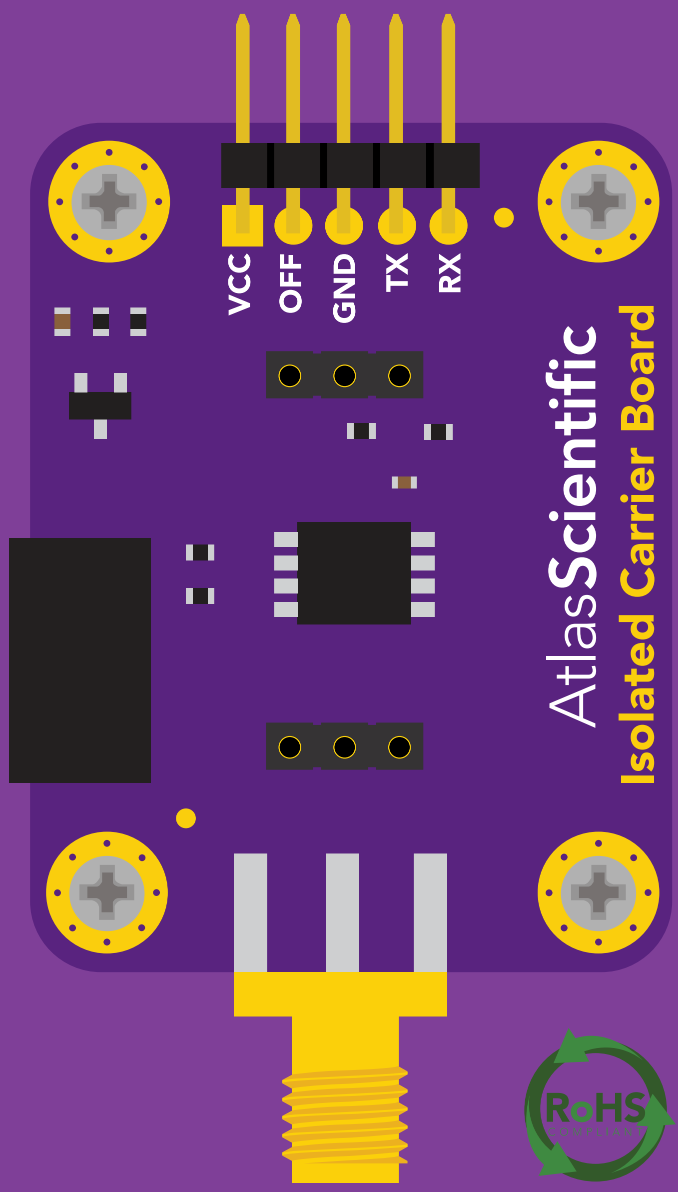
AtlasScientific™
Environmental Robotics
V 1.5

Electrically Isolated

EZO™ Carrier Board Gen 2

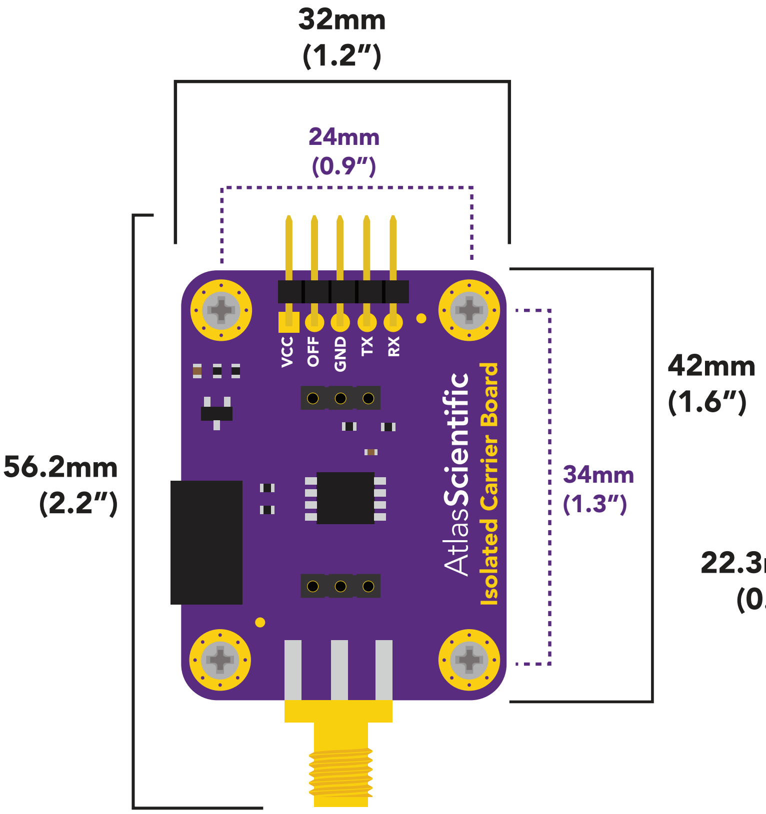
Data input	**UART or I^2C**
Voltage input	**3.0V – 5.0V**
Probe connector	**Female SMA**
Current consumption	5V **28 mA** 3.3V **22 mA**
Power saving mode *(OFF pin)*	**2.6mA**

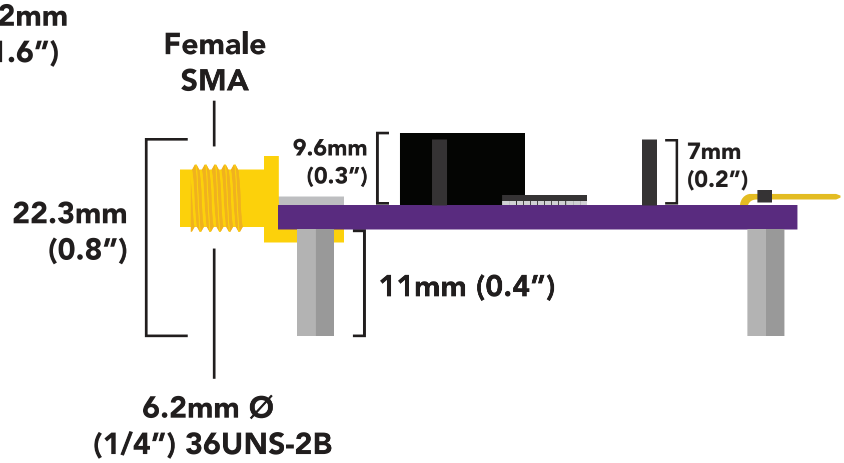


Written by Jordan Press
Designed by Noah Press

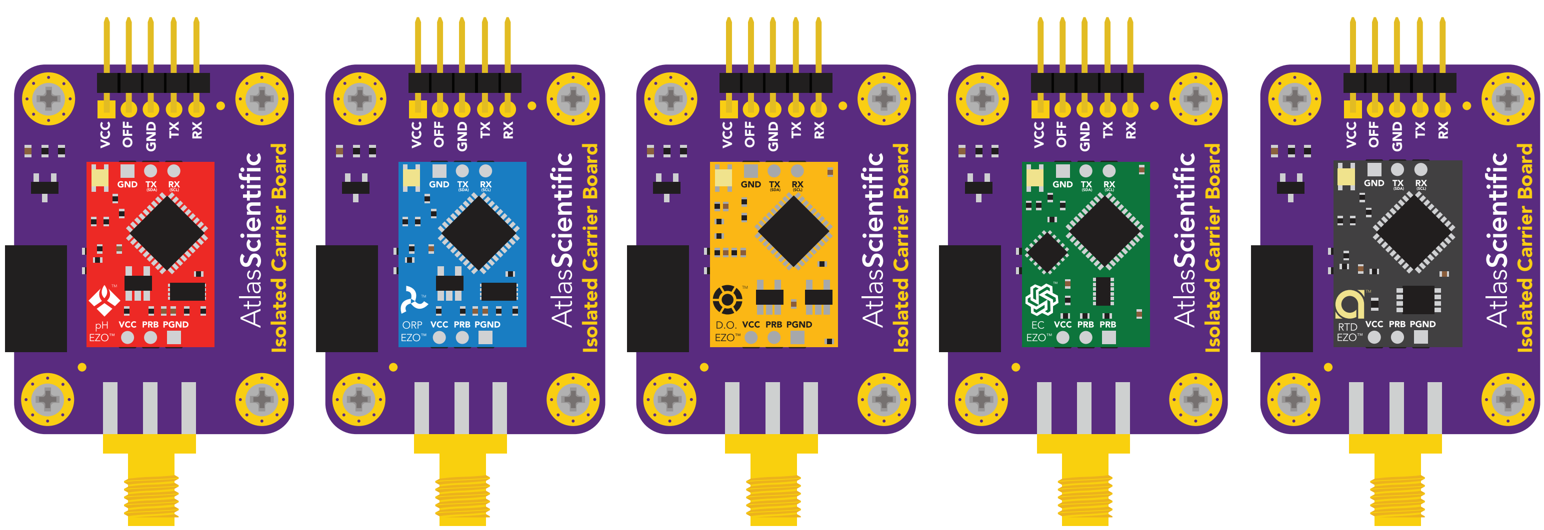
This is an evolving document, check back for updates.

Carrier board dimensions



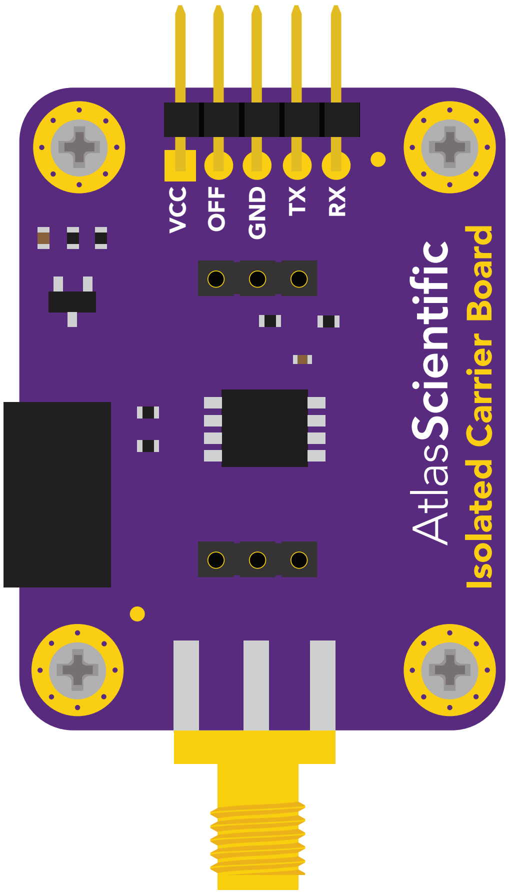


The Electrically Isolated EZO™ Carrier Board works with almost all EZO™ circuits, except the EZO™ Embedded Flow Meter Totalizer.

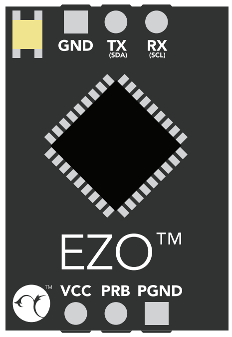


The Electrically Isolated EZO™ Carrier Board does not come with EZO™ class devices.

Current consumption



+



= 28 + XmA

i = XmA

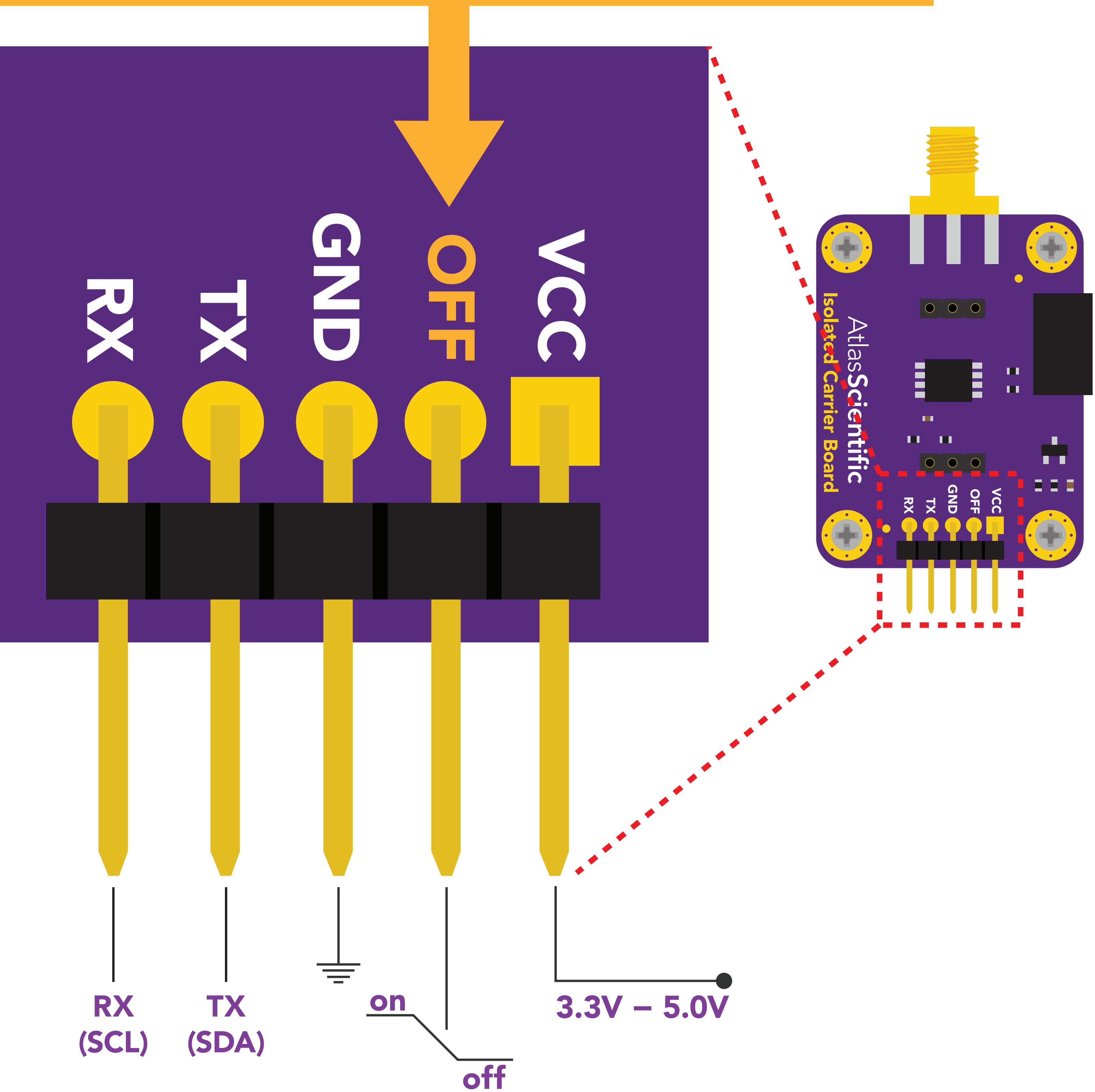
i = 28mA

The current consumption for the Electrically Isolated EZO™ Carrier Board is linear. The table below shows how much current will be consumed when an EZO™ circuit is connected to the Electrically Isolated EZO™ Carrier Board.

		5V	3.3V
Electrically Isolated EZO™ Carrier Board	No Load	28mA	22mA
	EZO™ pH	44mA	35mA
	EZO™ ORP	44mA	35mA
	EZO™ Dissolved oxygen	44mA	35mA
	EZO™ Conductivity *(no probe)*	55mA	43mA
	EZO™ RTD Temperature	44mA	35mA

Pin out

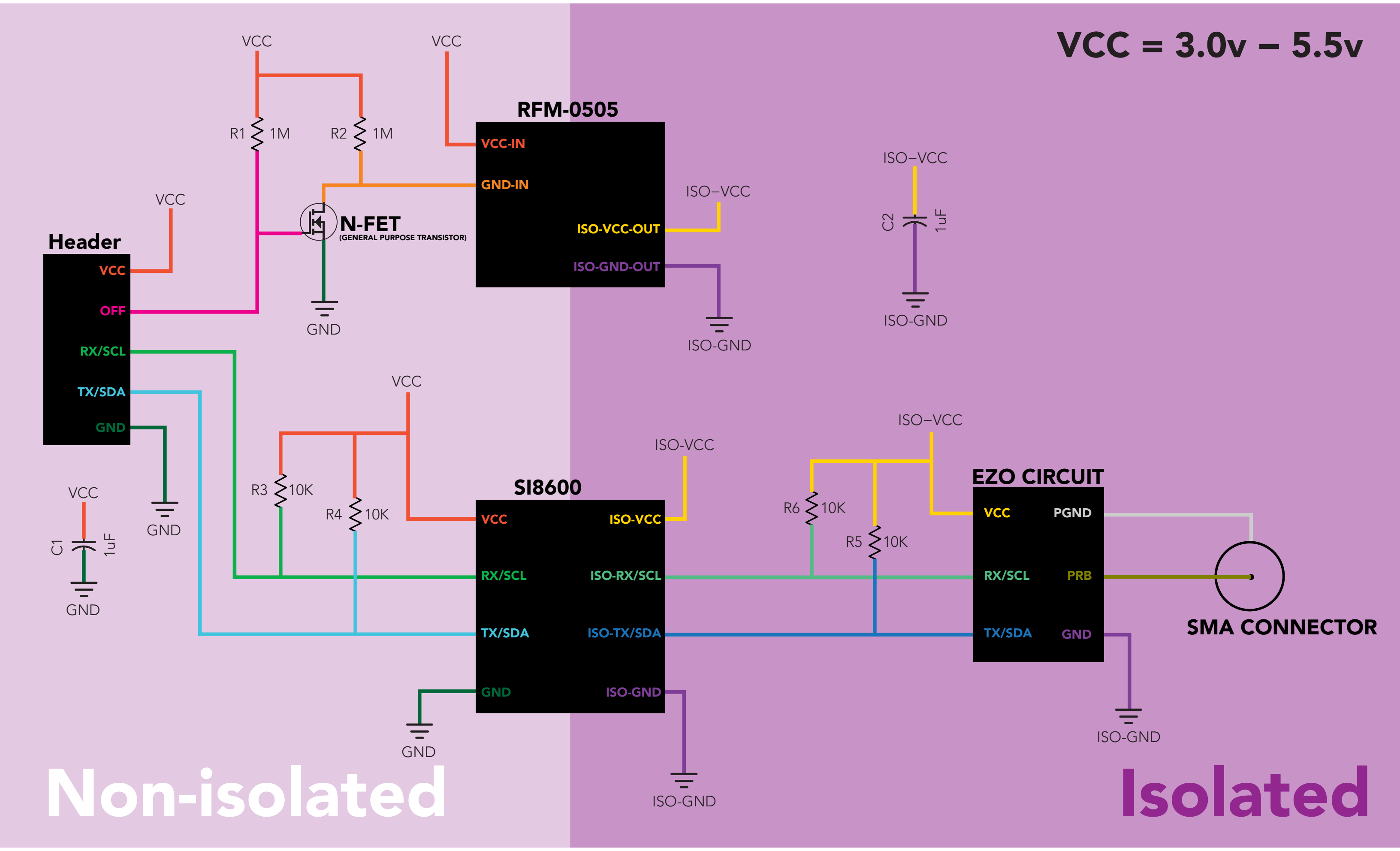
Setting the OFF pin low will shut off the Carrier Board, along with the connected EZO™ circuit. Current consumption will drop to **2.6mA**.
If the OFF pin is not used, leave it unconnected or pull to VCC.



Data isolation

This schematic shows exactly how we isolate power and data using the RFM-0505s, SI8600, and a few passive components. The RFM-0505s Isolates the power up to 200mA, 5 volts input = 5 volts output, 3 Volts input = 3 volts output, The SI8600 has two bidirectional data channels, making it ideal for UART and I2C communication. Each channel has a 10kΩ pull up resistor on both the isolated and non-isolated lines.

