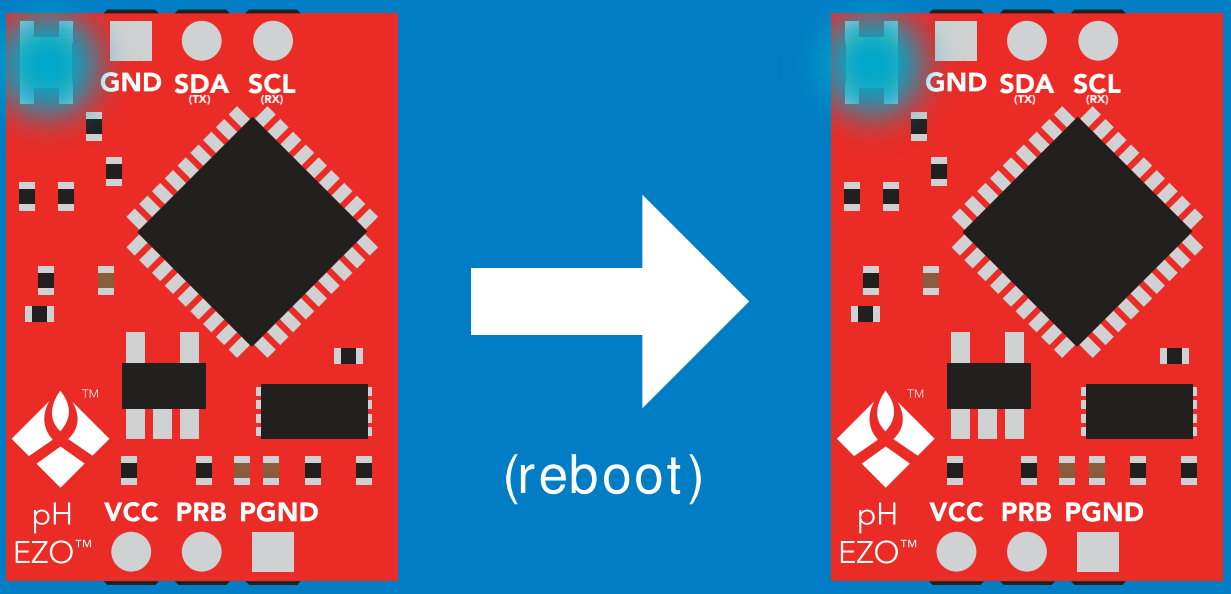
### Warning!

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

Default I²C address is 99 (0x63).