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



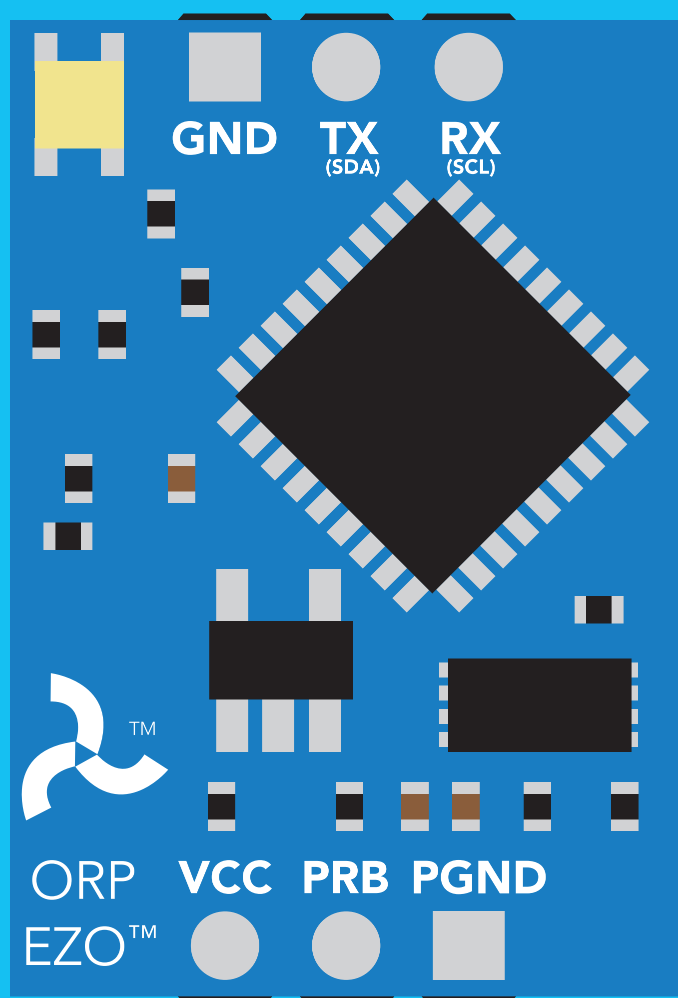
AtlasScientific Environmental Robotics

AtlasScientific™
Environmental Robotics
V 5.0
Revised 10/21

EZO-ORP™

Embedded ORP Circuit
ISO 11271 Compliant
(determination of redox potential)

Reads	**ORP**
Range	**-1019.9mV – 1019.9mV**
Accuracy	**+/– 1mV**
Response time	**1 reading per sec**
Supported probes	**Any type & brand**
Calibration	**Single point**
Temp compensation	**N/A**
Data protocol	**UART & I²C**
Default I²C address	**98 (0x62)**
Operating voltage	**3.3V – 5V**
Data format	**ASCII**



PATENT PROTECTED

Written by Jordan Press
Designed by Noah Press

This is an evolving document, check back for updates.

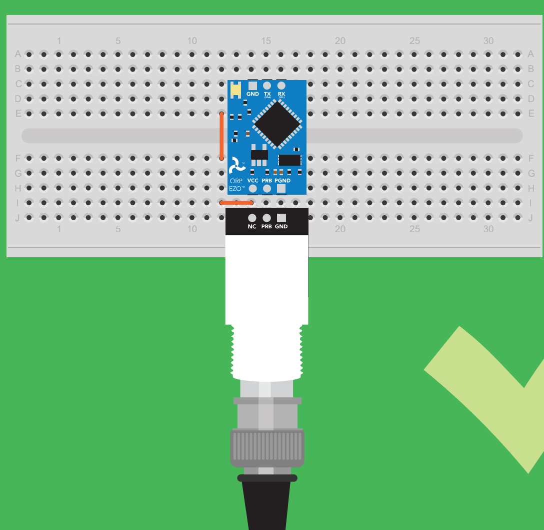
STOP

SOLDERING THIS DEVICE VOIDS YOUR WARRANTY.

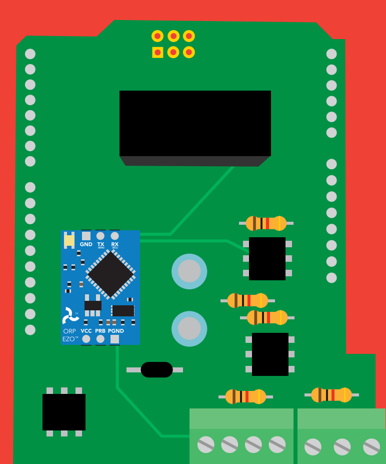
This is sensitive electronic equipment. Get this device working in a solderless breadboard first. Once this device has been soldered it is no longer covered by our warranty.

This device has been designed to be soldered and can be soldered at any time. Once that decision has been made, Atlas Scientific no longer assumes responsibility for the device's continued operation. The embedded systems engineer is now the responsible party.

Get this device working in a solderless breadboard first!



Do not embed this device without testing it in a solderless breadboard!



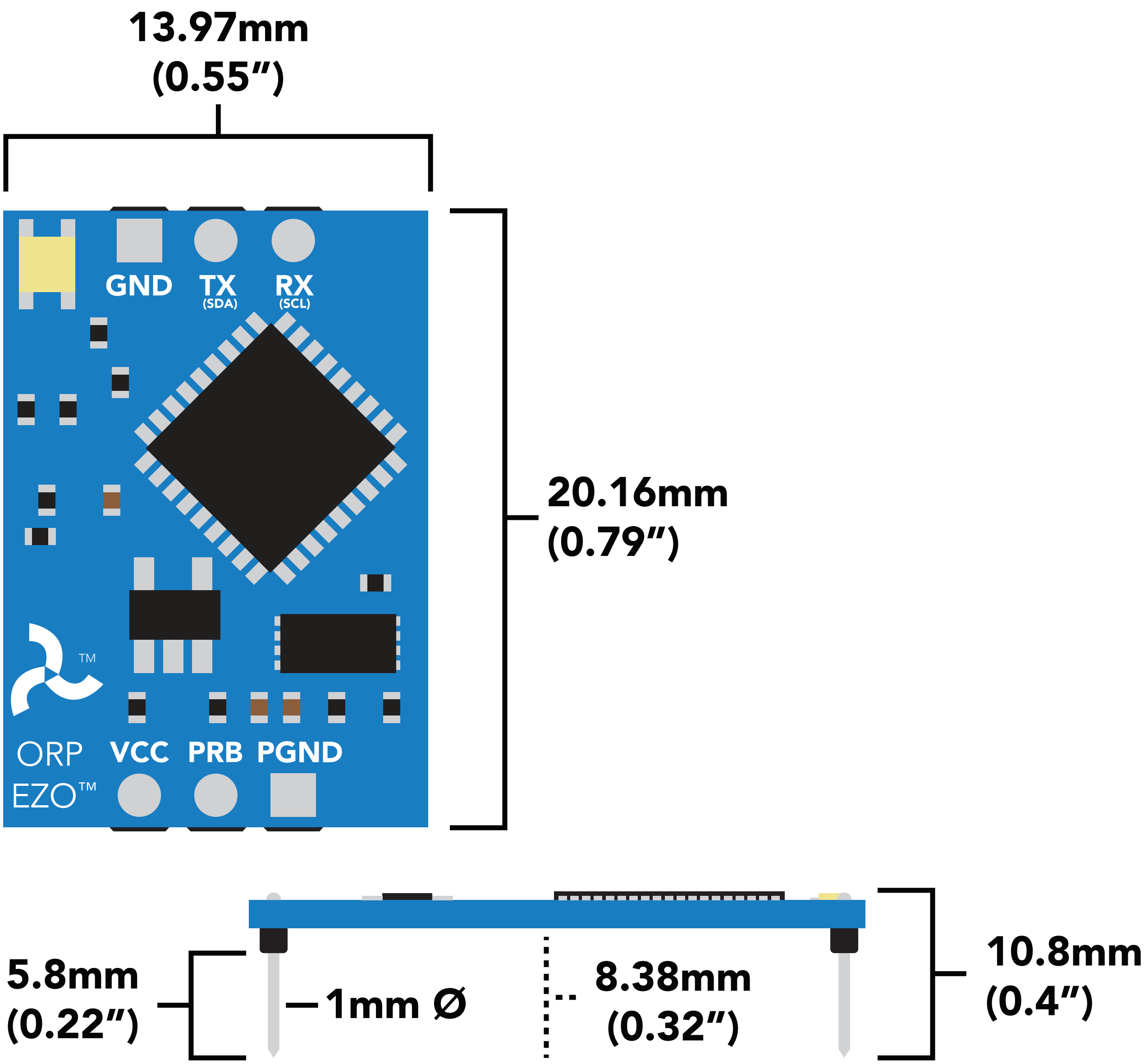
r 0.1

Table of contents

UART

I²C

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™ ORP)	-65 °C		125 °C
Operational temperature (EZO™ ORP)	-40 °C	25 °C	85 °C
VCC	3.3V	5V	5.5V

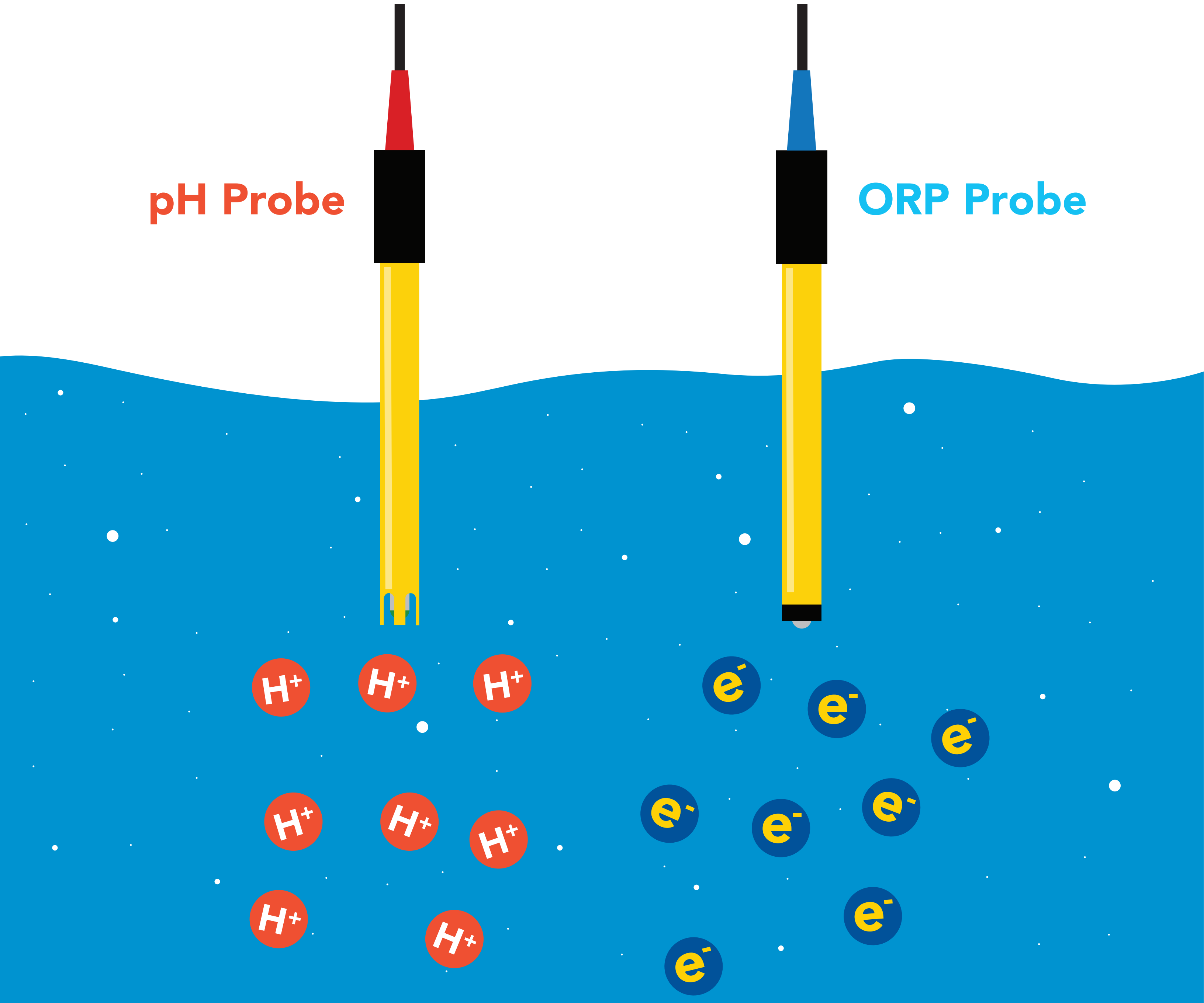
AtlasScientific™ Environmental Robotics

r 0.1

Operating principle

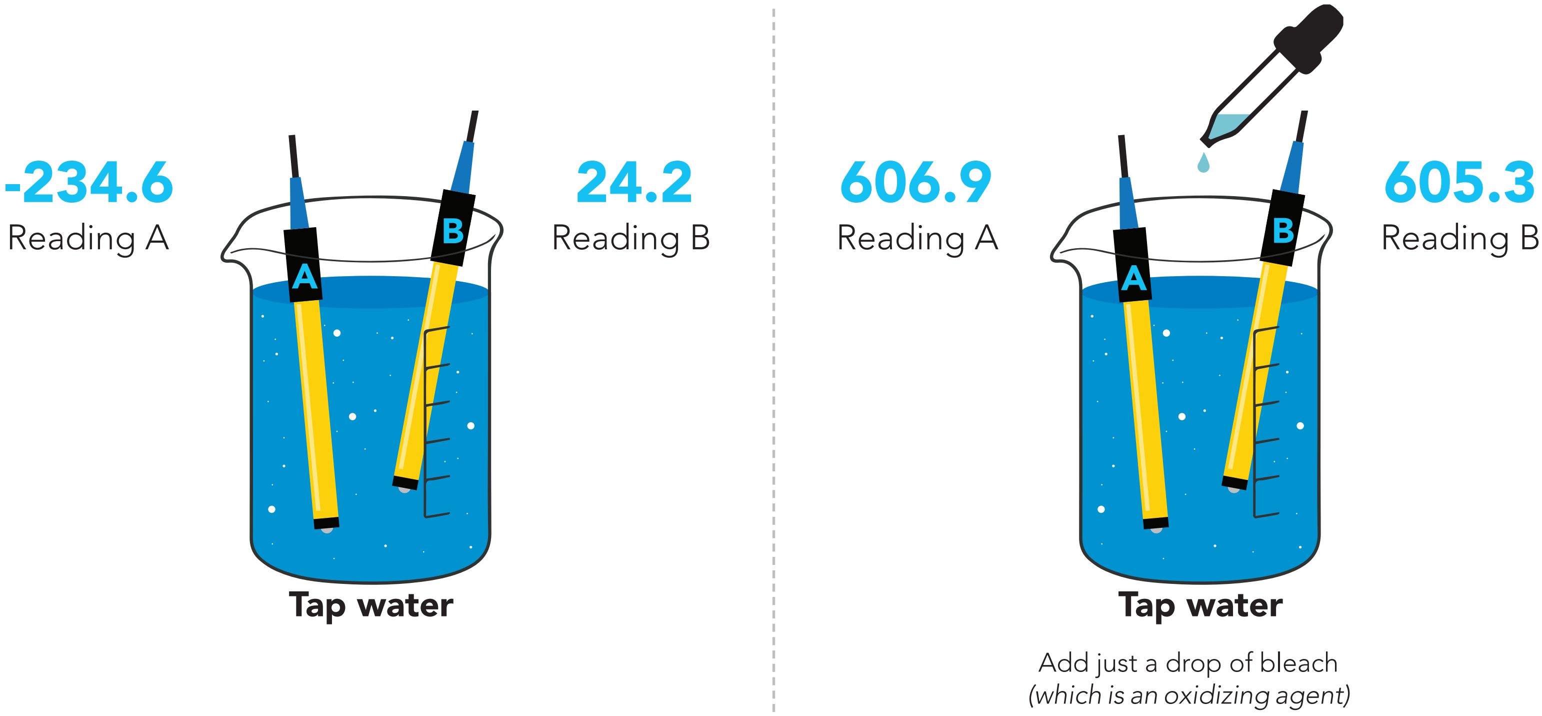
ORP stands for ***oxidation/reduction potential***. Oxidation is the loss of electrons and reduction is the gain of electrons. The output of the probe is represented in millivolts and can be positive or negative.

Just like a pH probe measures hydrogen ion activity in a liquid; an ORP probe measures electron activity in a liquid. The ORP readings represents how strongly electrons are transferred to or from substances in a liquid. Keeping in mind that the readings do not indicate the amount of electrons available for transfer.

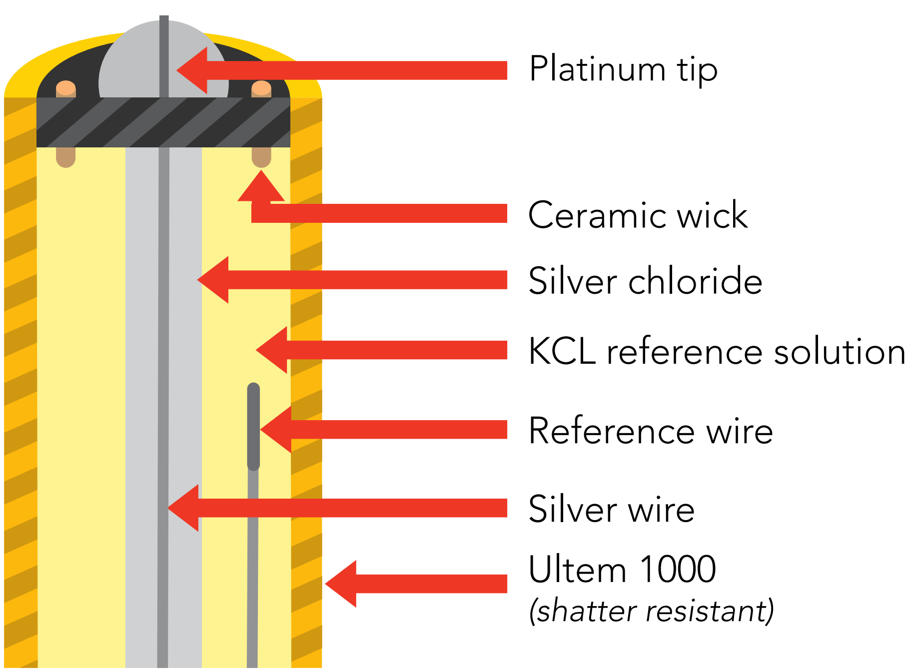


When reading the ORP of a liquid that has very few electrons available for transfer ORP readings can appear to be inconsistent.

The water is unreactive and has only trace amounts of electron movement. *These readings are equivalent to the readings you see with an unconnected multimeter.*



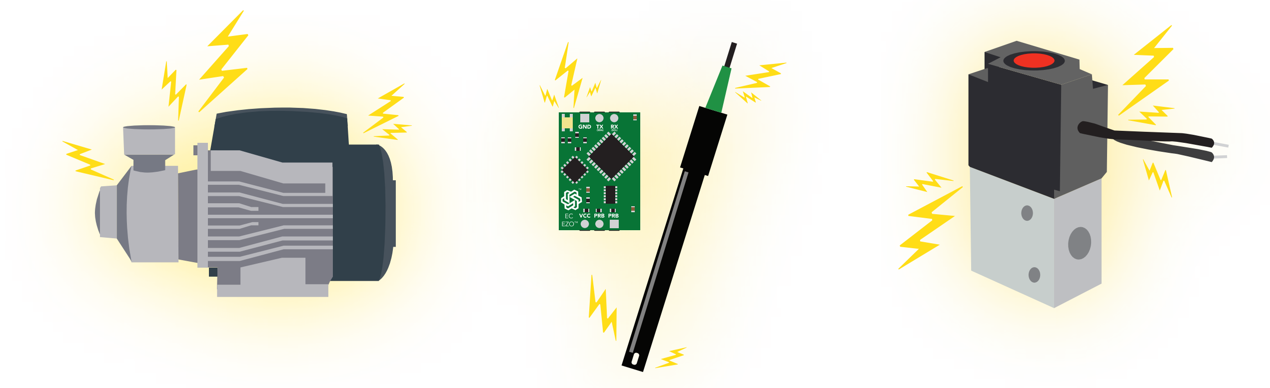
An ORP probe has a platinum tip that is connected to a silver wire, surrounded by silver chloride. That silver wire is then connected to a KCL reference solution. Because platinum is an unreactive metal it can "silently observe" the electron activity of the liquid without becoming apart of whatever reaction is occurring in the liquid.



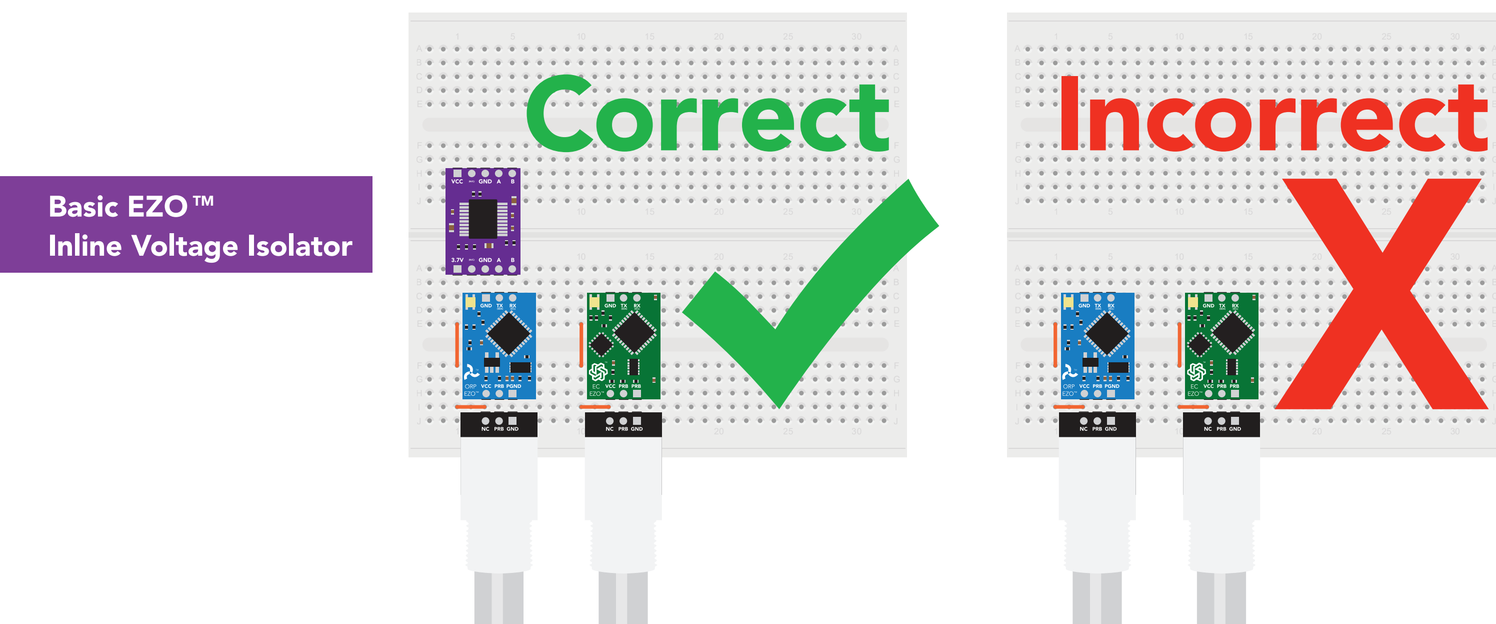
Power and data isolation

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

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



When reading ORP and Conductivity or Dissolved Oxygen together, it is ***strongly recommended*** that the EZO™ ORP circuit is electrically isolated from the EZO™ Conductivity or Dissolved Oxygen circuit.

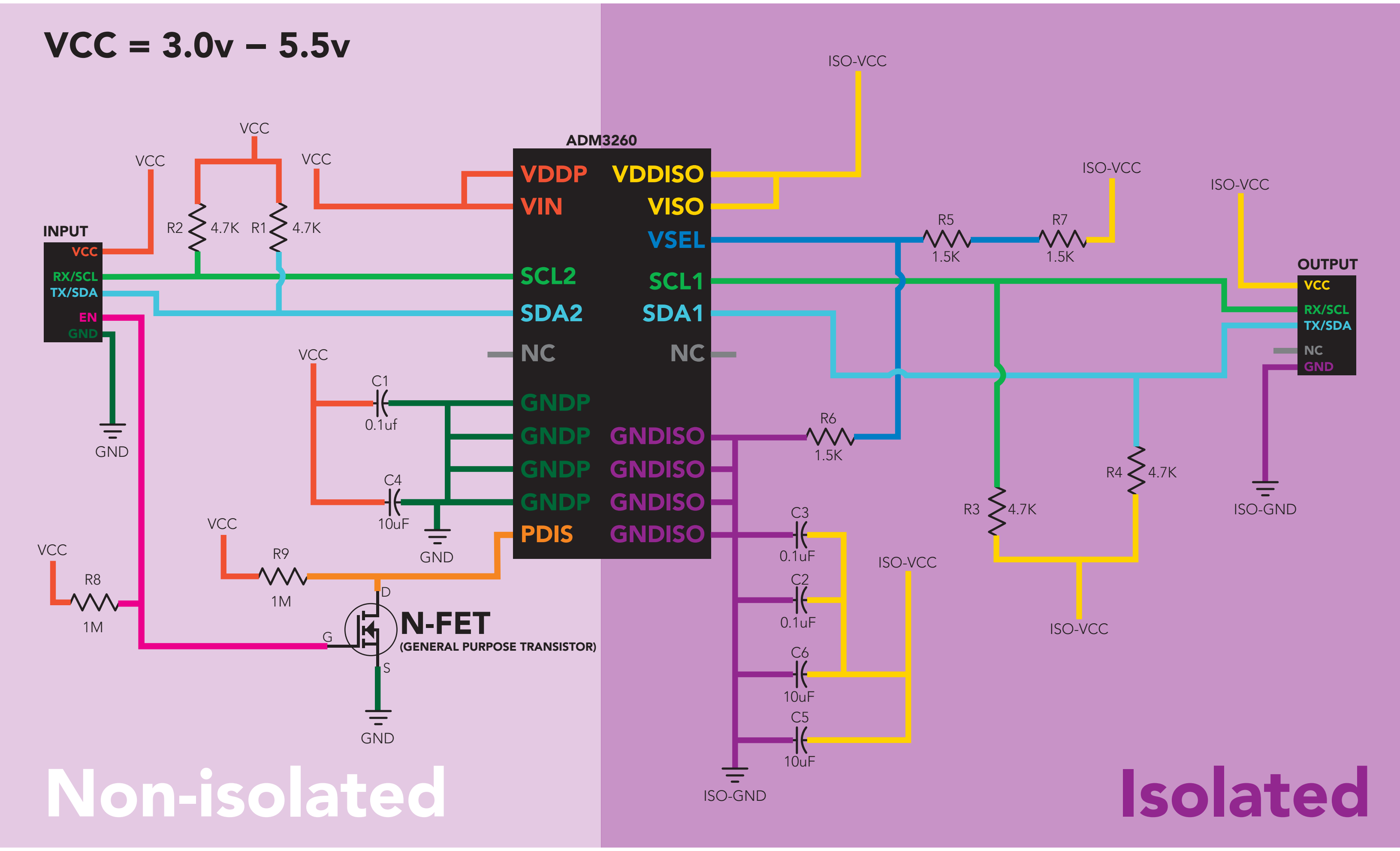


Without isolation, Conductivity and Dissolved Oxygen readings will effect ORP accuracy.

This schematic shows exactly how we isolate data and power using the and a few passive components. The ADM3260 can output isolated power up to 150 mW and incorporates two bidirectional data channels.

This technology works by using tiny transformers to induce the voltage across an air gap. PCB layout requires special attention for EMI/EMC and RF Control, having proper 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.

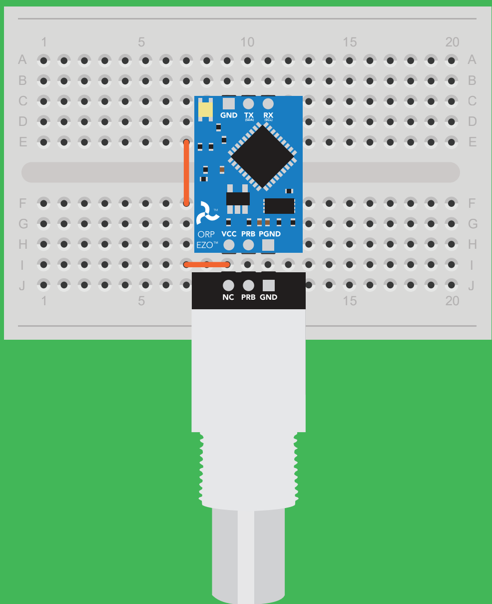
Isolated ground is different from non-isolated ground, these two lines should not be connected together.



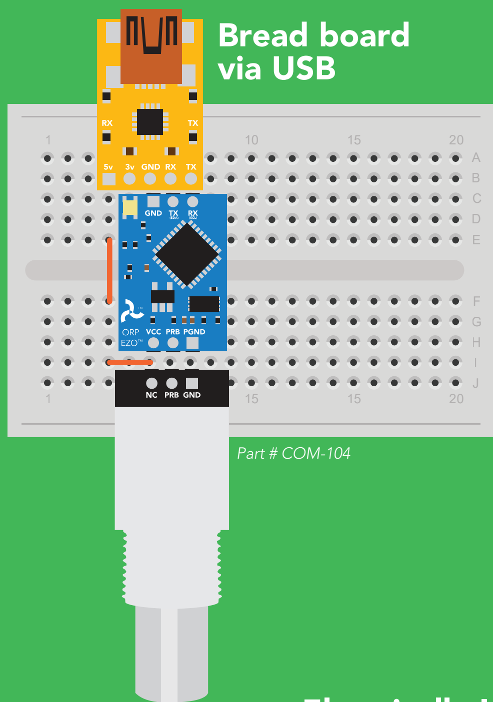
AtlasScientific™ Environmental Robotics
r 0.1

Correct wiring

Bread board

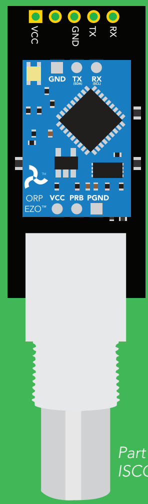


Bread board via USB



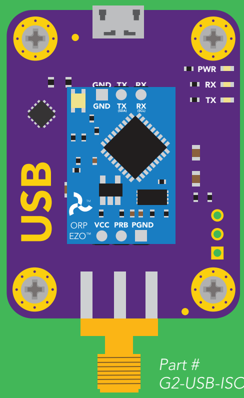
Part # COM-104

Carrier board



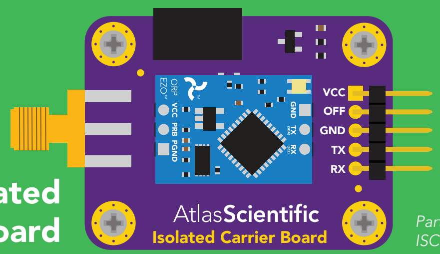
Part # ISCCB

USB carrier board



Part # G2-USB-ISO

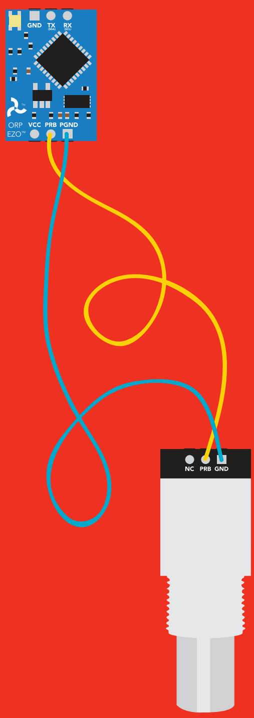
Electrically Isolated EZO™ Carrier Board



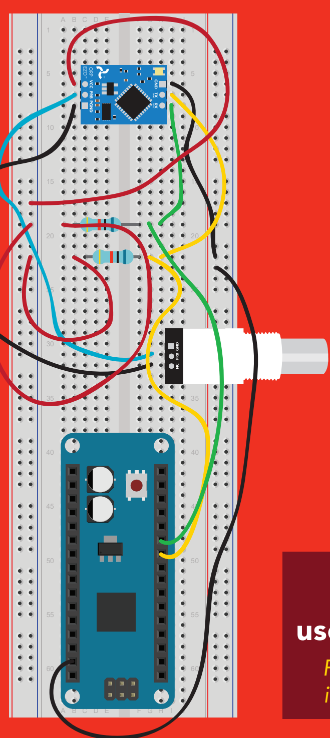
Part # ISCCB-2

Incorrect wiring

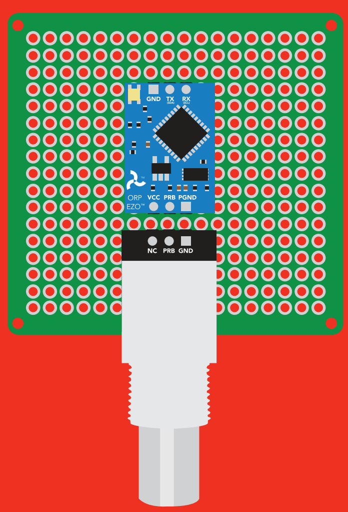
Extended leads



Sloppy setup

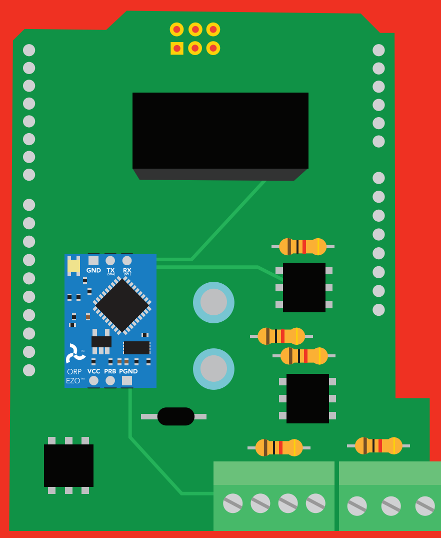


Perfboards or Protoboards



NEVER
use Perfboards or Protoboards
Flux residue and shorting wires make it very hard to get accurate readings.

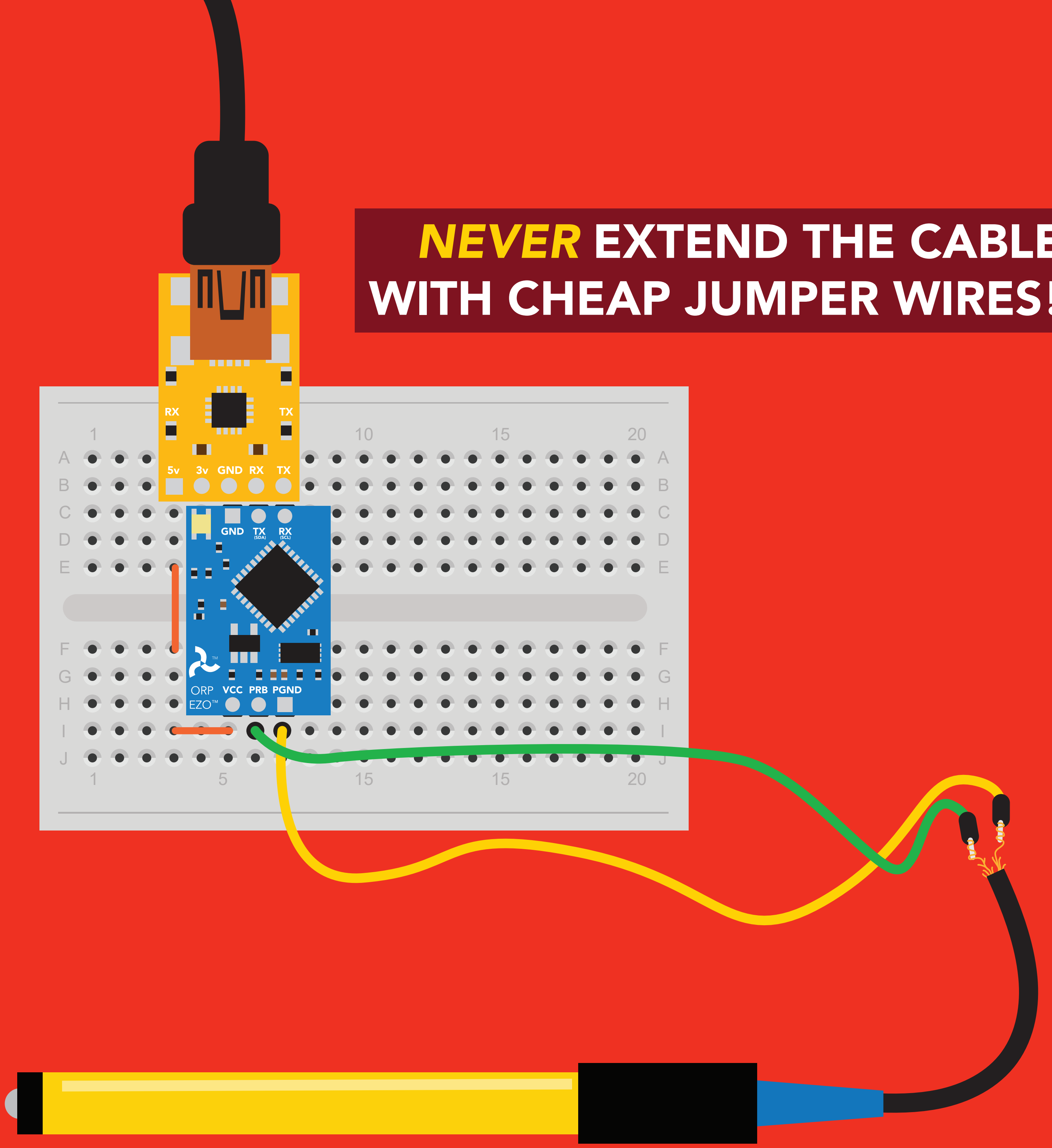
*Embedded into your device



***Only after you are familar with EZO™ circuits operation**

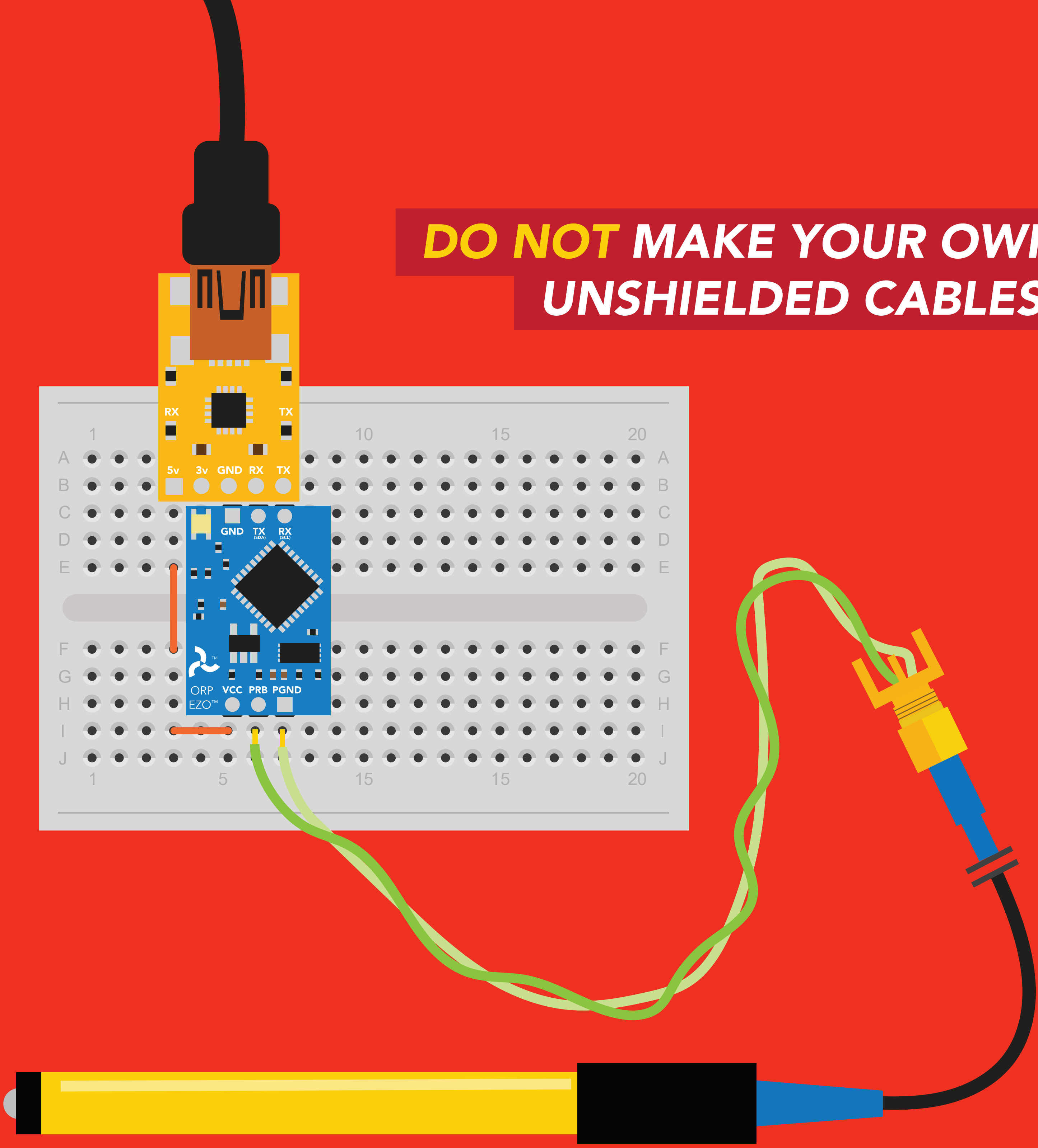
AtlasScientific
Environmental Robotics
r 0.1

NEVER EXTEND THE CABLE WITH CHEAP JUMPER WIRES!