n = any number 1 – 127

I2C,100







Atlas Scientific Environmental Robotics

# Factory reset

## Command syntax

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

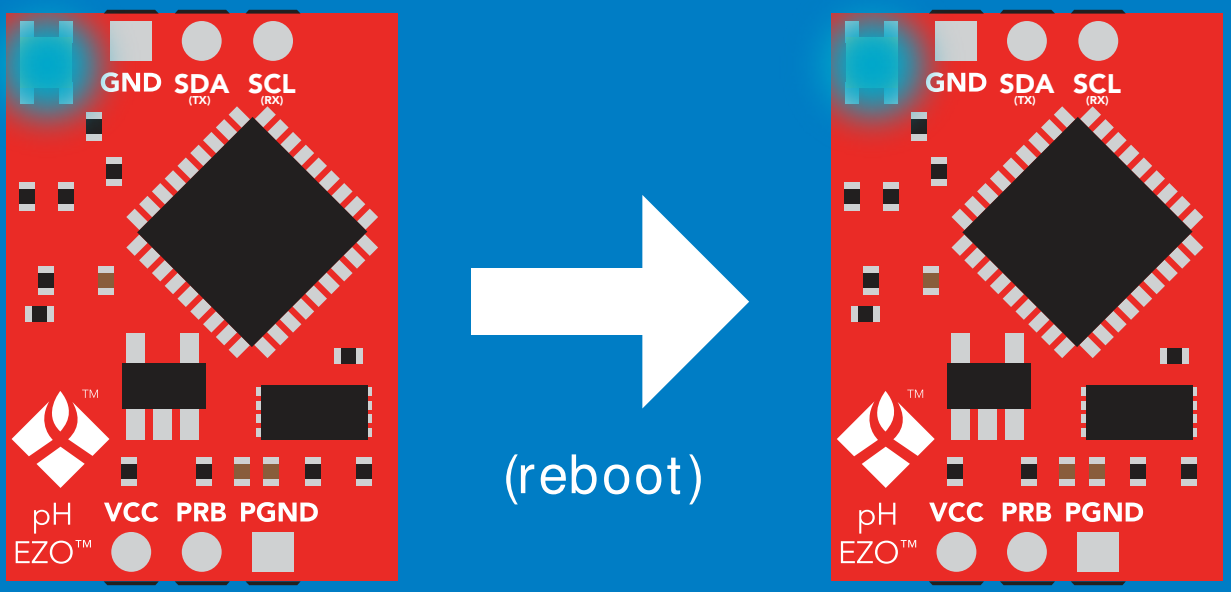
Factory enable factory reset

I²C address will not change

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

Clears calibration
LED on
Response codes enabled

### Factory



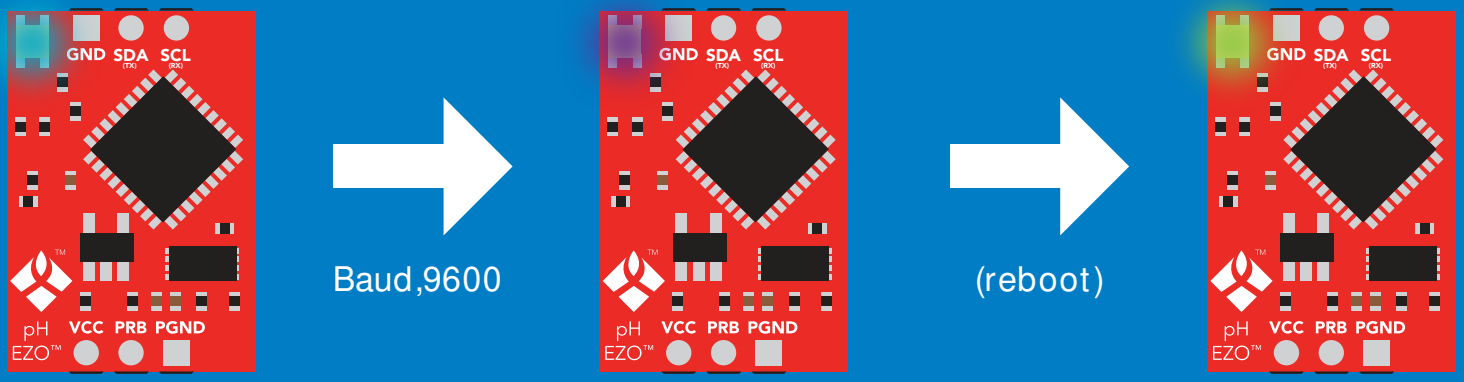
# Change to UART mode

## 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

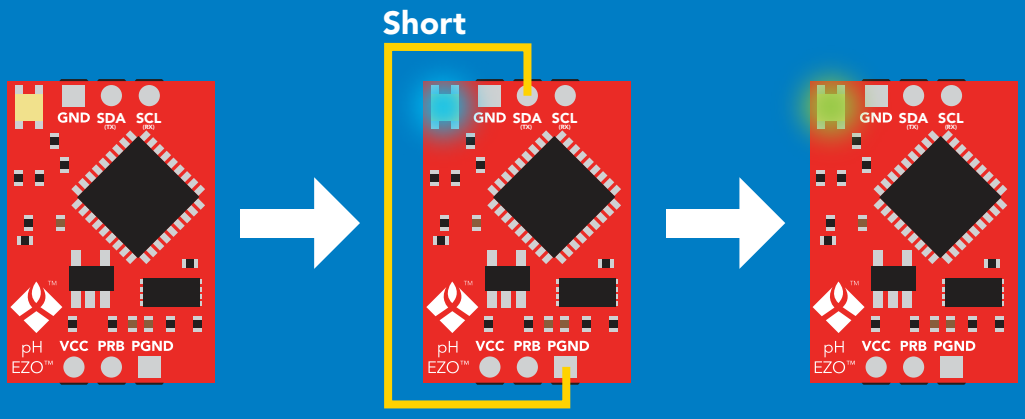



Changing to UART mode

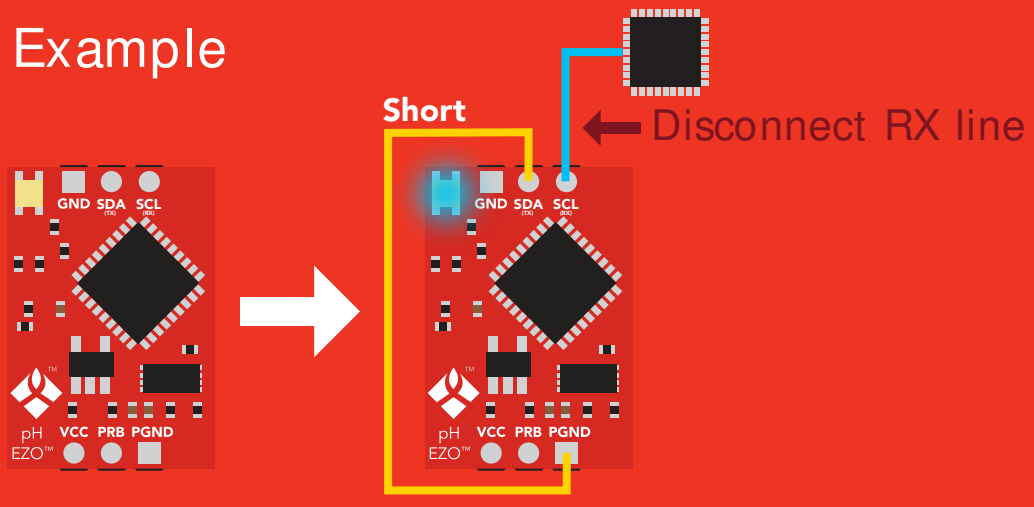
# Manual switching to UART

- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to PGND
- 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




## Wrong Example

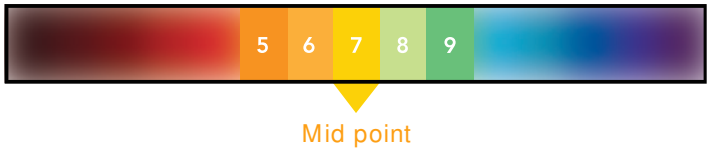


# Calibration theory

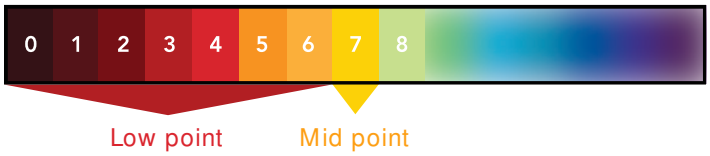
The accuracy of your readings is directly related to the quality of your calibration.
*(Calibration is not difficult, and a little bit of care goes a long way).*