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

DO NOT MAKE YOUR OWN UNSHIELDED CABLES!

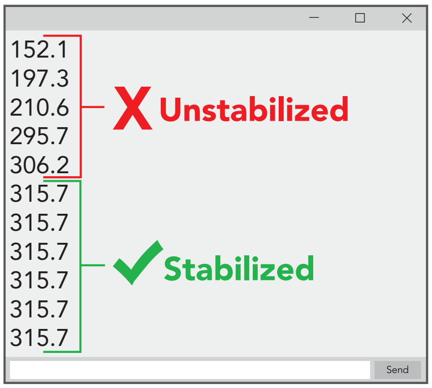


ONLY USE SHIELDED CABLES.
REFER TO *THIS DOCUMENT!*

Calibration theory

Simple calibration
UART mode
Continuous readings

Advanced calibration
I²C mode
Continuously request readings



The most important part of calibration is watching the readings during the calibration process.

It's easiest to calibrate the device in its default state (UART mode, with continuous readings enabled).

Switching the device to I²C mode after calibration ***will not*** affect the stored calibration. If the device must be calibrated in I²C mode be sure to ***continuously request readings*** so you can see the output from the probe.

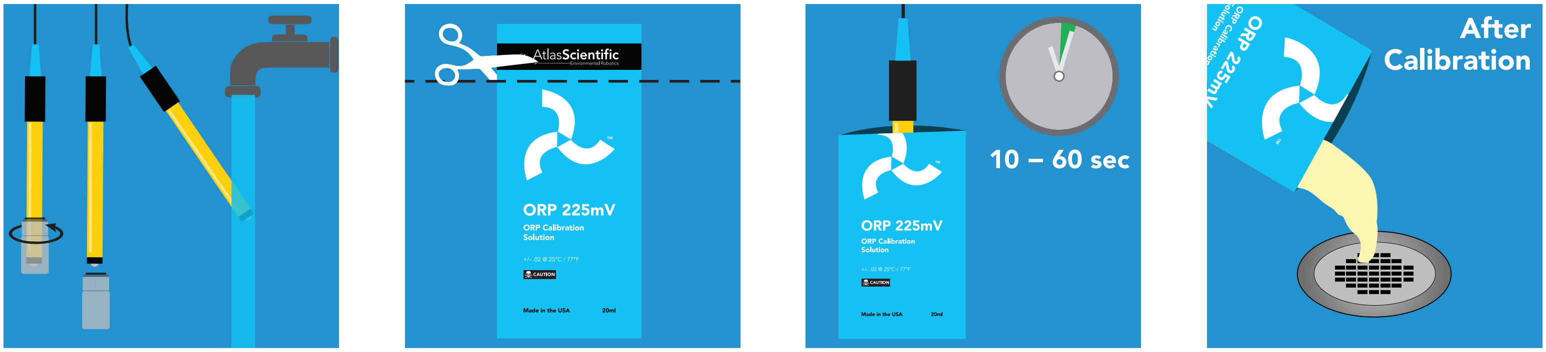
The Atlas Scientific EZO™ ORP circuit has a flexible calibration protocol, allowing single point calibration to ***any off the shelf calibration solution.***

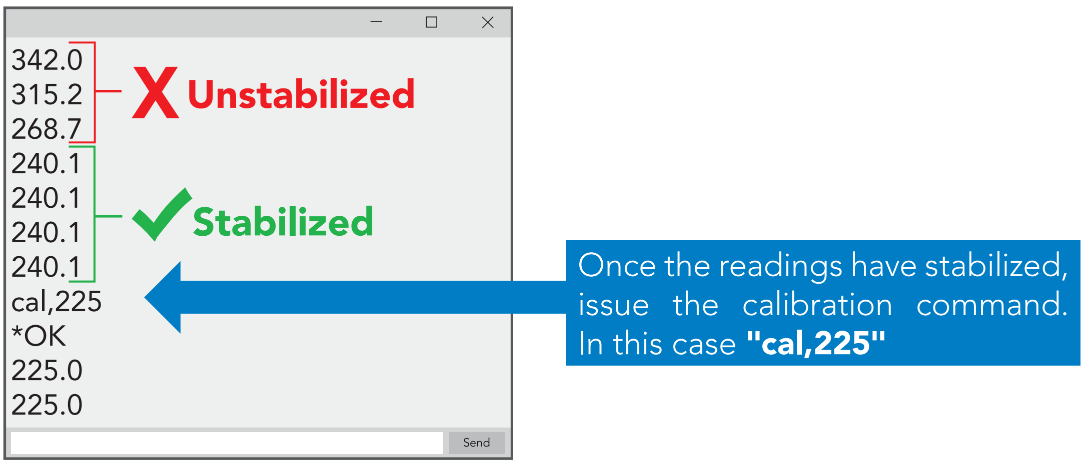
However, If this is your first time calibrating the EZO™ ORP circuit, Atlas Scientific recommends using the 225mv calibration solution.

AtlasScientific Environmental Robotics

Single point calibration

Remove the soaker bottle and rinse off the ORP probe. Remove the top of the **ORP 225mV** calibration solution pouch. Insert the ORP probe directly into the pouch, and let the probe sit in the calibration solution until the readings stabilize *(small movement from one reading to the next is normal).*





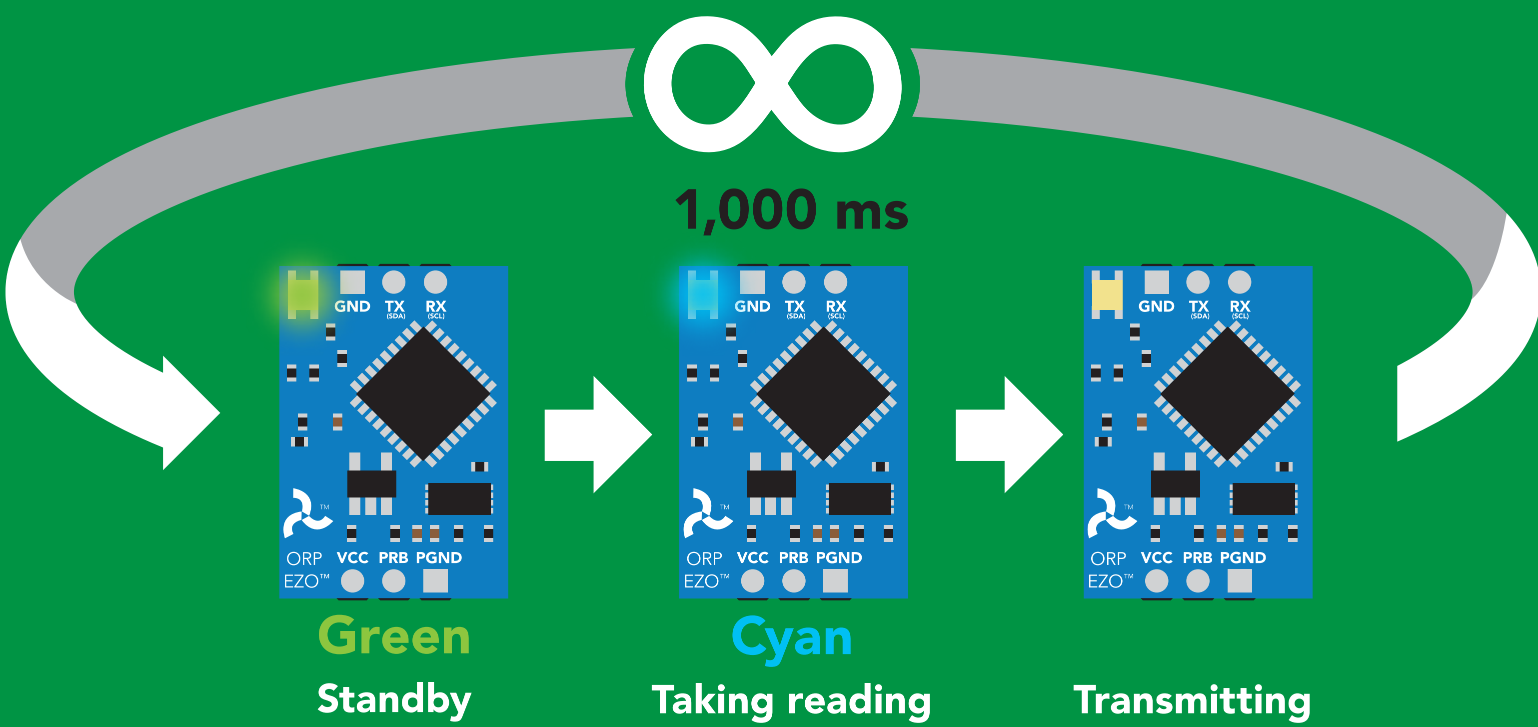
Calibration should be done at least once per year

If the ORP that's being read is continuously on the extremes of the scale *(~ -900mV or +900mV)* calibration may have to be done more often. The exact frequency of calibration will have to be determined by your engineering team.

Default state

UART mode

Baud	**9,600**
Readings	**continuous**
Speed	**1 reading per second**
LED	**on**



Available data protocols

UART

Default

I^2C

Unavailable data protocols

SPI

Analog

RS-485

Mod Bus

4–20mA

UART mode

Settings that are retained if power is cut

Baud rate
Calibration
Continuous mode
Device name
Enable/disable response codes
Hardware switch to I²C mode
LED control
Protocol lock
Software switch to I²C mode

Settings that are *NOT* retained if power is cut

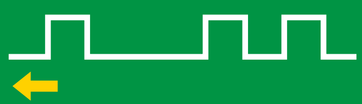
Find
Sleep mode

UART mode

8 data bits	no parity
1 stop bit	no flow control

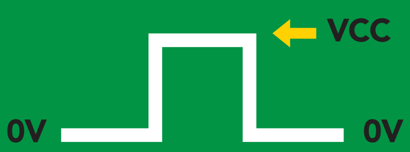
Baud
300
1,200
2,400
9,600 default
19,200
38,400
57,600
115,200

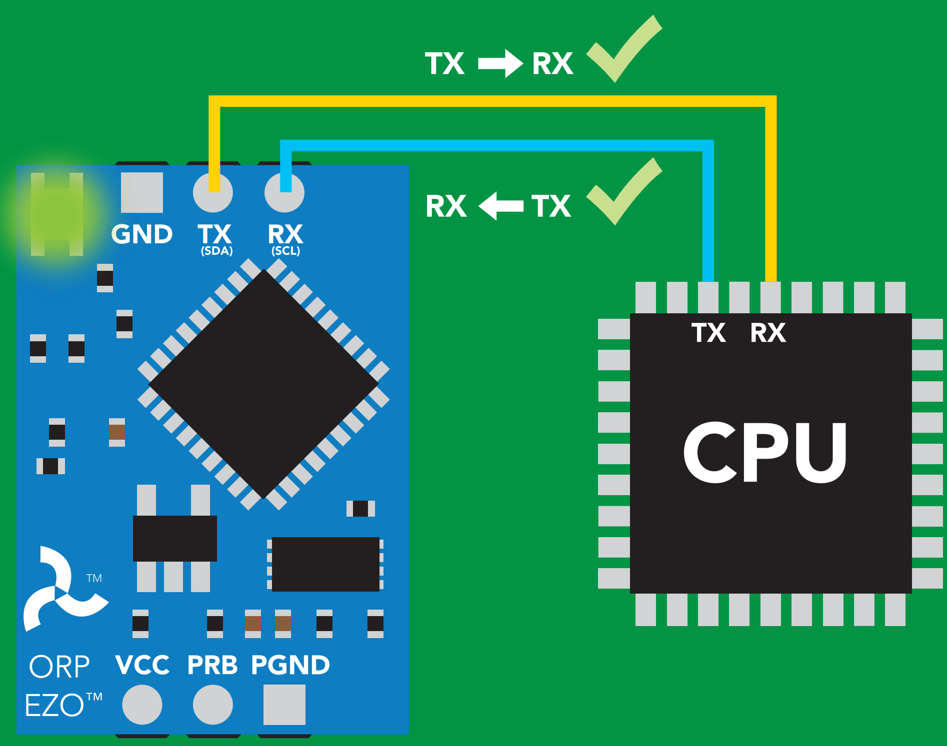
RX
Data in



TX
Data out

Vcc 3.3V – 5.5V





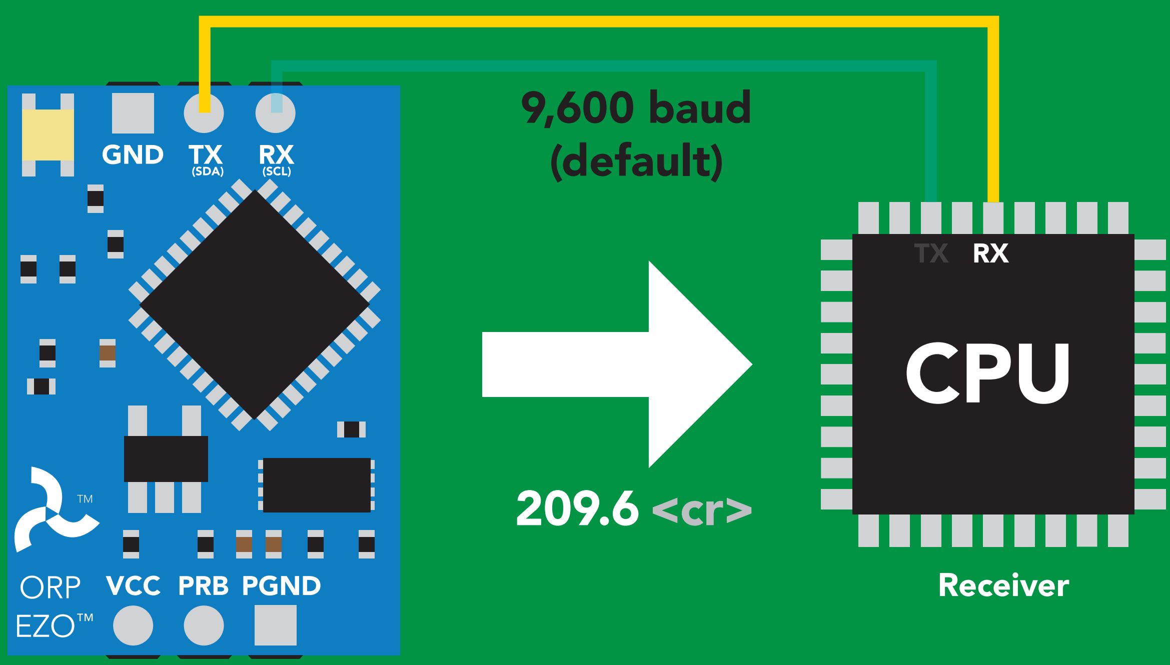
Data format

Reading	ORP	Data type	floating point
Units	mV	Decimal places	1
Encoding	ASCII	Smallest string	2 characters
Format	string	Largest string	40 characters
Terminator	carriage return		

Receiving data from device

2 parts

ASCII data string	Carriage return <cr>
Command	Terminator



Advanced

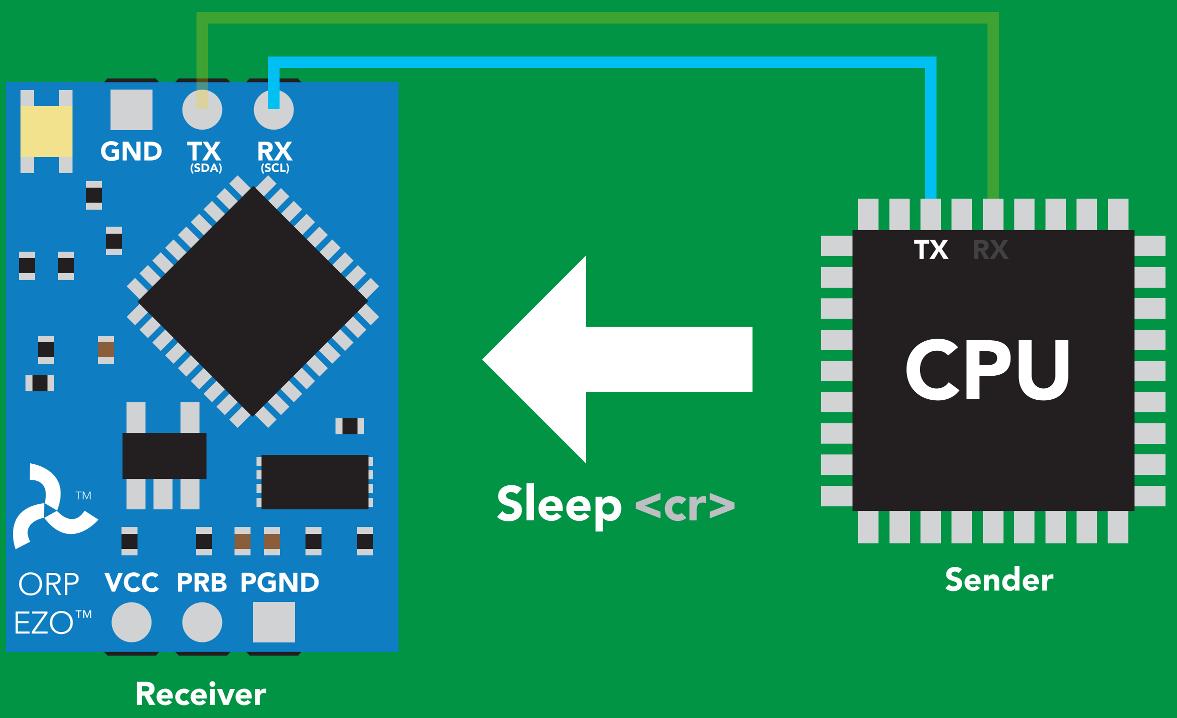
ASCII:	2	0	9	.	6	<cr>
Hex:	32	30	39	2E	36	0D
Dec:	50	48	57	46	54	13

Sending commands to device

2 parts

Command (not case sensitive) — ASCII data string

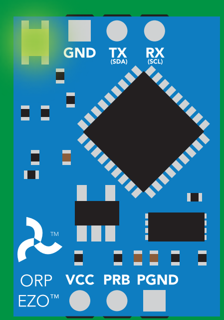
Carriage return <cr> — Terminator



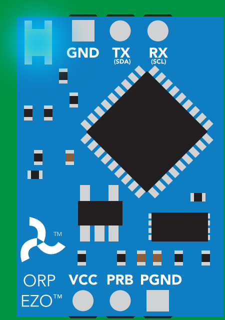
Advanced

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

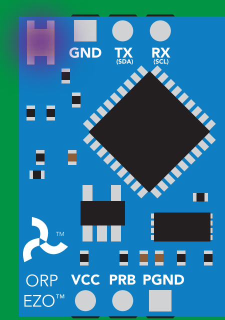
LED color definition



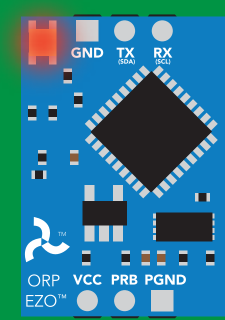
Green
UART standby



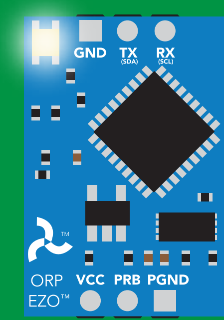
Cyan
Taking reading



Purple
Changing baud rate



Red
Command not understood



White
Find

	LED ON
5V	**+2.2 mA**
3.3V	**+0.6 mA**

UART mode

command quick reference

All commands are ASCII strings or single ASCII characters.