## Single, Two point, or Three point calibration accuracy

Single point calibration

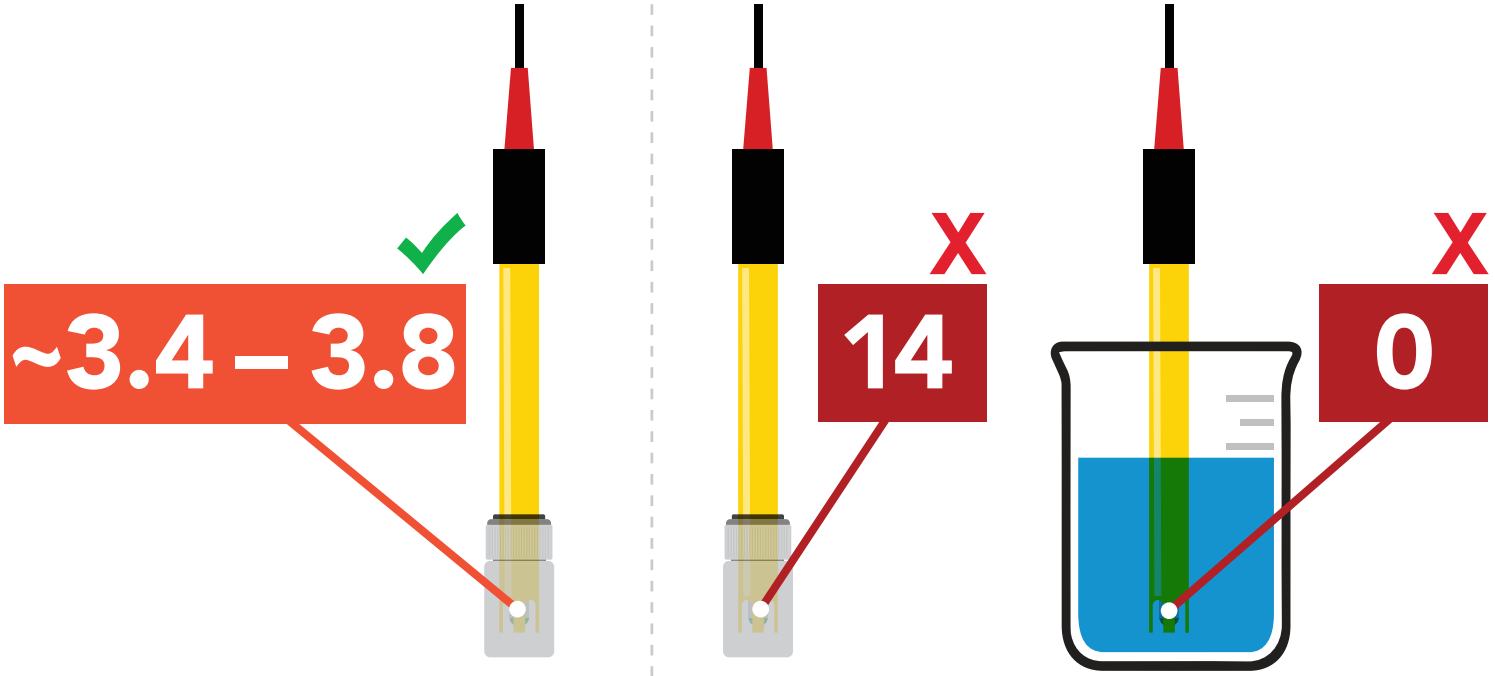



Two point calibration




Three point calibration


## Confirm the pH probe is working correctly




A new Atlas Scientific pH probe, still in its soaker bottle will read a pH of ***~3.4 – 3.8***

If your pH probe gives a reading of ***zero***, ***seven*** or ***14*** continuously and that reading cannot be changed no matter what solution the probe is in, your probe cannot be calibrated and may be damaged.

*Contact Atlas Scientific customer support for assistance.*




AtlasScientific™ Environmental Robotics

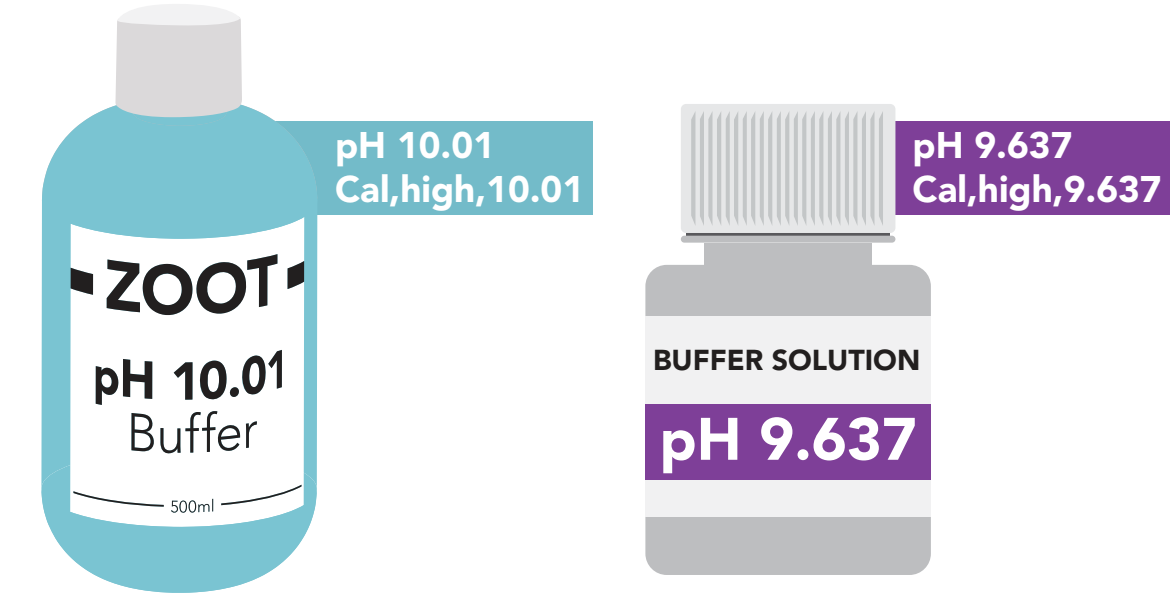
# Calibration order

If this is your first time calibrating the EZO™ pH circuit, we recommend that you follow this calibration order.


1 **Mid point** 2 **Low point** 3 **High point**

## Calibration solutions

The Atlas Scientific EZO™ pH circuit can work with any brand or value of calibration solution. ***We recommend using calibration solutions that have simple values.***




✓ **Simple value** X **Complex value**

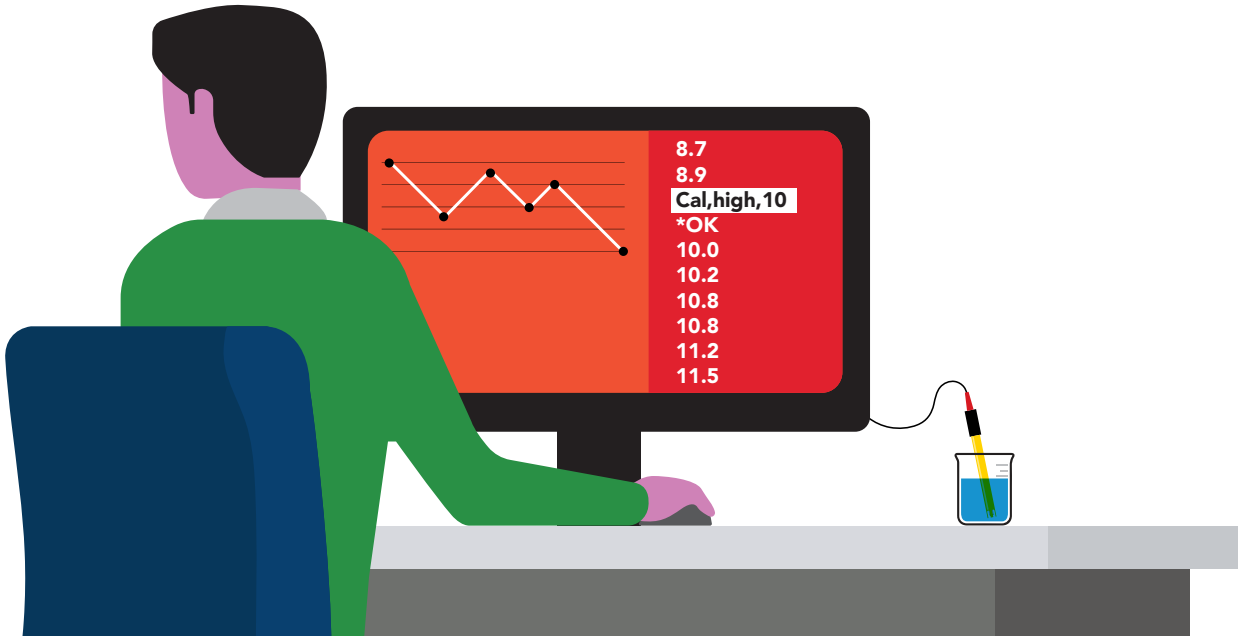
While you can use calibration solutions that have complex values, we recommend avoiding unnecessary complexity. ***Unusually specific calibration values should be treated with suspicion.***

# Best practices for calibration

Always watch the readings throughout the calibration process.
Issue calibration commands once the readings have stabilized.


## ⚠ Never do a blind calibration! ⚠

Issuing a calibration command before the readings stabilize will result in drifting readings.



# Best practices for calibration

## Avoid extended stabilization time.



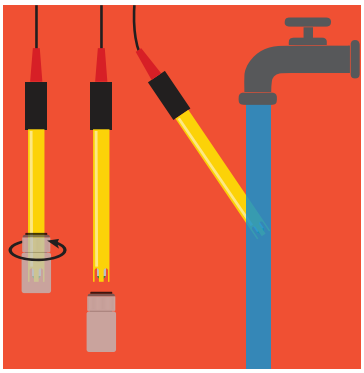
Letting the probes pre-calibrtion readings stabilize over an extended period will cause your calibrated readings to take a long time to stabilize.

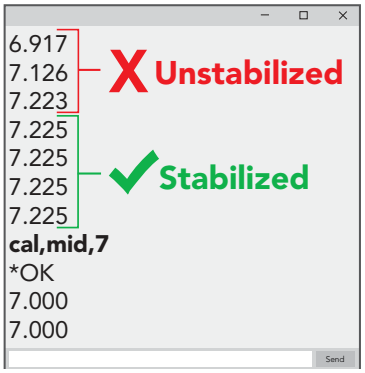
## Avoid frequent recalibrations.

**if it ain't broke, don't fix it.**

pH probes loose accuracy slowly. Frequent recalibrations to insure high accuracy will often have the opposite effect. It is far more llikly that you will misscalibrate the probe rather then improve its accuracy.

# 1. Mid point calibration

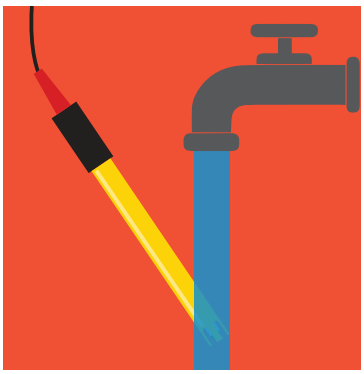


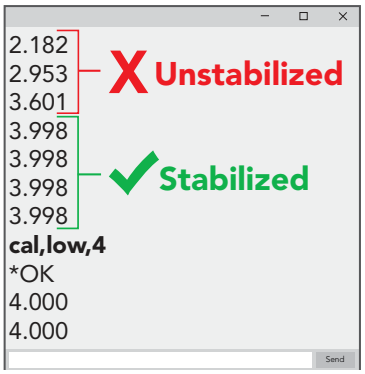





Mid point calibrated

# 2. Low point calibration

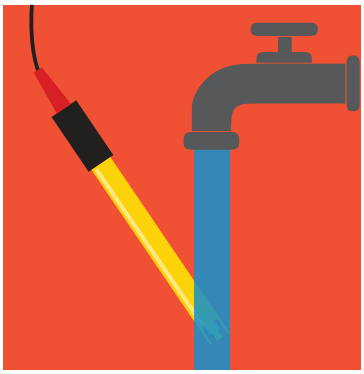


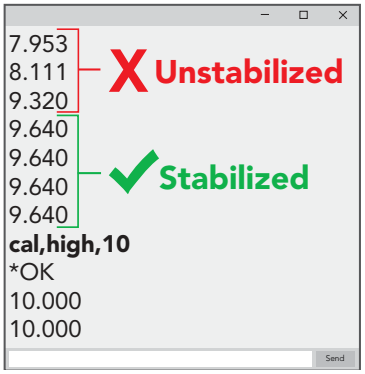





Low point calibrated

# 3. High point calibration





High point calibrated

## Optional steps:

Confirm your calibration accuracy using the slope command.
Recalibre a single point if required.

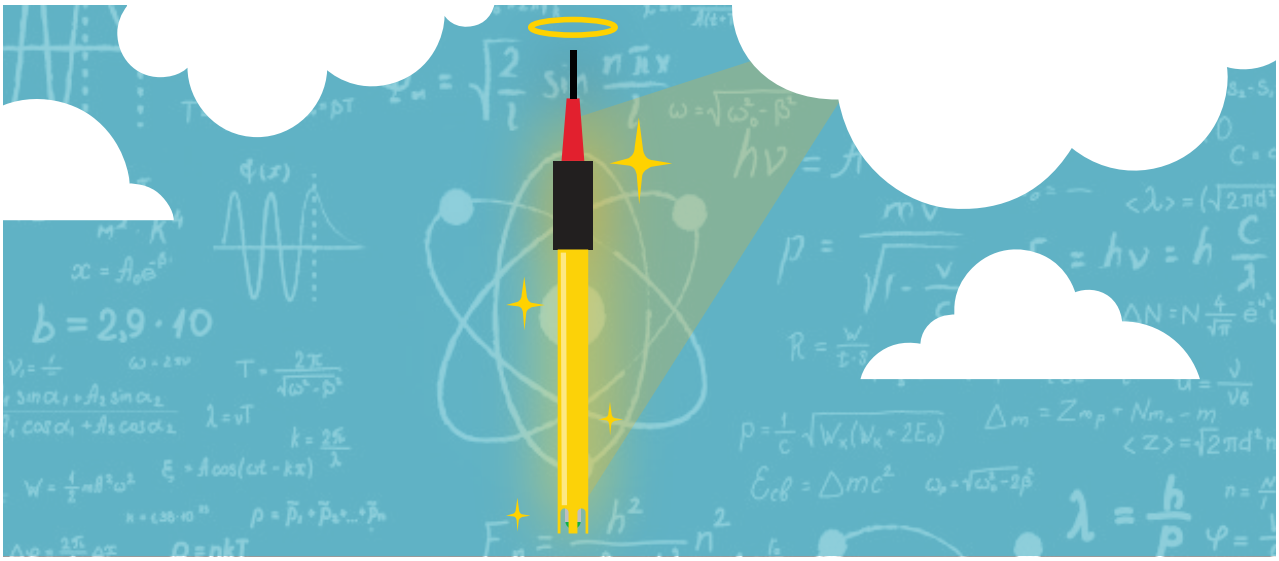
Atlas Scientific™
Environmental Robotics