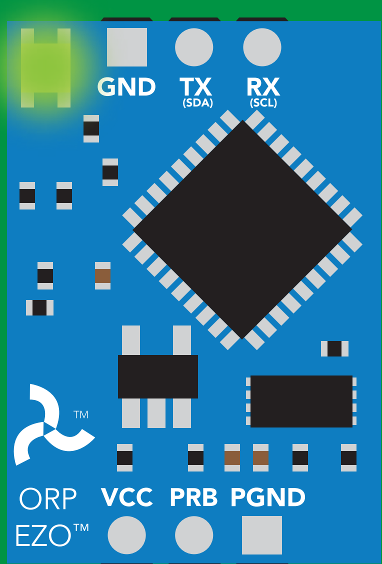
Command	Function		Default state
Baud	change baud rate	pg. 34	9,600
C	enable/disable continuous reading	pg. 24	enabled
Cal	performs calibration	pg. 26	n/a
Export	export calibration	pg. 27	n/a
Factory	enable factory reset	pg. 36	n/a
Find	finds device with blinking white LED	pg. 23	n/a
i	device information	pg. 30	n/a
I2C	change to I²C mode	pg. 37	not set
Import	import calibration	pg. 28	n/a
L	enable/disable LED	pg. 22	enabled
Name	set/show name of device	pg. 29	not set
Plock	enable/disable protocol lock	pg. 35	disabled
R	returns a single reading	pg. 25	n/a
Sleep	enter sleep mode/low power	pg. 33	n/a
Status	retrieve status information	pg. 32	n/a
*OK	enable/disable response codes	pg. 31	enable

LED control

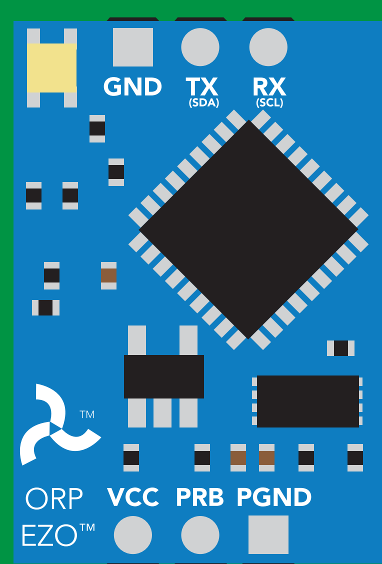
Command syntax

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> *OK <cr>



L,1



L,0

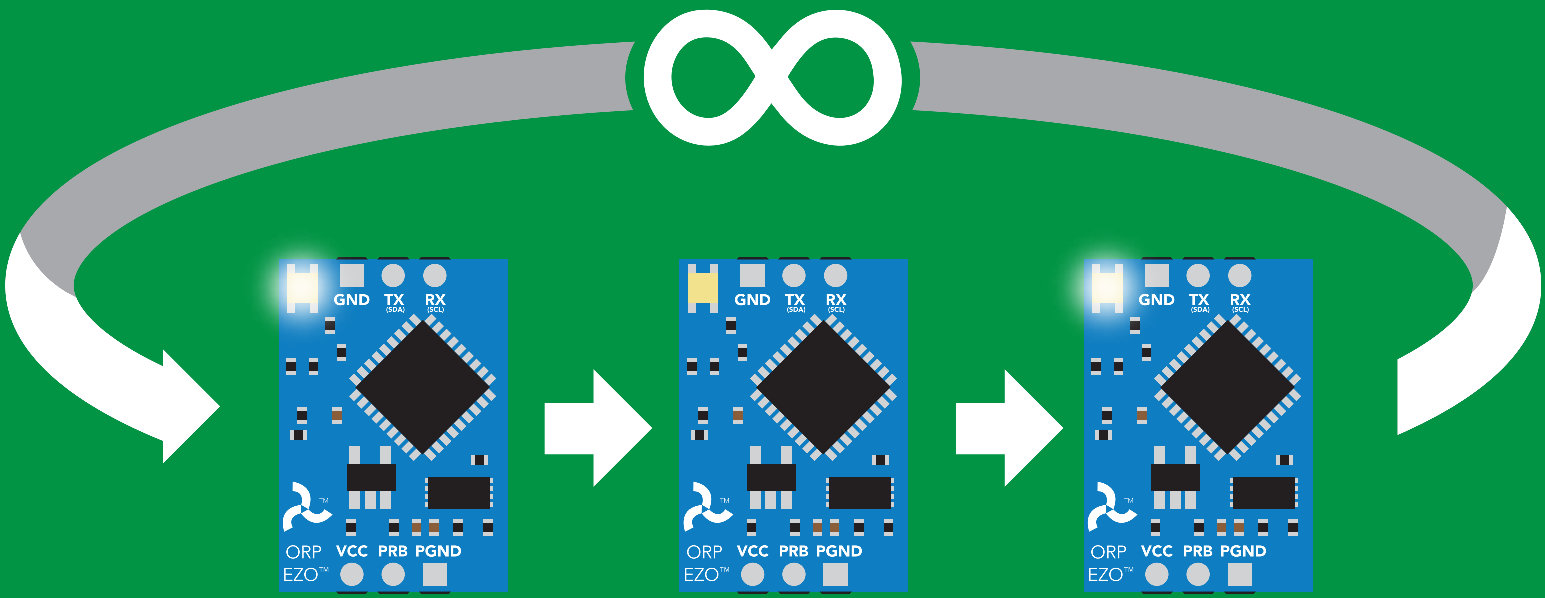
Find

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 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?	

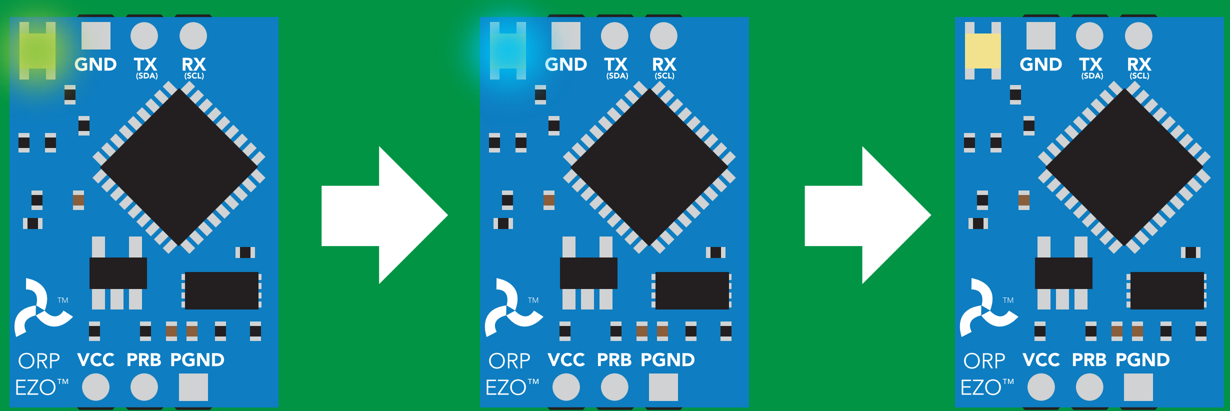
Example	Response
C,1 <cr>	*OK <cr> ORP (1 sec) <cr> ORP (2 sec) <cr> ORP (n sec) <cr>
C,30 <cr>	*OK <cr> ORP (30 sec) <cr> ORP (60 sec) <cr> ORP (90 sec) <cr>
C,0 <cr>	*OK <cr>
C,? <cr>	?C,1 <cr> or ?C,0 <cr> or ?C,30 <cr> *OK <cr>

Single reading mode

Command syntax

R <cr> **takes single reading**

Example	Response
R <cr>	209.6 <cr> *OK <cr>



Green	Cyan	
Standby	Taking reading	Transmitting

800 ms

AtlasScientific™
Environmental Robotics
r 0.1

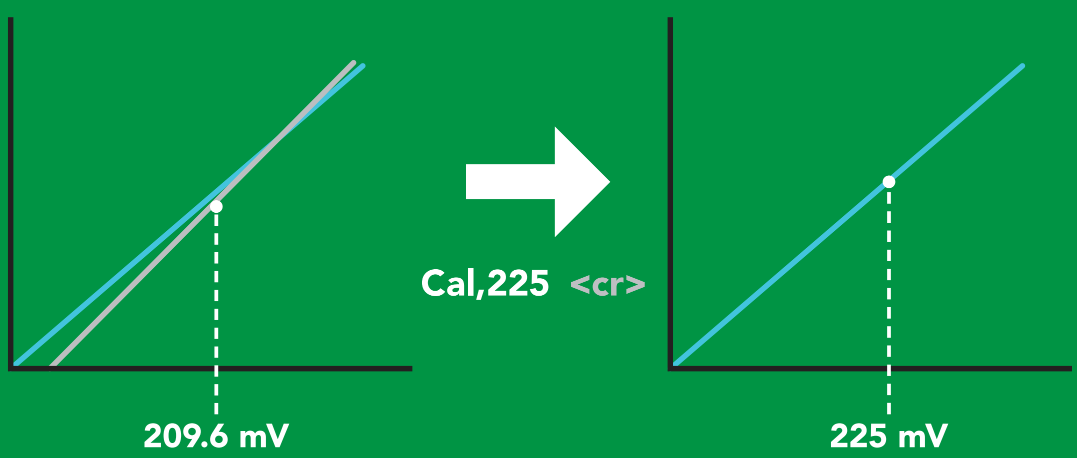
Calibration

Command syntax

The EZO™ ORP circuit can be calibrated to any known ORP value

Cal,n	<cr>	calibrates the ORP circuit to a set value
Cal,clear	<cr>	delete calibration data
Cal,?	<cr>	device calibrated?

Example	Response
Cal,225 <cr>	*OK <cr>
Cal,clear <cr>	*OK <cr>
Cal,? <cr>	?Cal,0 <cr> or ?Cal,1 <cr> *OK <cr>



Export calibration

Command syntax

Export: Use this command to download calibration settings

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

Example	Response
Export,? <cr>	10,120 <cr>

Response breakdown

10, 120

10 = # of strings to export

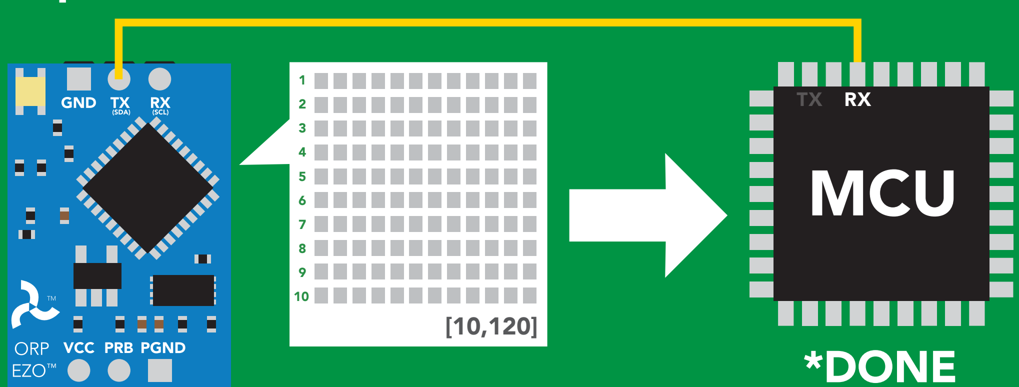
120 = # of bytes to export

Export strings can be up to 12 characters long, and is always followed by <cr>

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

Disabling *OK simplifies this process

Export <cr>



Atlas Scientific™ Environmental Robotics

Import calibration

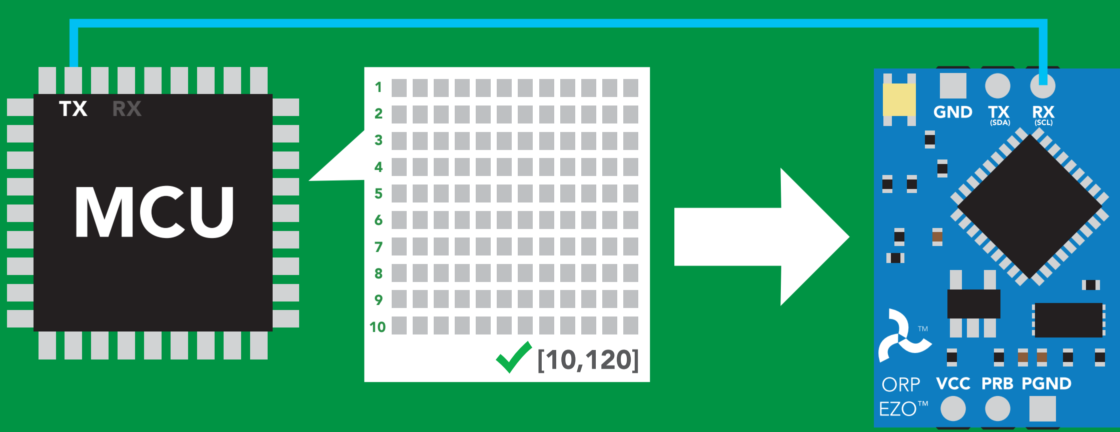
Command syntax

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

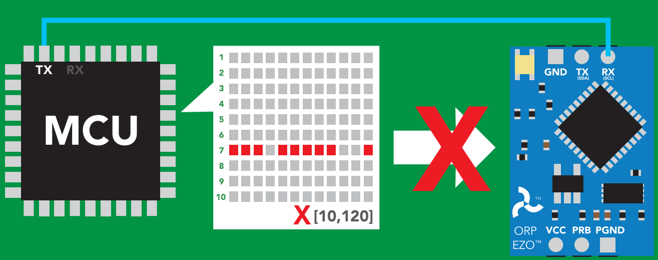
Import,n <cr> **import calibration string to new device**

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>

Import,n <cr>



*OK <cr>
system will reboot



*ER <cr>

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

Naming device

Command syntax

Do not use spaces in the name

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

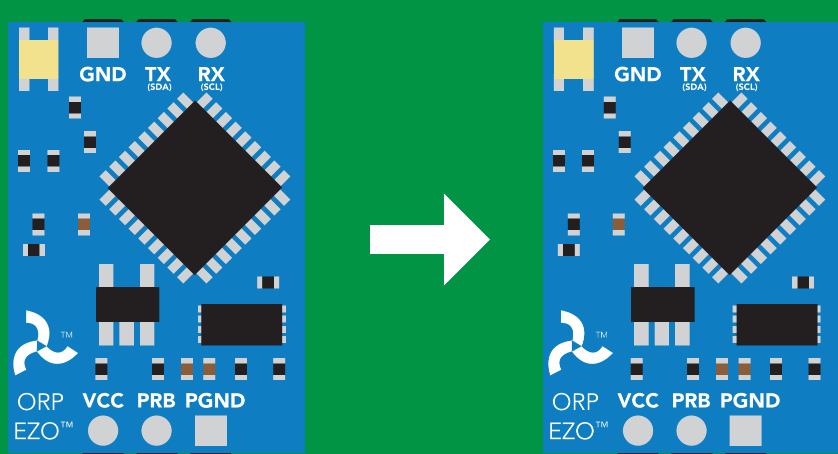
n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

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> *OK <cr>

Name,zzt

Name,?



*OK <cr>

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

AtlasScientific Environmental Robotics

r 0.2

Device information

Command syntax

i <cr> **device information**

Example	Response
i <cr>	?i,ORP,1.97 <cr> *OK <cr>

Response breakdown

?i,	ORP,	1.97
	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>	209.6 <cr> *OK <cr>
*OK,0 <cr>	no response, *OK disabled
R <cr>	209.6 <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

AtlasScientific Environmental Robotics
r 0.1

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> ***OK** <cr>

Response breakdown

?Status,	P,	5.038
	Reason for restart	Voltage at Vcc

Restart codes

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

Atlas Scientific™
Environmental Robotics
r 0.1

Sleep mode/low power

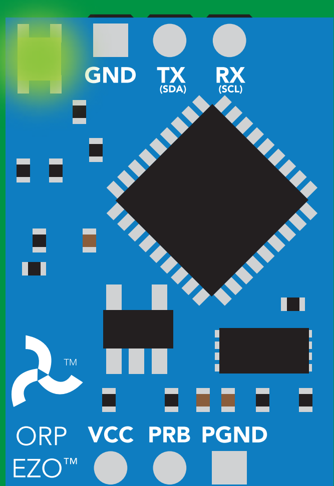
Command syntax

Send any character or command to awaken device.

Sleep <cr> enter sleep mode/low power

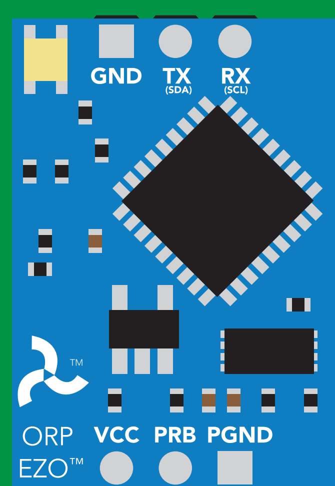
Example	Response
Sleep <cr>	*OK <cr> *SL <cr>
Any command	*WA <cr> wakes up device

	STANDBY	SLEEP
5V	16 mA	1.16 mA
3.3V	13.9 mA	0.995 mA



Standby
16 mA

Sleep <cr>



Sleep
1.16 mA

AtlasScientific Environmental Robotics

r 0.1

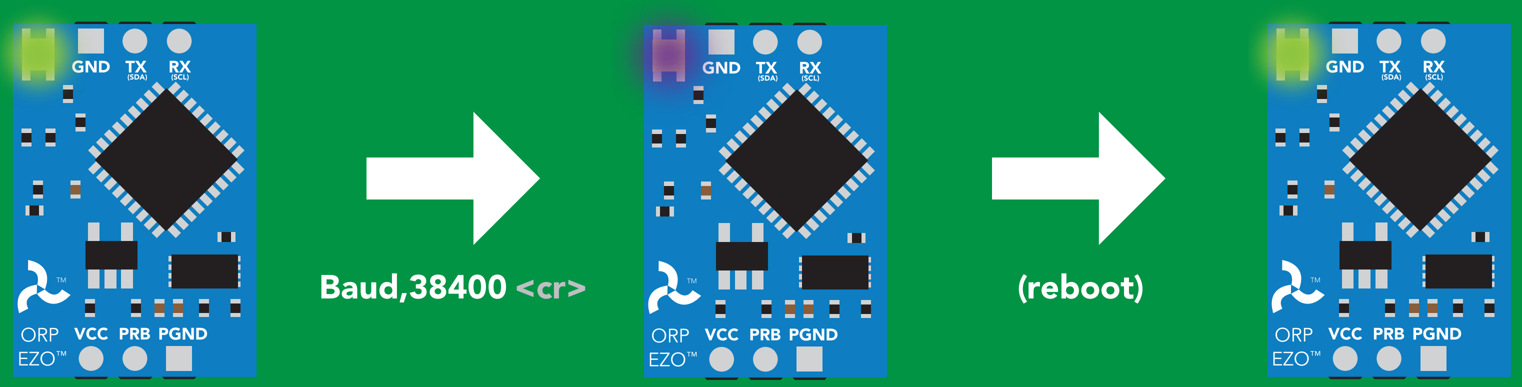
Change baud rate

Command syntax

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

Example	Response
Baud,38400 <cr>	*OK <cr>
Baud,? <cr>	?Baud,38400 <cr> *OK <cr>

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



Standby

Changing baud rate

*OK <cr>

Standby

AtlasScientific™ Environmental Robotics

r 0.1

Protocol lock

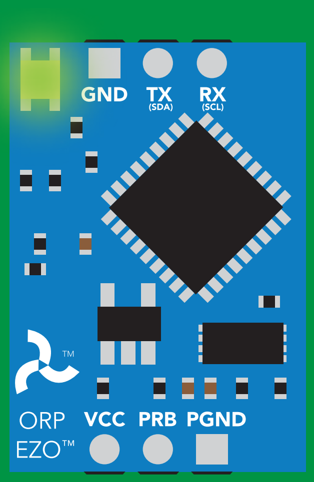
Command syntax

Locks device to UART mode.

Plock,1	<cr>	enable Plock	
Plock,0	<cr>	disable Plock	default
Plock,?	<cr>	Plock on/off?	

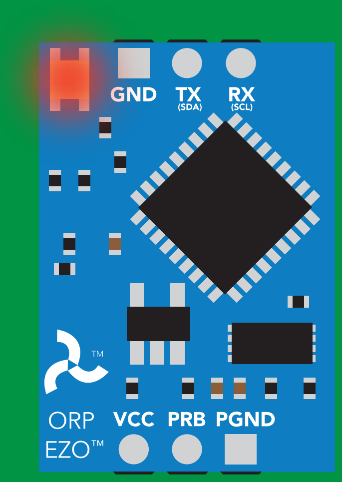
Example	Response
Plock,1 <cr>	*OK <cr>
Plock,0 <cr>	*OK <cr>
Plock,? <cr>	?Plock,1 <cr> or ?Plock,0 <cr>

Plock,1



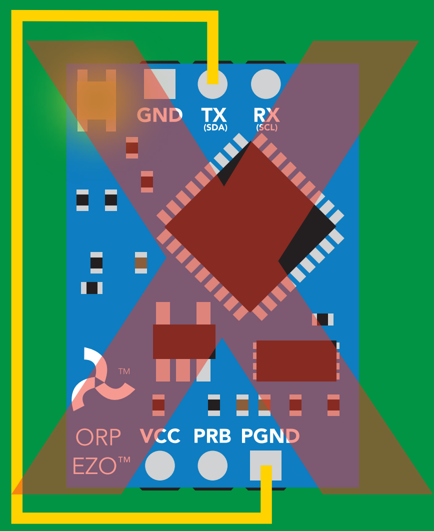
*OK <cr>

I2C,100



cannot change to I²C
*ER <cr>

Short



cannot change to I²C

Factory reset

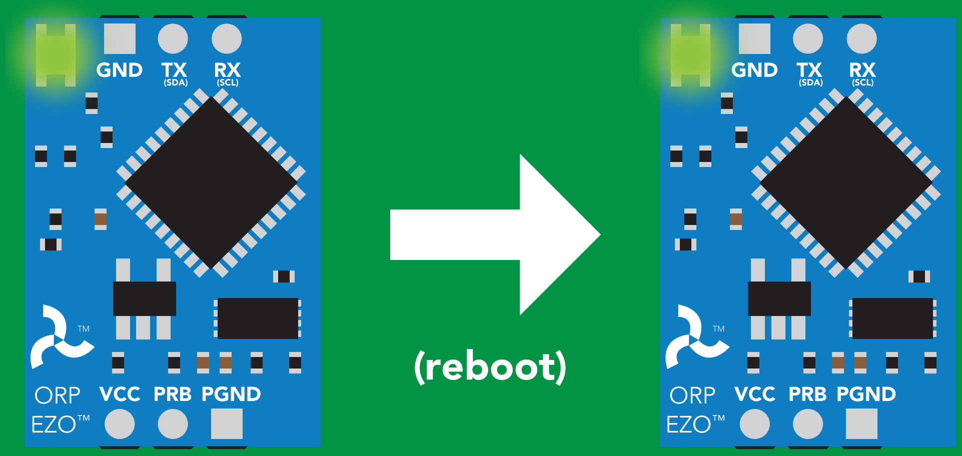
Command syntax

Clears calibration
LED on
"*OK" enabled

Factory <cr> **enable factory reset**

Example	Response
Factory <cr>	*OK <cr>

Factory <cr>



*OK <cr>

*RS <cr>
*RE <cr>

Baud rate will not change

Change to I²C mode

Command syntax

Default I²C address 98 (0x62)

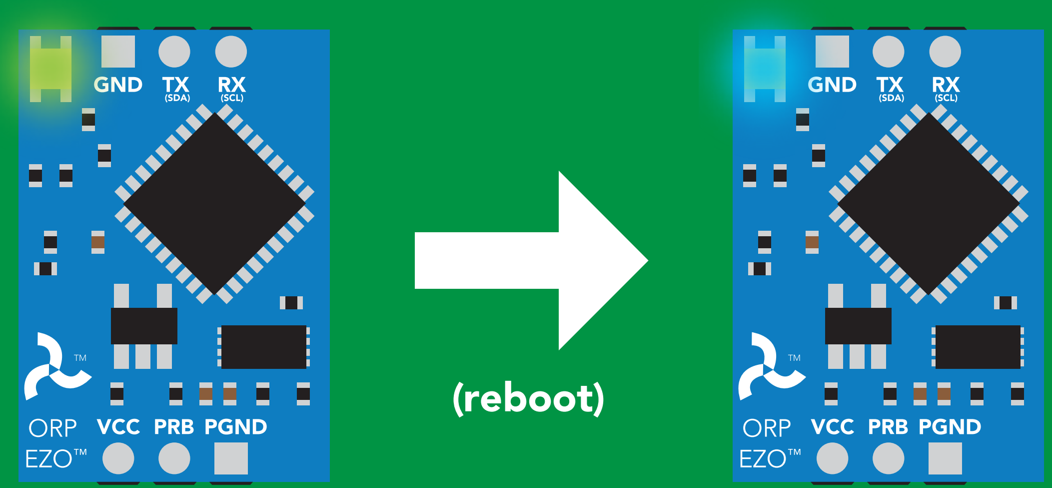
I2C,n <cr> sets I²C address and reboots into I²C mode

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



Green
*OK <cr>

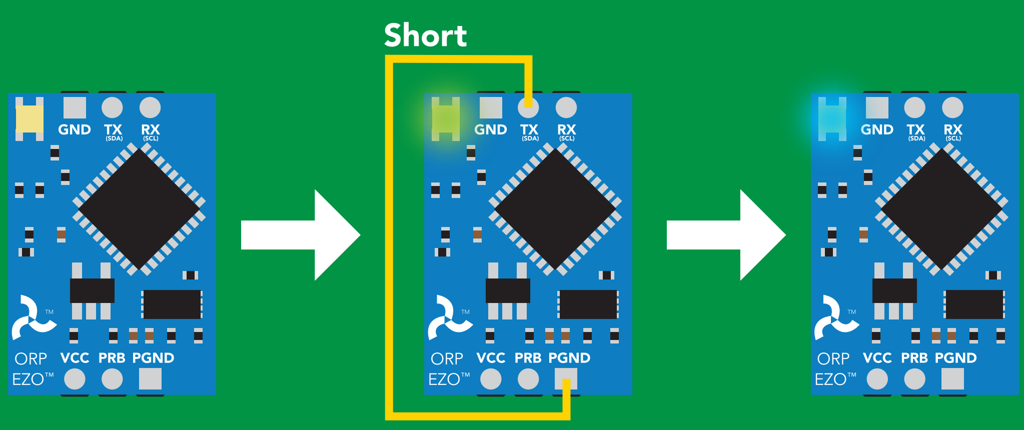
Blue
now in I²C mode

Manual switching to I²C

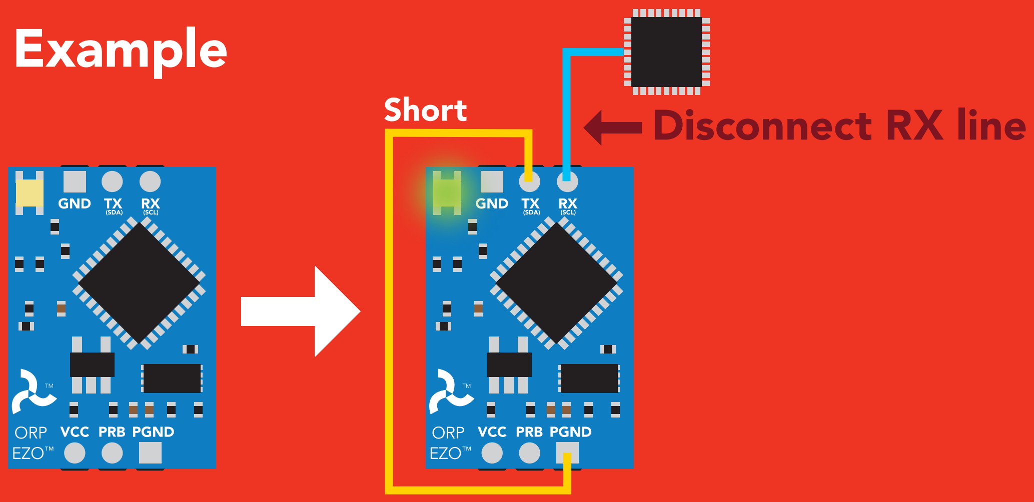
- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to PGND
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from Green to Blue
- Disconnect ground (power off)
- Reconnect all data and power

Manually switching to I²C will set the I²C address to 98 (0x62)

Example



Wrong Example



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™ device into I²C mode click here

Settings that are retained if power is cut

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

Settings that are *NOT* retained if power is cut

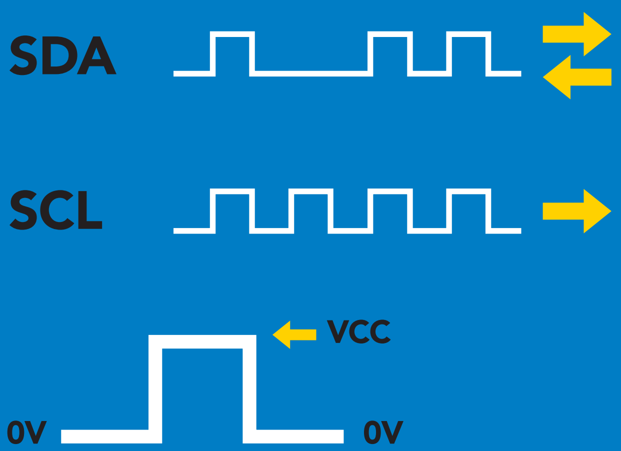
Find
Sleep mode

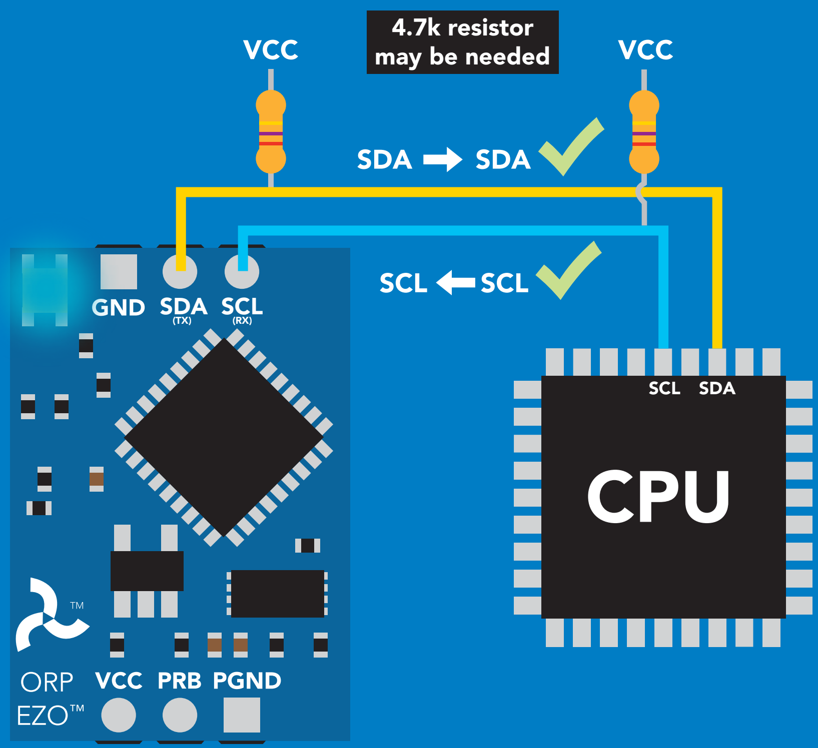
AtlasScientific™
Environmental Robotics

r 0.2

I^2C mode

I^2C address	(0x01 – 0x7F) **98 (0x62) default**
Vcc	3.3V – 5.5V
Clock speed	100 – 400 kHz





Data format

Reading	ORP	**Data type**	floating point
Units	mV	**Decimal places**	1
Encoding	ASCII	**Smallest string**	2 characters
Format	string	**Largest string**	40 characters

Sending commands to device

5 parts

Start | **I²C address** | **Write** | **Command (not case sensitive)** | **Stop**

I²C address: 98 (0x62)

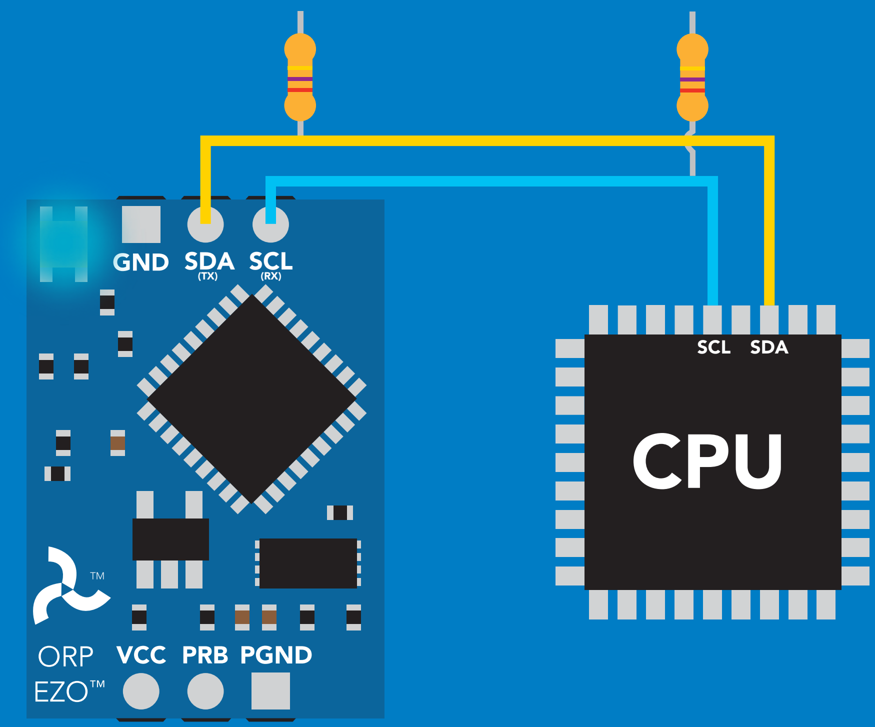
Command: ASCII command string

Example

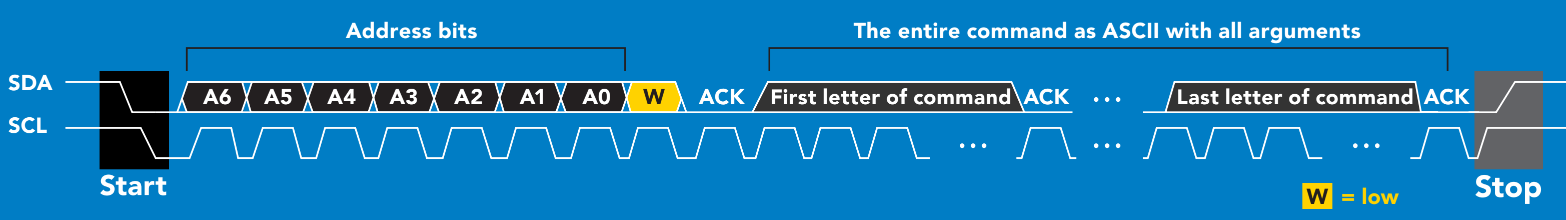
Start | **98 (0x62)** | **Write** | **Sleep** | **Stop**