# Understanding pH slope

The slope function is a powerful tool used to verify calibration and determine the overall health of a pH probe. By evaluating the slope of a pH probe's response curve, you can determine how well a pH probe was calibrated or when that probe is reaching end of life.

**Slope and calibration are directly related. The slope is updated when a calibration command is given. The slope does not update automatically.**

Generally speaking, all pH probes behave the same way. This means a probe's response to calibration can be compared to a simulated pH probe that is mathematically perfect in all ways.



The slope is broken into three sections; acid, base, and neutral.
Each section is evaluated separately.

Acid (pH 1–6.9)
Base (pH 7.1–14)
Neutral (pH 7)

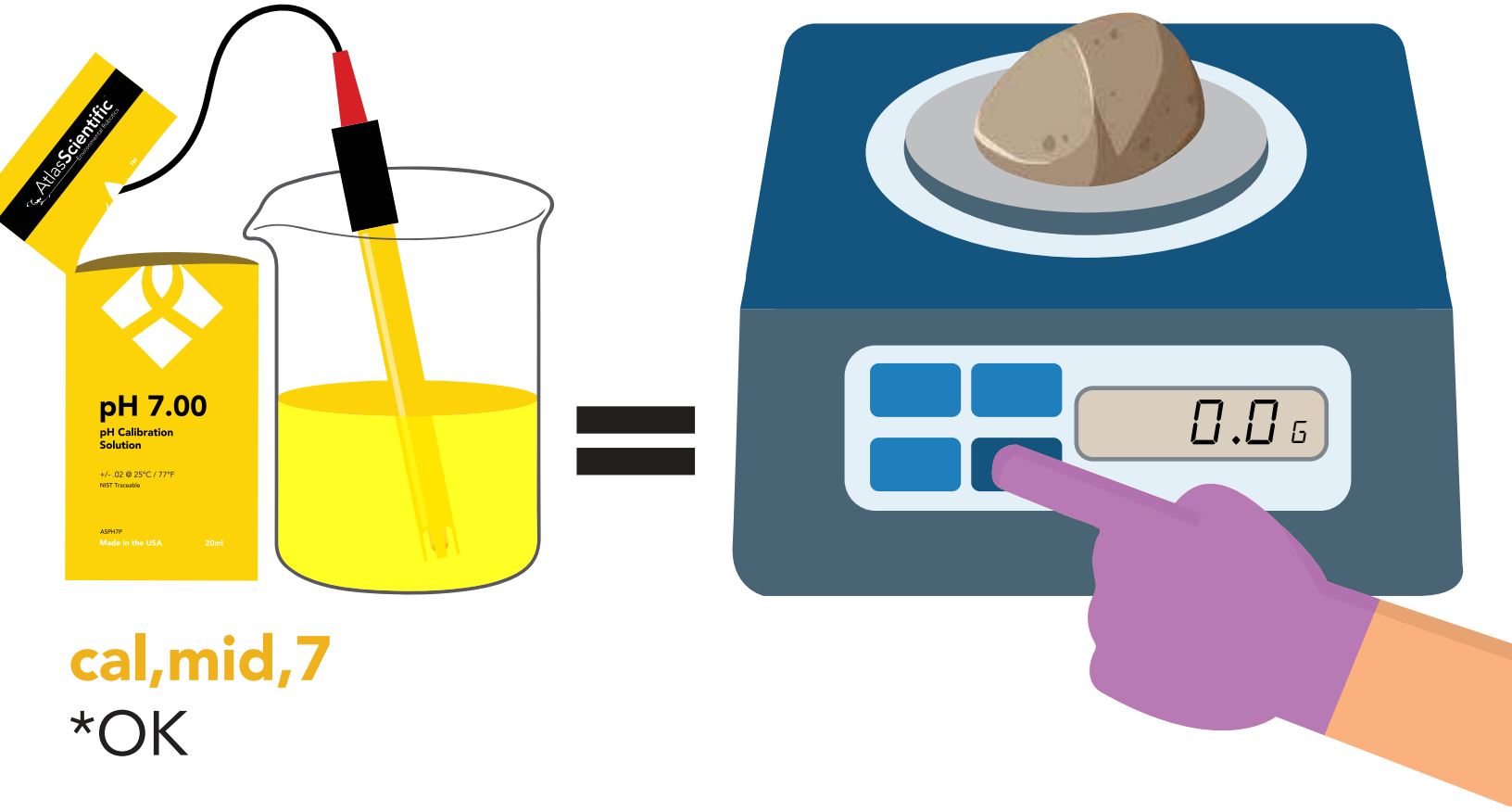
An uncalibrated pH probe will have a mathematically perfect slope. Because no pH probe is mathematically perfect, the slope can be used to determine if the pH was calibrated.

**Uncalibrated slope:** 100, 100, 0 (acid, base, neutral)
% % mV

The first two numbers are percentages, and the third is millivolts. The slope shows that the probe's response to acid and base is 100% correct, and it detects 0 mv in a pH 7. Because such perfection does not exist in the real world, we know this probe was not calibrated.

# Understanding pH slope

pH 7 is the absence of pH; it is not an acid or a base. Therefore it should always be your first calibration point. It is equivalent to the tare function on a scale because it establishes the probe's zero point.




**cal,mid,7**

*OK

After pH 7 calibration, use the slope command to see how the probe performed during calibration.