98 (0x62) = I²C address

Sleep = Command

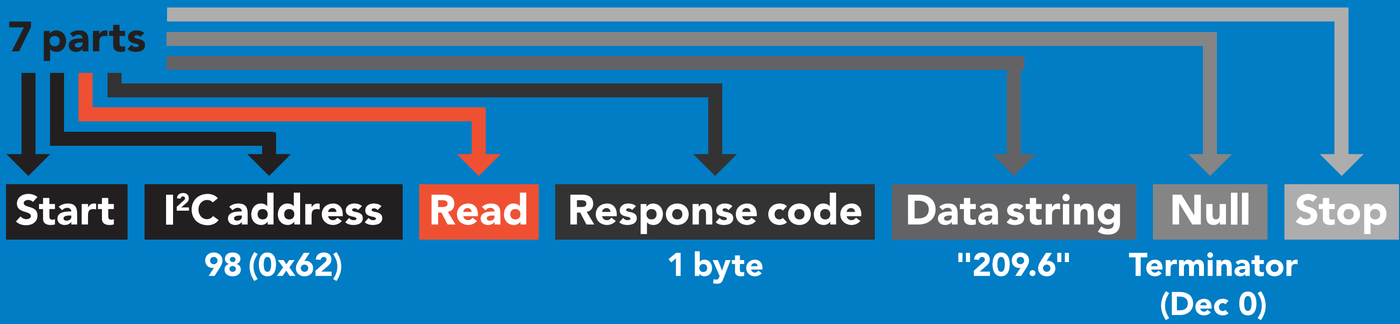


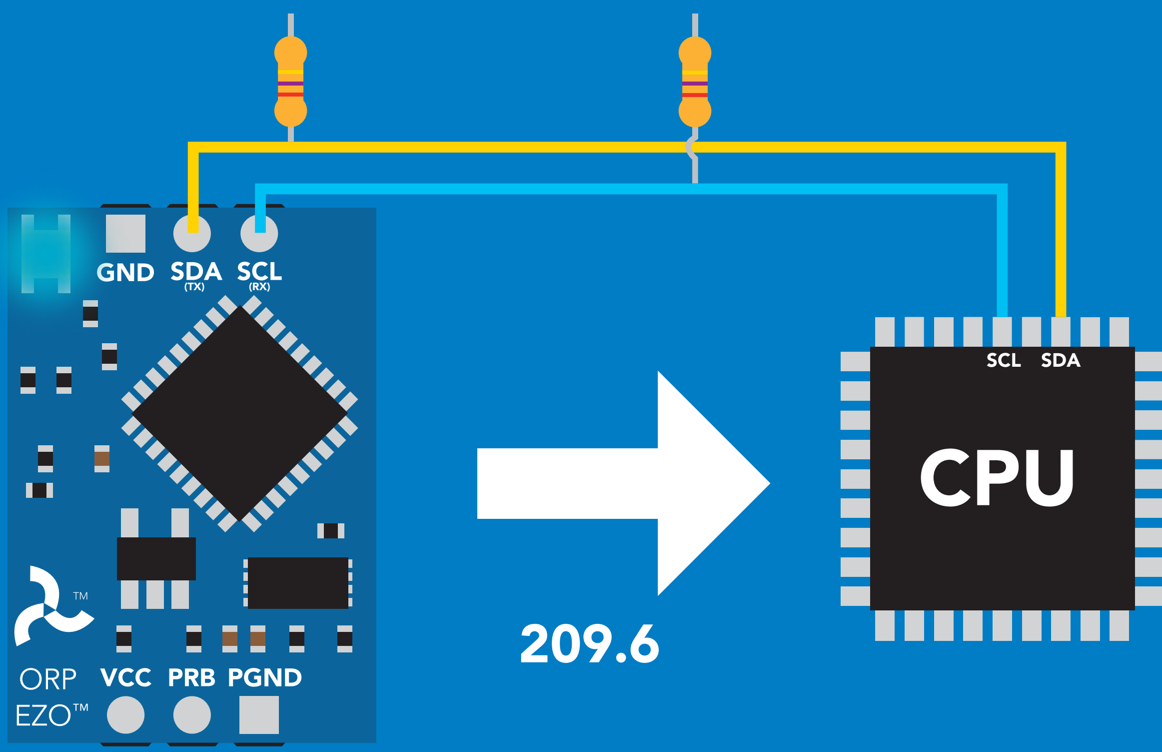
Advanced



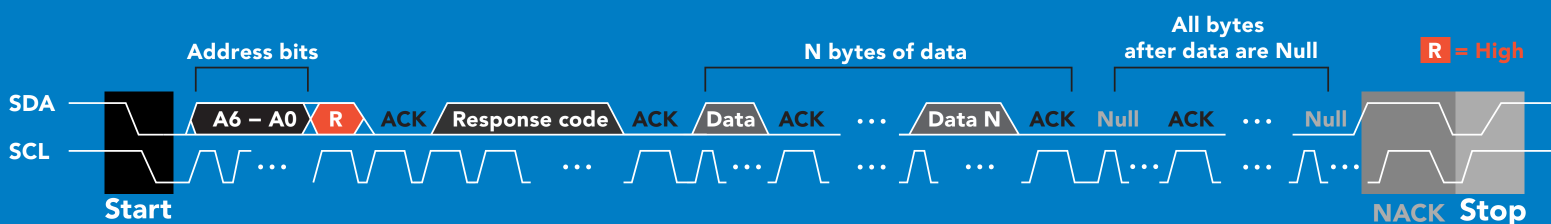
AtlasScientific
Environmental Robotics

Requesting data from device





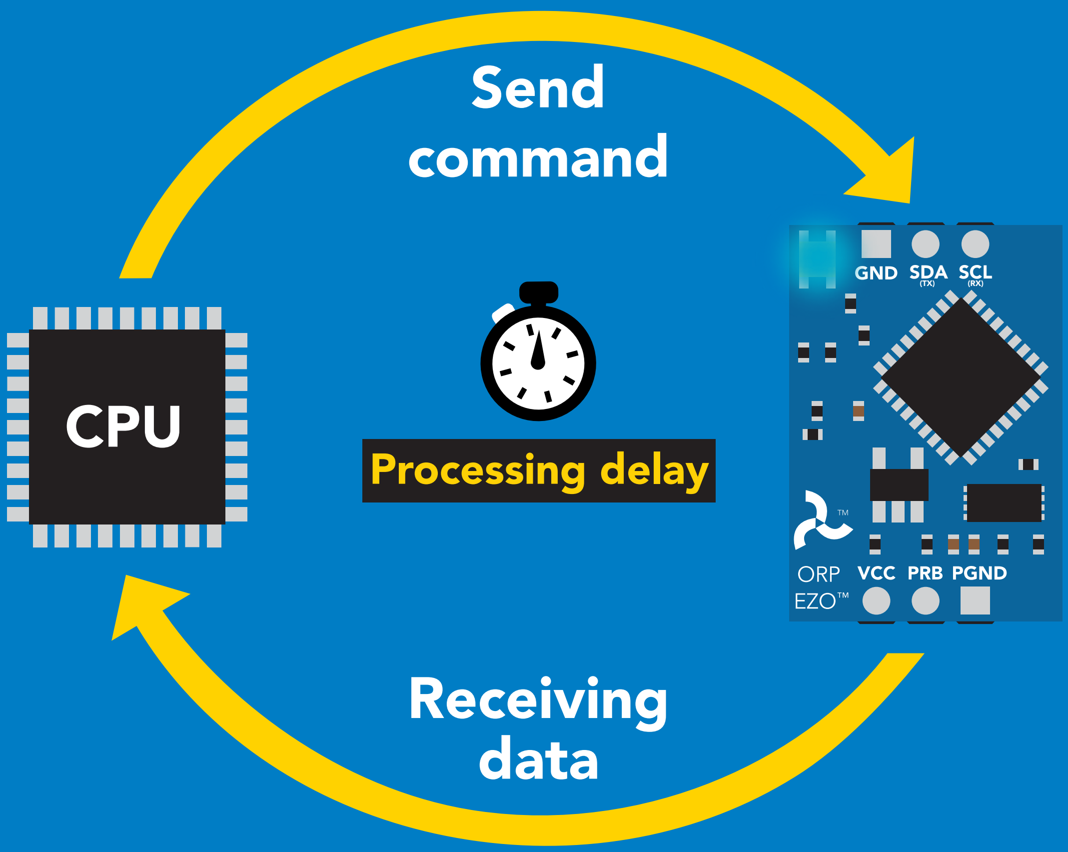
Advanced



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;
```

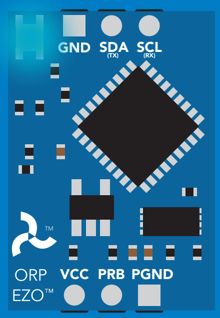
If there is no processing delay or the processing delay is too short, the response code will always be 254.

Response codes

Single byte, not string

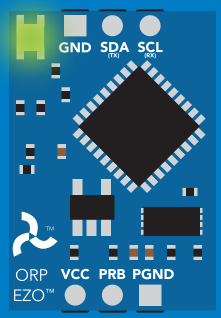
255	no data to send
254	still processing, not ready
2	syntax error
1	successful request

LED color definition



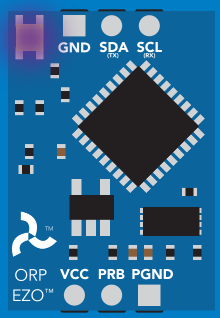
Blue

I²C standby



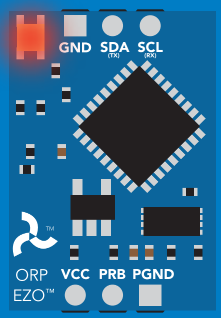
Green

Taking reading



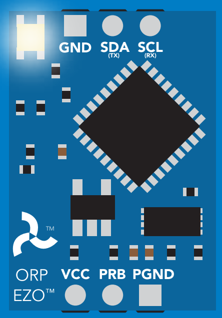
Purple

Changing I²C address



Red

Command not understood



White

Find

	LED ON
5V	**+2.2 mA**
3.3V	**+0.6 mA**

I^2C mode
command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function	
Baud	switch back to UART mode	pg. 59
Cal	performs calibration	pg. 49
Export	export calibration	pg. 50
Factory	enable factory reset	pg. 58
Find	finds device with blinking white LED	pg. 47
i	device information	pg. 53
I2C	change I^2C address	pg. 57
Import	import calibration	pg. 51
L	enable/disable LED	pg. 46
Name	set/show name of device	pg. 52
Plock	enable/disable protocol lock	pg. 56
R	returns a single reading	pg. 48
Sleep	enter sleep mode/low power	pg. 55
Status	retrieve status information	pg. 54

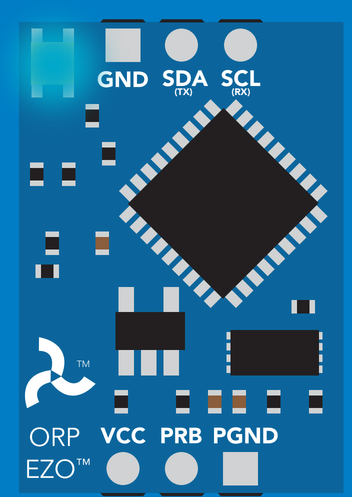
LED control

Command syntax

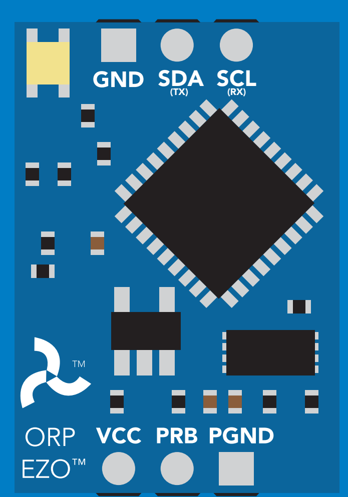
300ms processing delay

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 1 (Dec) ?L,0 (ASCII) 0 (Null)



L,1



L,0

Find

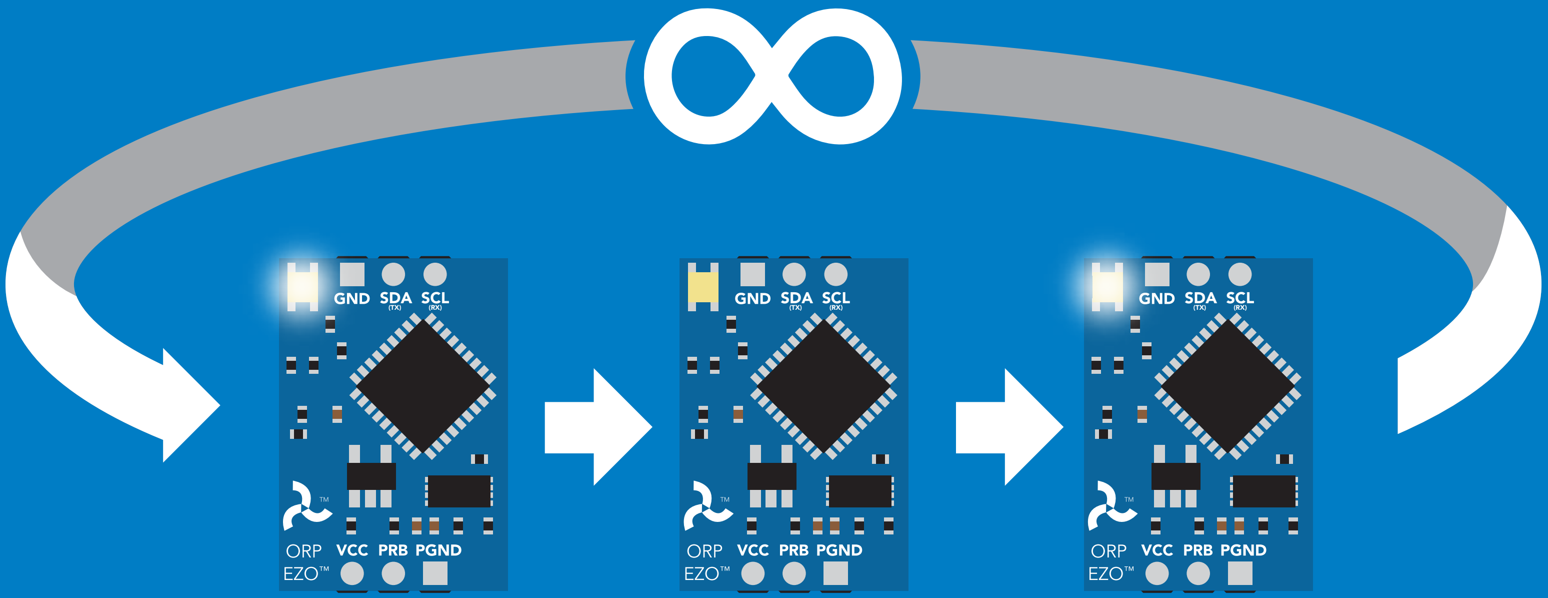
300ms processing delay

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 1 Dec 0 Null



AtlasScientific™ Environmental Robotics

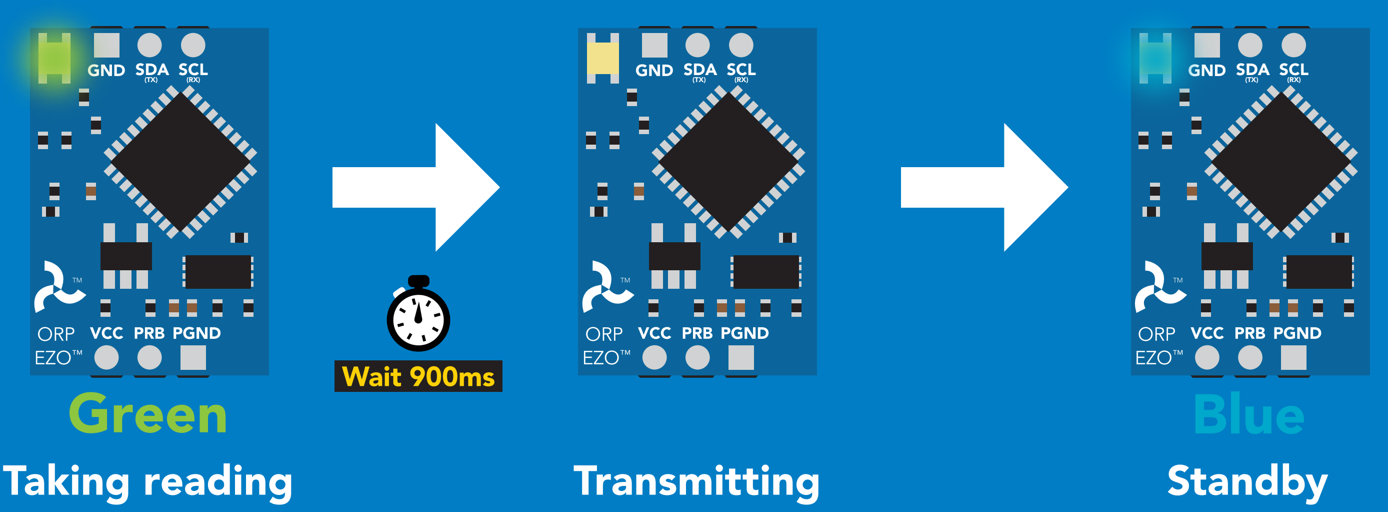
Taking reading

Command syntax

900ms processing delay

R	**return 1 reading**

Example	Response
R	Wait 900ms · 1 (Dec) · 209.6 (ASCII) · 0 (Null)



Calibration

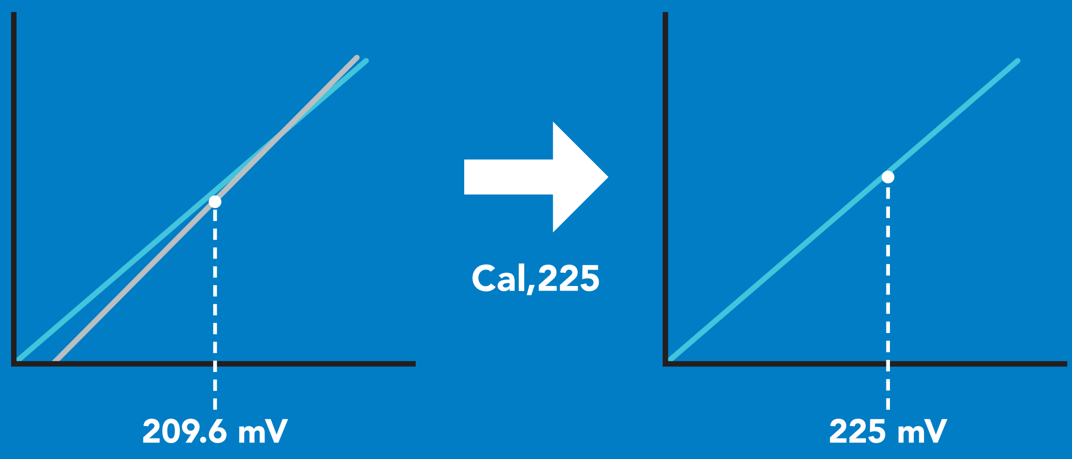
Command syntax

900ms processing delay

Cal,n	calibrates the ORP circuit to a set value
Cal,clear	delete calibration data
Cal,?	device calibrated?

The EZO™ ORP circuit can be calibrated to any known ORP value

Example	Response
Cal,225	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) 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				
(optional) Export,?	Wait 300ms	1 Dec	10,120 ASCII	0 Null	
Export	Wait 300ms	1 Dec	59 6F 75 20 61 72 ASCII	0 Null	(1 of 10)
Export	Wait 300ms	1 Dec	65 20 61 20 63 6F ASCII	0 Null	(2 of 10)
(7 more)	⋮				
Export	Wait 300ms	1 Dec	6F 6C 20 67 75 79 ASCII	0 Null	(10 of 10)
Export	Wait 300ms	1 Dec	*DONE ASCII	0 Null	

Response breakdown

10, 120

10: # of strings to export
120: # of bytes to export

Export strings can be up to 12 characters long

Import calibration

300ms processing delay

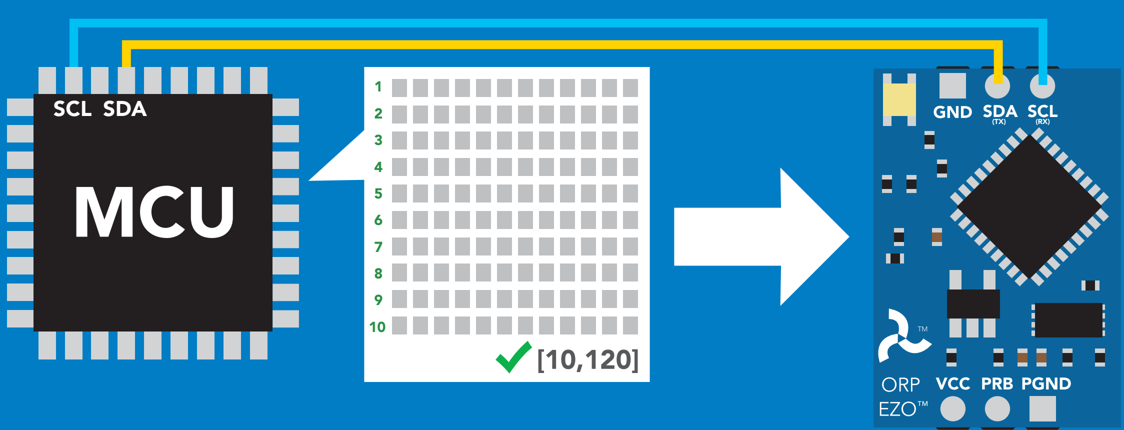
Command syntax

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

Import,n	**import calibration string to new device**

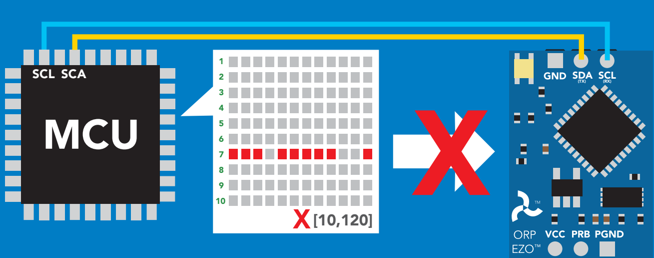
Example		Response
Import, 59 6F 75 20 61 72	(1 of 10)	Wait 300ms · 1 Dec · 0 Null
Import, 65 20 61 20 63 6F	(2 of 10)	Wait 300ms · 1 Dec · 0 Null
⋮		⋮
Import, 6F 6C 20 67 75 79	(10 of 10)	Wait 300ms · 1 Dec · 0 Null

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.**

AtlasScientific™ Environmental Robotics

Naming device

300ms processing delay

Command syntax

Do not use spaces in the name

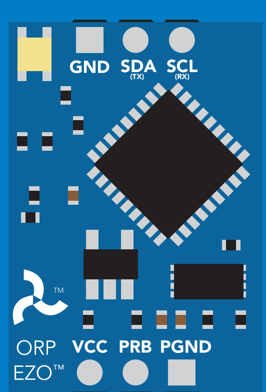
Name,n	set name
Name,	clears name
Name,?	show name

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

Up to 16 ASCII characters

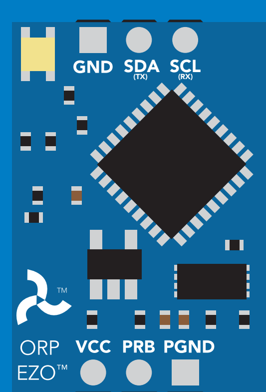
Example	Response
Name,	Wait 300ms · 1 Dec · 0 Null · name has been cleared
Name,zzt	Wait 300ms · 1 Dec · 0 Null
Name,?	Wait 300ms · 1 Dec · ?Name,zzt ASCII · 0 Null

Name,zzt



1 0

Name,?



1 ?Name,zzt 0

Device information

Command syntax

300ms processing delay

i	device information

Example

i

Response

Wait 300ms

1	?i,ORP, 19.7	0
Dec	ASCII	Null

Response breakdown

?i,	ORP,	1.97
	Device	Firmware

Reading device status

Command syntax

300ms processing delay

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

Example	Response
Status	Wait 300ms · 1 (Dec) · ?Status,P,5.038 (ASCII) · 0 (Null)