## The slope after pH 7 calibration: 100, 100, -1.2

Here we see the probe reads -1.2mV in pH 7. The closer this number is to 0, the better. A new pH probe should give a millivolt offset no greater than -5mV to 5mV. Over time this number's distance to 0 may increase; the larger the number, the lower the accuracy. A reading >10mV will result in noticeable performance issues.

It is important to remember that a high number is not definitive evidence that the probe is inaccurate or malfunctioning. It is very common to see a high number if the calibration solution was contaminated and not actually its stated value.

# Understanding pH slope

The next two calibration points *(pH 4 and pH 10)* report their slope in percentage. A new pH probe should have a slope of >95%.

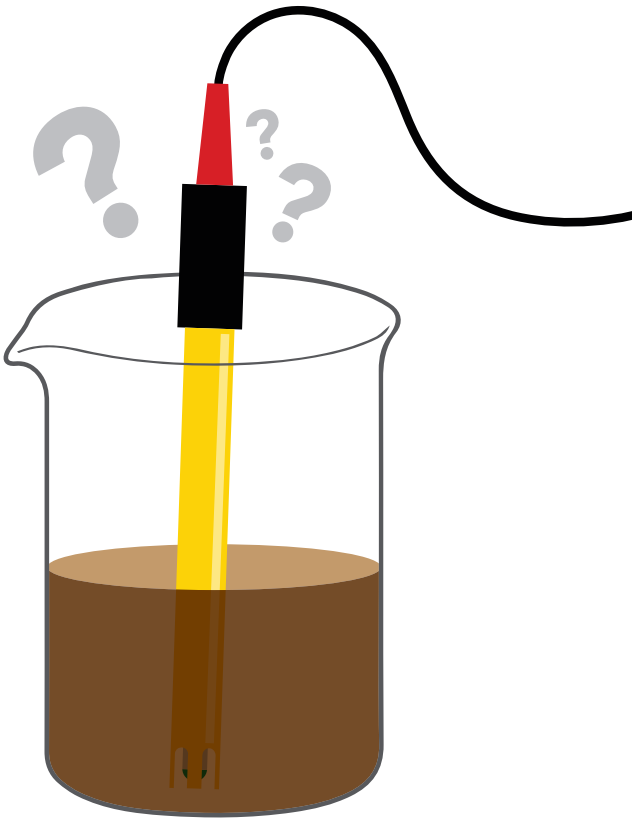
**The slope after pH 4 calibration:** 98.2, 100, -1.2

**The slope after pH 10 calibration:** 98.2, 97.8, -1.2

## Tips:

Throughout this explanation, we have looked at the slope after each calibration event. This is unnecessary; in reality, it is best to fully calibrate the probe and look at the slope once calibration has been completed.

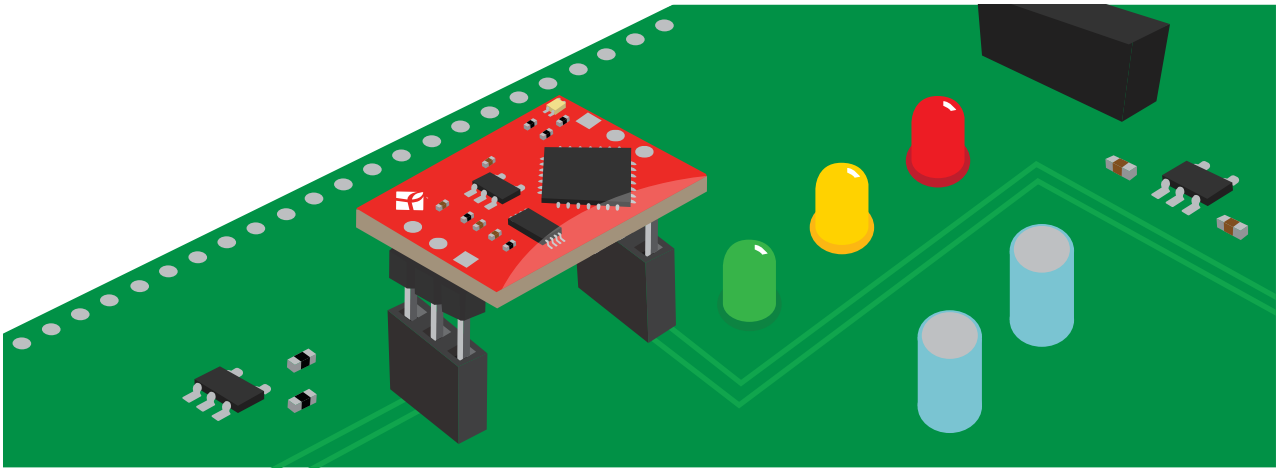
To gain a deeper understanding of how slope affects the stability and accuracy of a pH probe, intentionally miscalibrate the probe and see how it affects the slope.

# Soldering

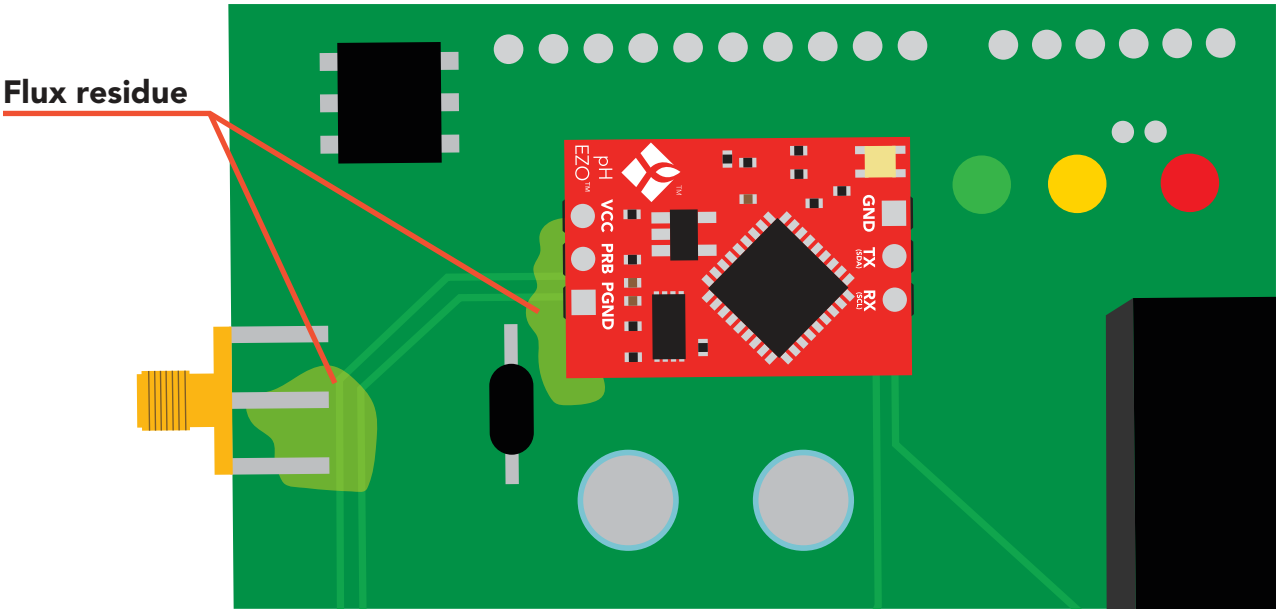
**Do not directly solder an EZO circuit to your PCB.** If something goes wrong during the soldering process it may become impossible to correct the problem. It is simply not worth the risk.

Instead, solder female header pins to your PCB and place the EZO device in the female headers.



Avoid using rosin core solder.
Use as little flux as possible.

**Flux residue will severely affect your readings.** Any Flux residue that comes in contact with the PRB pins or your probes connector will cause a "flux short".

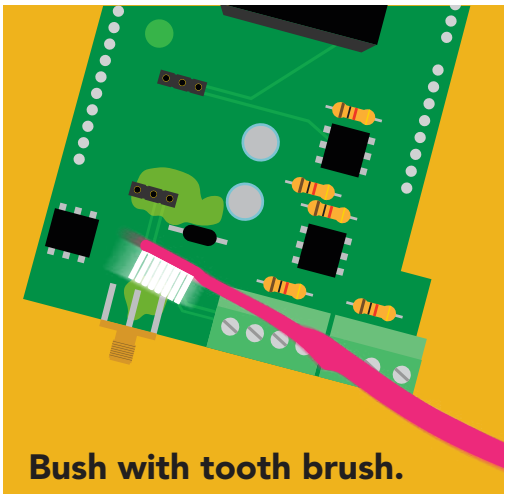



You ***MUST*** remove all the flux residue from your PCB after soldering.

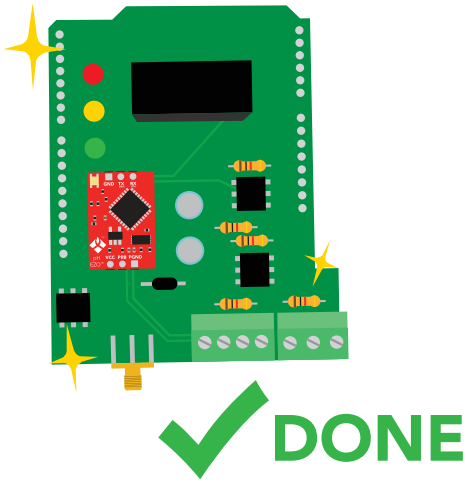
# Soldering

Removing flux residue can be done with commercially available products such as flux off or you can use alcohol and a tooth brush.





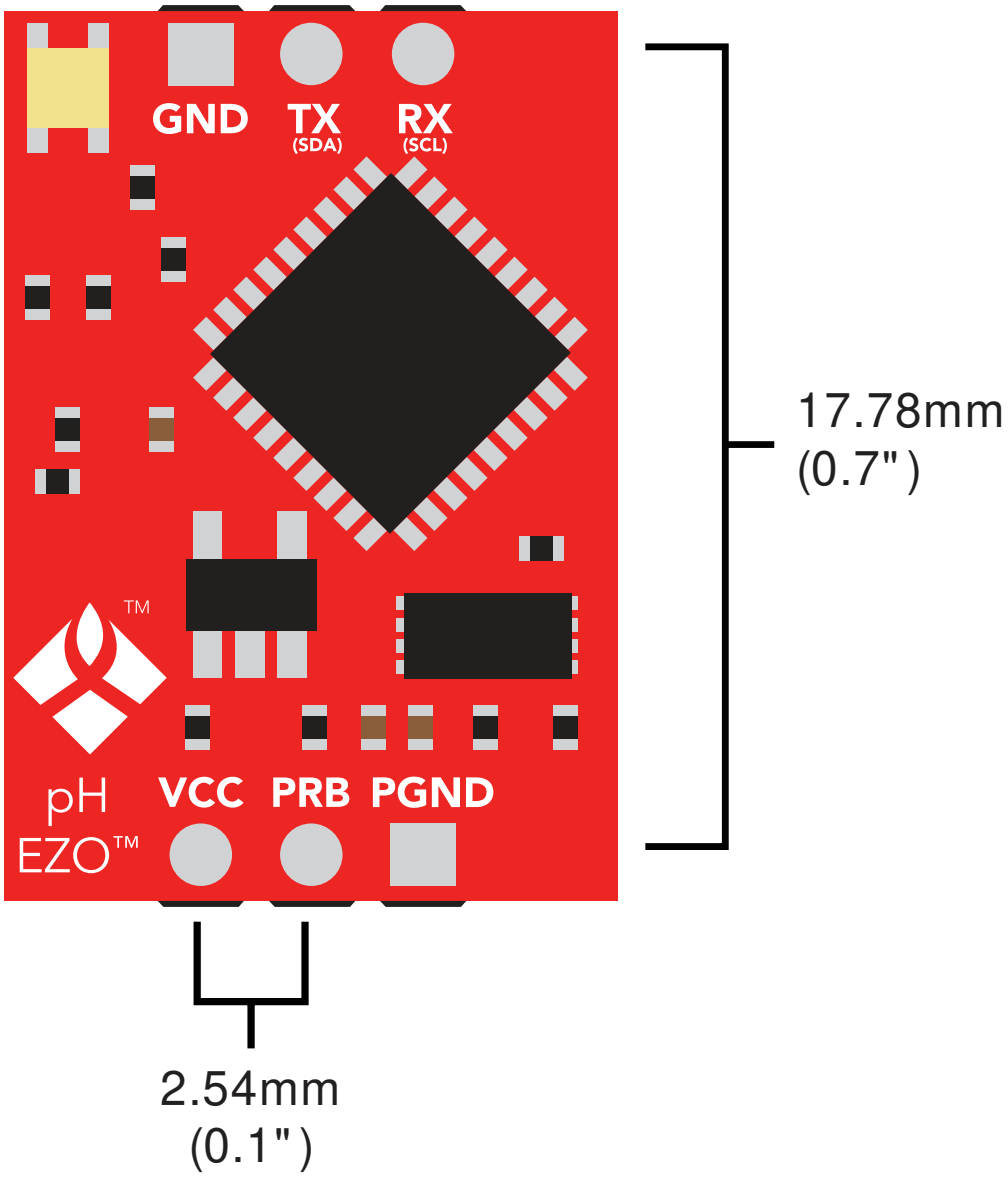


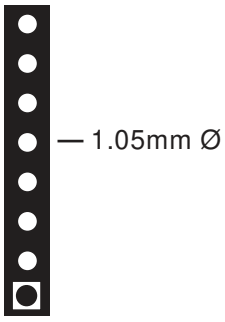

## What does a flux short look like?

1: Readings move slowly and take serval minutes to reach the correct value.
2: Readings are pinned to 0, 7 or 14.

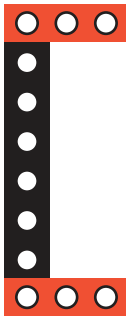
# EZO™ circuit footprint



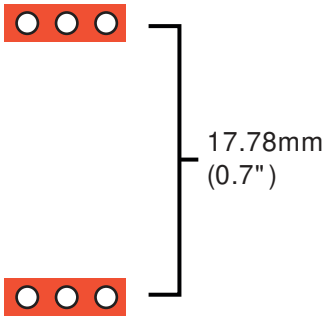

1 In your CAD software place a 8 position header.




2 Place a 3 position header at both top and bottom of the 8 position.



3 Delete the 8 position header. The two 3 position headers are now 17.78mm (0.7") apart from each other.



# Datasheet change log

## Datasheet V 6.1

Revised electrical isolation section on page 7.

## Datasheet V 6.0

Revised entire document.

## Datasheet V 5.9

Revised naming device info on pages 32 & 58.

## Datasheet V 5.8

Revised calibration info and art on pages 11 & 12.

## Datasheet V 5.7

Added new command:
"Extended pH Scale" pages 30 (UART) & 56 ($I^2C$).

## Datasheet V 5.6

Revised information on the slope command found on pages 29 & 54.

## Datasheet V 5.5

Revised artwork within datasheet.

## Datasheet V 5.4

Moved the Default state to pg 14.

## Datasheet V 5.3

Revised response for the sleep command in UART mode on pg 35.

## Datasheet V 5.2

Revised calibration theory on page 11, and added more information on the Export calibration and Import calibration commands.

## Datasheet V 5.1

Revised isolation schematic on pg 10.

## Datasheet V 5.0

Added more information about temperature compensation on pages 29 & 53.

## Datasheet V 4.9

Changed "Max rate" to "Response time" on cover page.

## Datasheet V 4.8

Added new command:
"RT,n" for Temperature compensation located on pages 29 (UART) & 53 ($I^2C$).
Added firmware information to Firmware update list.

## Datasheet V 4.7

Removed note from certain commands about firmware version.

## Datasheet V 4.6

Added information to calibration theory on pg 7.

## Datasheet V 4.5

Revised definition of response codes on pg 44.

## Datasheet V 4.4

Added resolution range to cover page.

## Datasheet V 4.3

Revised isolation information on pg 9.

## Datasheet V 4.2

Revised Plock pages to show default value.

## Datasheet V 4.1

Added new commands:
"Find" pages 23 (UART) & 46 ($I^2C$).
"Export/Import calibration" pages 27 (UART) & 49 ($I^2C$).
Added new feature to continous mode "C,n" pg 24.

## Datasheet V 4.0

Added accuracy range on cover page, and revised isolation info on pg. 10.

## Datasheet V 3.9

Revised calibration theory on pg. 7.

## Datasheet V 3.8

Revised entire datasheet.

# Firmware updates

V1.5 – Baud rate change (Nov 6, 2014)
- Change default baud rate to 9600

V1.6 – I²C bug (Dec 1, 2014)
- Fixed I²C bug where the circuit may inappropriately respond when other I²C devices are connected.

V1.7 – Factory (April 14, 2015)
- Changed "X" command to "Factory"

V1.95 – Plock (March 31, 2016)
- Added protocol lock feature "Plock"

V1.96 – EEPROM (April 26, 2016)
- Fixed bug where EEPROM would get erased if the circuit lost power 900ms into startup

V1.97 – EEPROM (Oct 10, 2016)
- Added the option to save and load calibration.

V1.98 – EEPROM (Nov 14, 2016)
- Fixed bug during calibration process.

V2.10 – (May 9, 2017)
- Added "Find" command.
- Added "Export/import" command.
- Modified continuous mode to be able to send readings every "n" seconds.

V2.11 – (June 12, 2017)
- Fixed "I" command to return "pH" instead of "PH".

V2.12 – (April 16, 2018)
- Fixed "cal,clear" was not clearing stored calibration in EEPROM.
- Added "RT" command to Temperature compensation.

V2.13 – (June 25, 2019)
- Added calibration offset to slope.
- Added calibration with temperature compensation.

V2.14 – (June 10, 2020)
- Added extended pH scale.

v2.15 – (Nov 3, 2021)
- Internal update for new part compatibility.

v2.16 – (Nov 19, 2021)
- Fixed bug in I2C mode with timing and sleep mode.

# Warranty

Atlas Scientific™ Warranties the EZO™ class pH circuit to be free of defect during the debugging phase of device implementation, or 30 days after receiving the EZO™class pH circuit (which ever comes first).

# The debugging phase

The debugging phase as defined by Atlas Scientific™ is the time period when the EZO™ class pH circuit is inserted into a bread board, or shield. If the EZO™ class pH circuit is being debugged in a bread board, the bread board must be devoid of other components. If the EZO™ class pH circuit is being connected to a microcontroller, the microcontroller must be running code that has been designed to drive the EZO™ class pH circuit exclusively and output the EZO™ class pH circuit 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™ class pH circuit warranty:

- Soldering any part of the EZO™ class pH circuit.
- Running any code, that does not exclusively drive the EZO™ class pH circuit and output its data in a serial string.
- Embedding the EZO™ class pH circuit 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™ class pH circuit, against the thousands of possible variables that may cause the EZO™ class pH circuit 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™ class pH circuits continued operation. This is because that would be equivalent to Atlas Scientific™ taking responsibility over the correct operation of your entire device.