Response breakdown

?Status,	P,	5.038
	Reason for restart	Voltage at Vcc

Restart codes

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

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	Do not read status byte after issuing sleep command.
Any command	wakes up device	

	STANDBY	SLEEP
5V	16 mA	1.16 mA
3.3V	13.9 mA	0.995 mA

Standby **Sleep**

Protocol lock

Command syntax

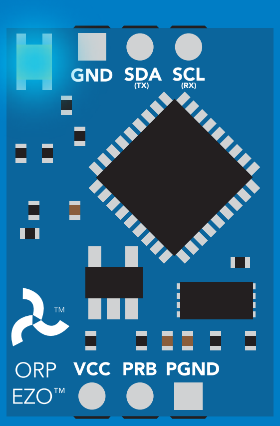
300ms processing delay

Plock,1	enable Plock	
Plock,0	disable Plock	default
Plock,?	Plock on/off?	

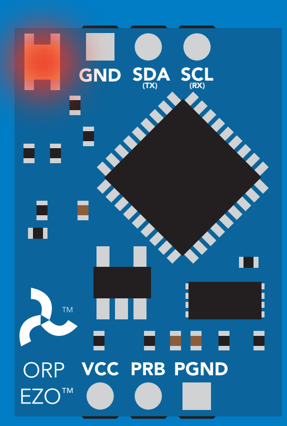
Locks device to I²C mode.

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)

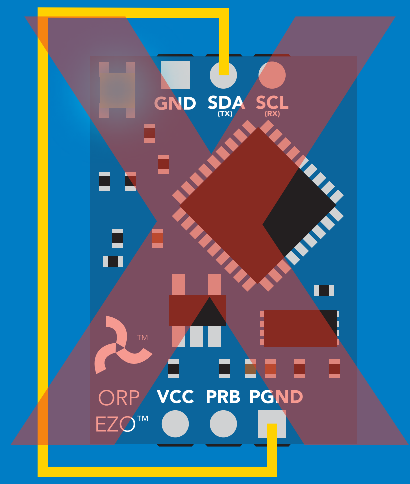
Plock,1



Baud, 9600



cannot change to UART



cannot change to UART

I²C address change

Command syntax

300ms processing delay

I2C,n sets I²C address and reboots into I²C mode

Example	Response
I2C,100	device reboot (no response given)

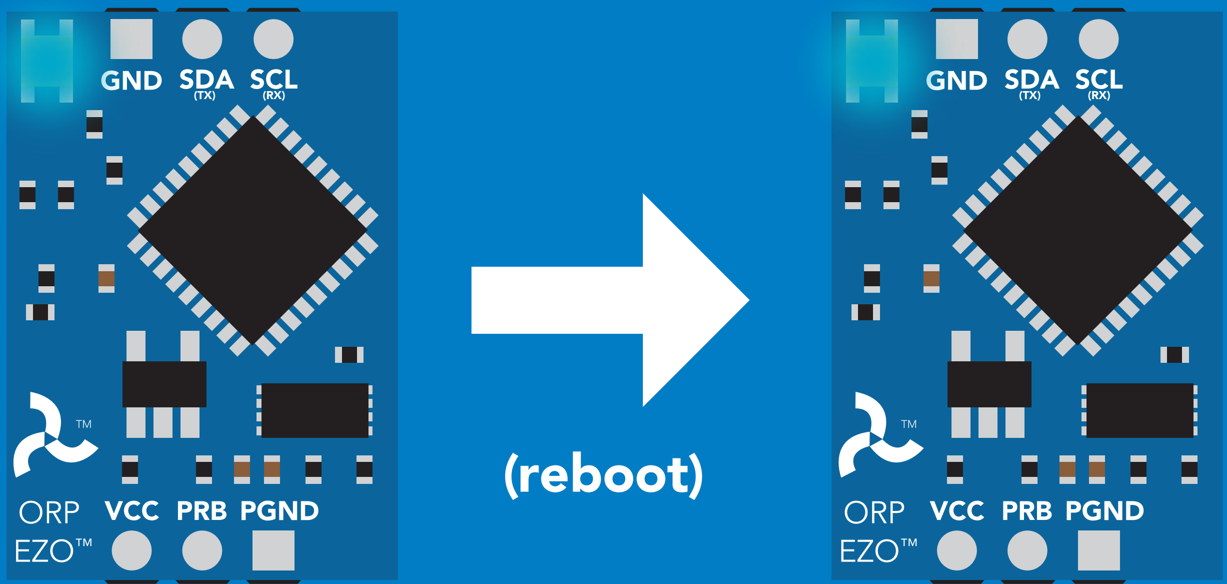
n = any number 1 – 127

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 98 (0x62).

I2C,100



Factory reset

Command syntax

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

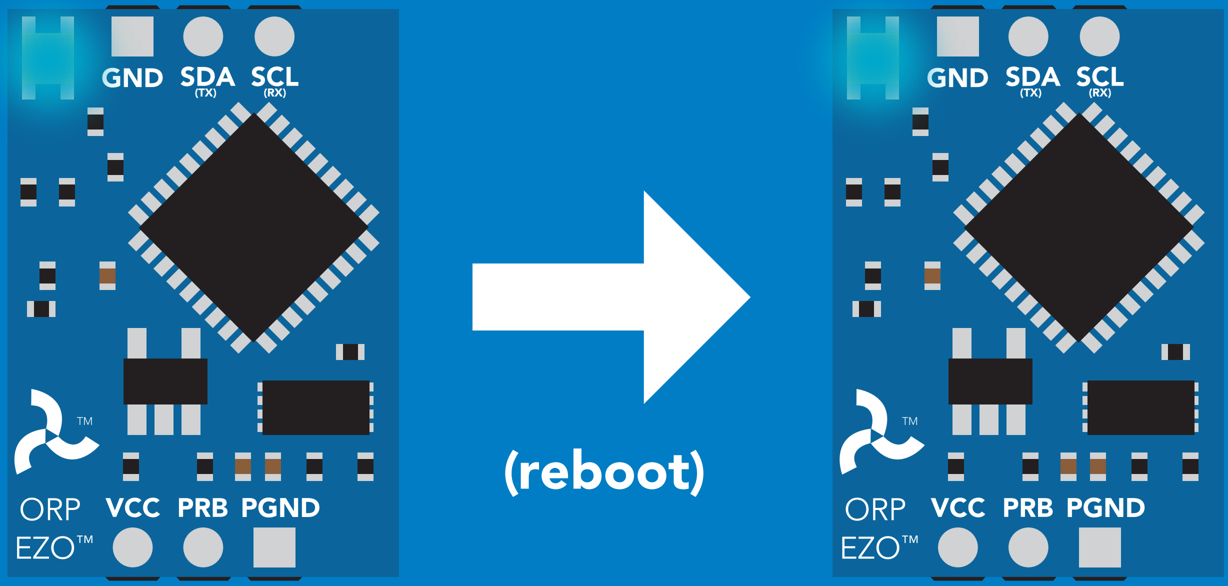
Factory enable factory reset

I²C address will not change

Example	Response
Factory	device reboot (no response given)

Clears calibration
LED on
Response codes enabled

Factory



AtlasScientific
Environmental Robotics

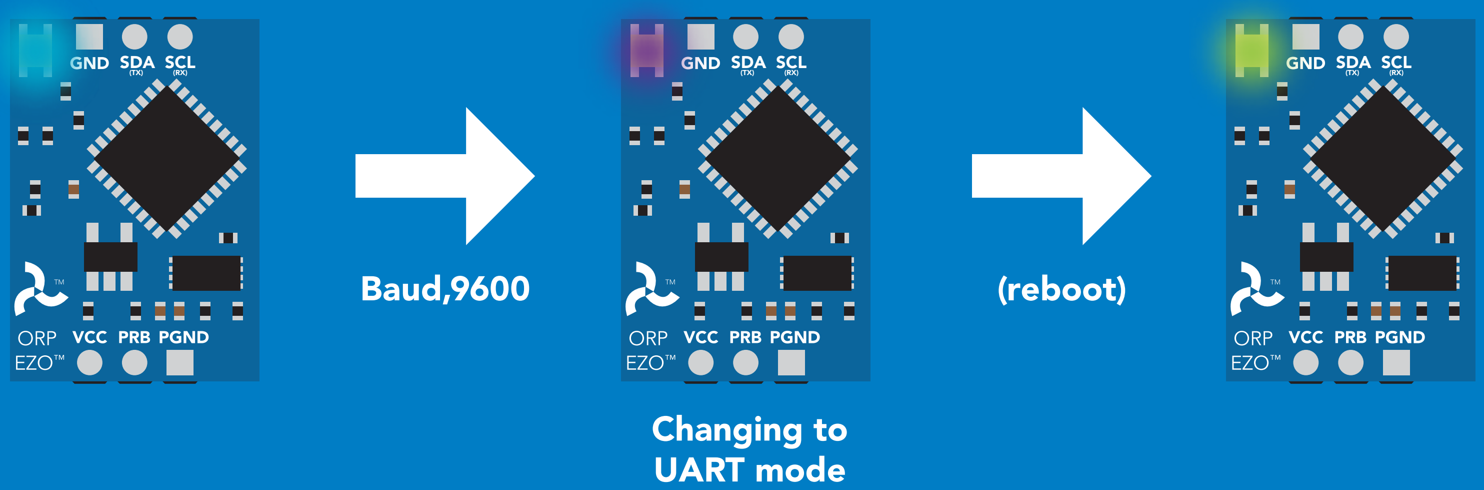
Change to UART mode

Command syntax

Baud,n switch from I²C to UART

Example	Response
Baud,9600	reboot in UART mode (no response given)

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

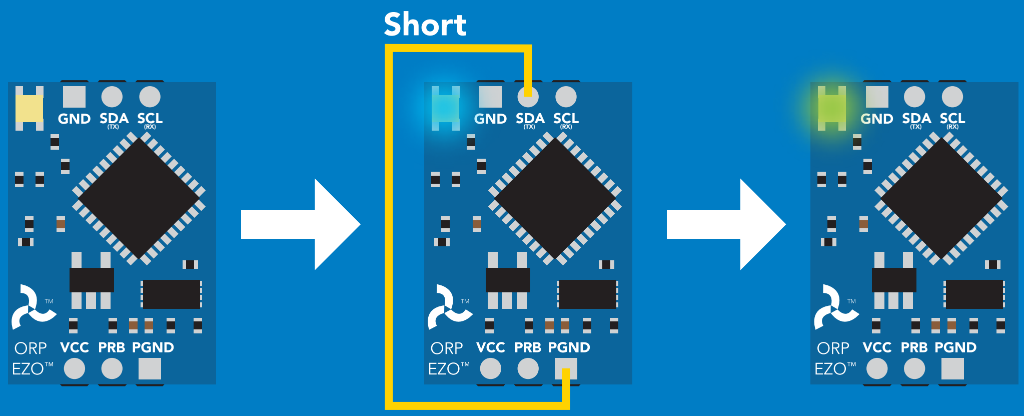


AtlasScientific
Environmental Robotics
r 0.1

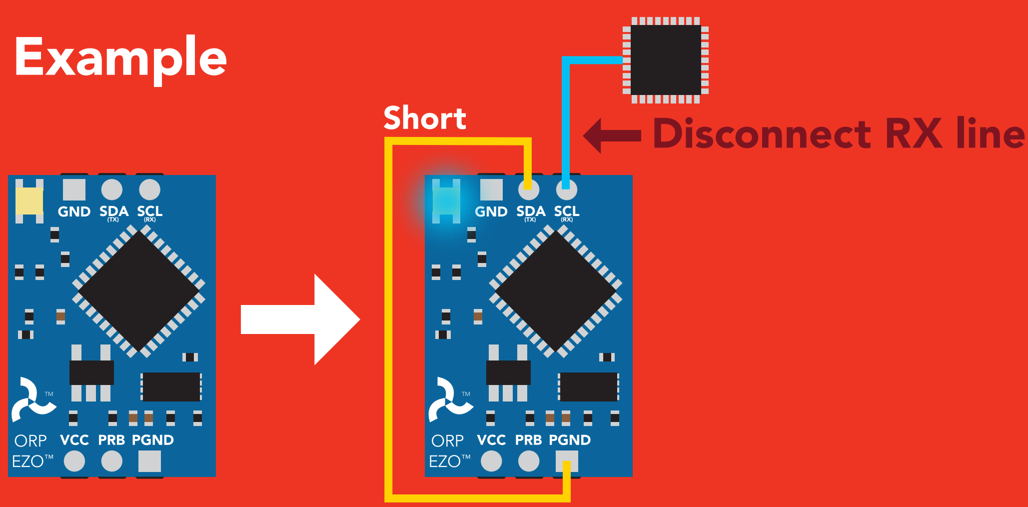
Manual switching to UART

- 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

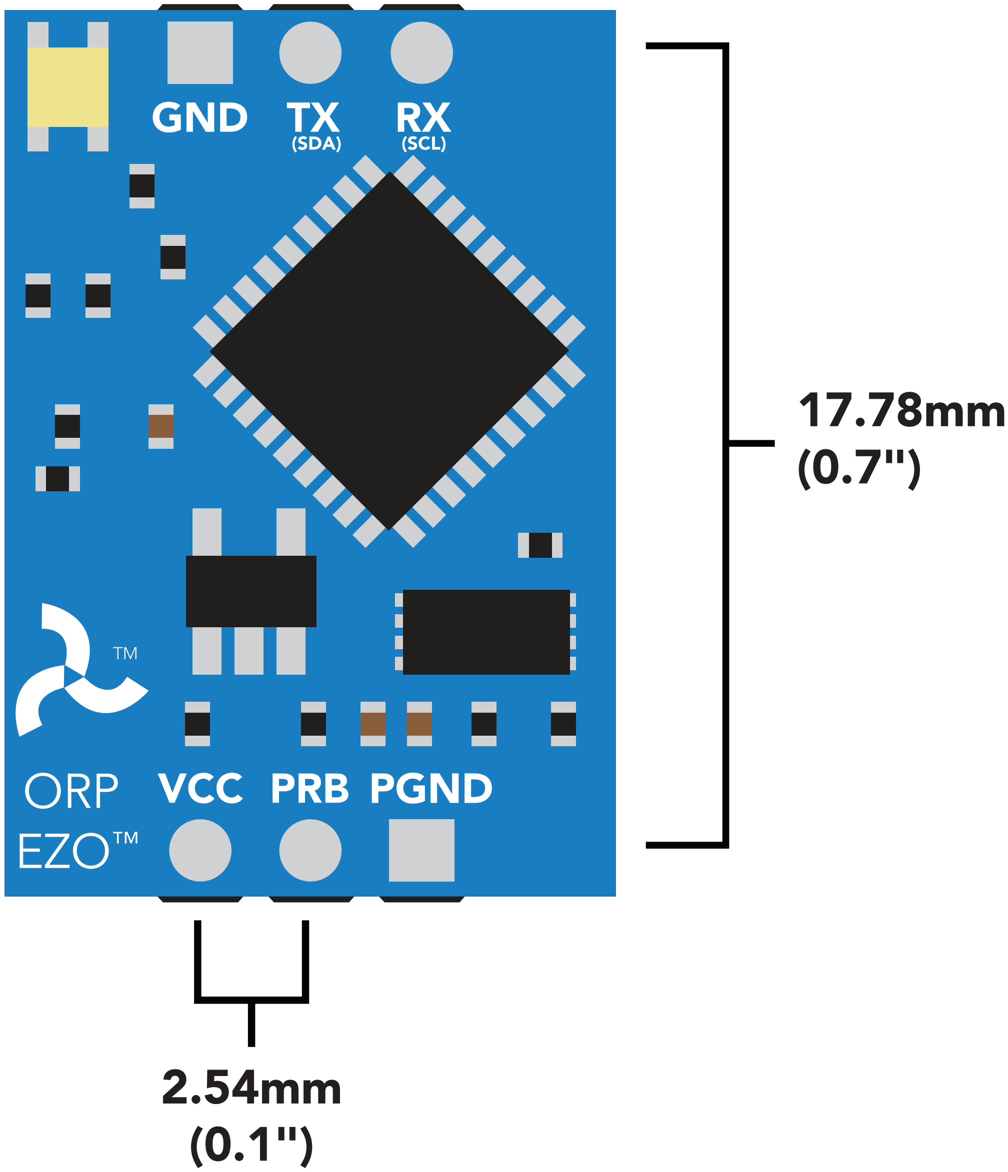
Example



Wrong Example



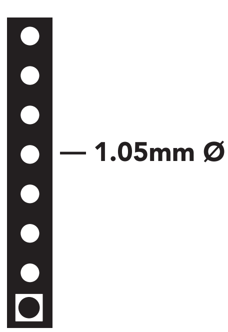
EZO™ circuit footprint

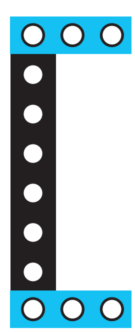


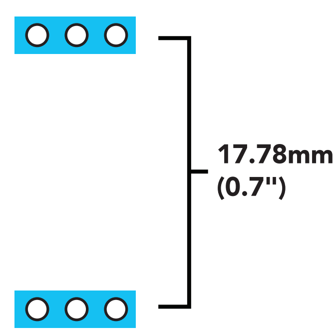
1 In your CAD software place a 8 position header.

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.







r 0.1

Datasheet change log

Datasheet V 5.0

Revised naming device info on pages 29 & 52.

Datasheet V 4.9

Revised single point calibration information and art on pg 13.

Datasheet V 4.8

Moved Default state to pg 14.

Datasheet V 4.7

Updated firmware to V2.11 on pg 63.

Datasheet V 4.6

Revised response for the sleep command in UART mode on pg 33.

Datasheet V 4.5

Revised calibration theory on page 12, and added more information on the Export calibration and Import calibration commands.

Datasheet V 4.4

Revised isolation schematic on pg. 10

Datasheet V 4.3

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

Datasheet V 4.2

Removed note from certain commands about firmware version.

Datasheet V 4.1

Added information to calibration theory on pg 8.

Datasheet V 4.0

Revised definition of response codes on pg 42.

Datasheet V 3.9

Revised isolation information on pg 9.

Datasheet V 3.8

Revised Plock pages to show default value.

Datasheet V 3.7

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 3.6

Revised circuit illustrations throughout datasheet.

Datasheet V 3.5

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

Datasheet V 3.4

Revised entire datasheet.

Firmware updates

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

V1.6 – I^2C bug (Dec 1, 2014)
- Fixed I^2C bug where the circuit may inappropriately respond when other I^2C 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)
- Fixed bug in the cal clear command, improves how it calculates the ORP
- Added calibration saving and loading

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 – (July 17, 2017)
- Fixed bug where calibration would restore itself after restart, despite being cleared.

V2.12 – (Oct 18, 2021)
- Internal update for new part compatibility.

V2.13 – (Nov 12, 2021)
- Fixed bug in I2C mode with timing and sleep mode.

Warranty

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

The debugging phase

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

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

Atlas Scientific™
Environmental Robotics

V 6.1

Revised 2/24

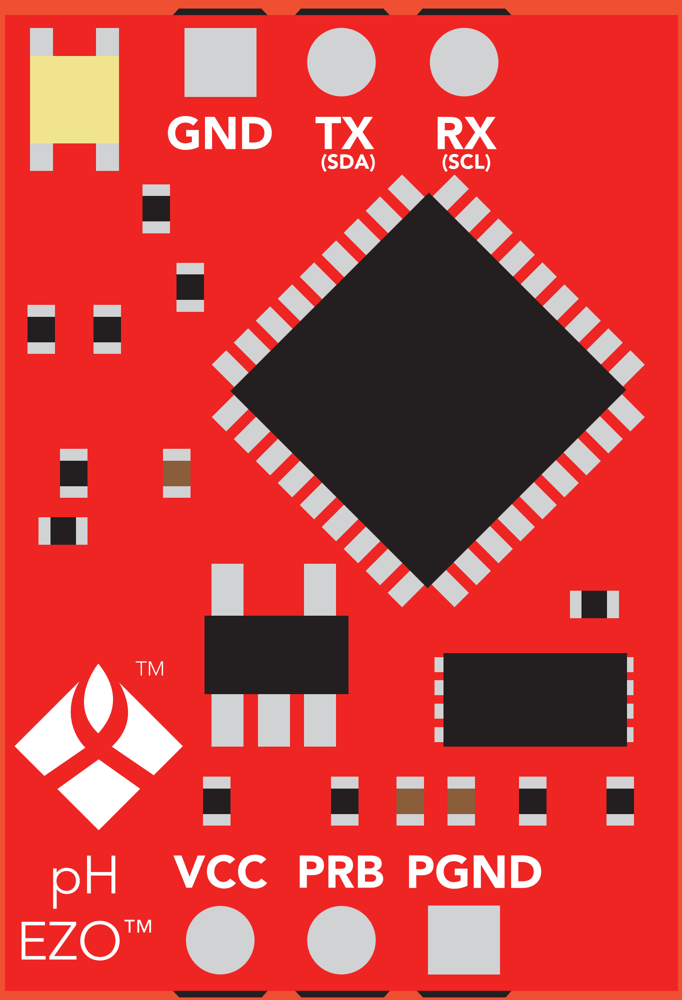
EZO-pH™

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 & I^2C**
Default I^2C address	**99 (0x63)**
Operating voltage	**3.3V – 5V**
Data format	**ASCII**

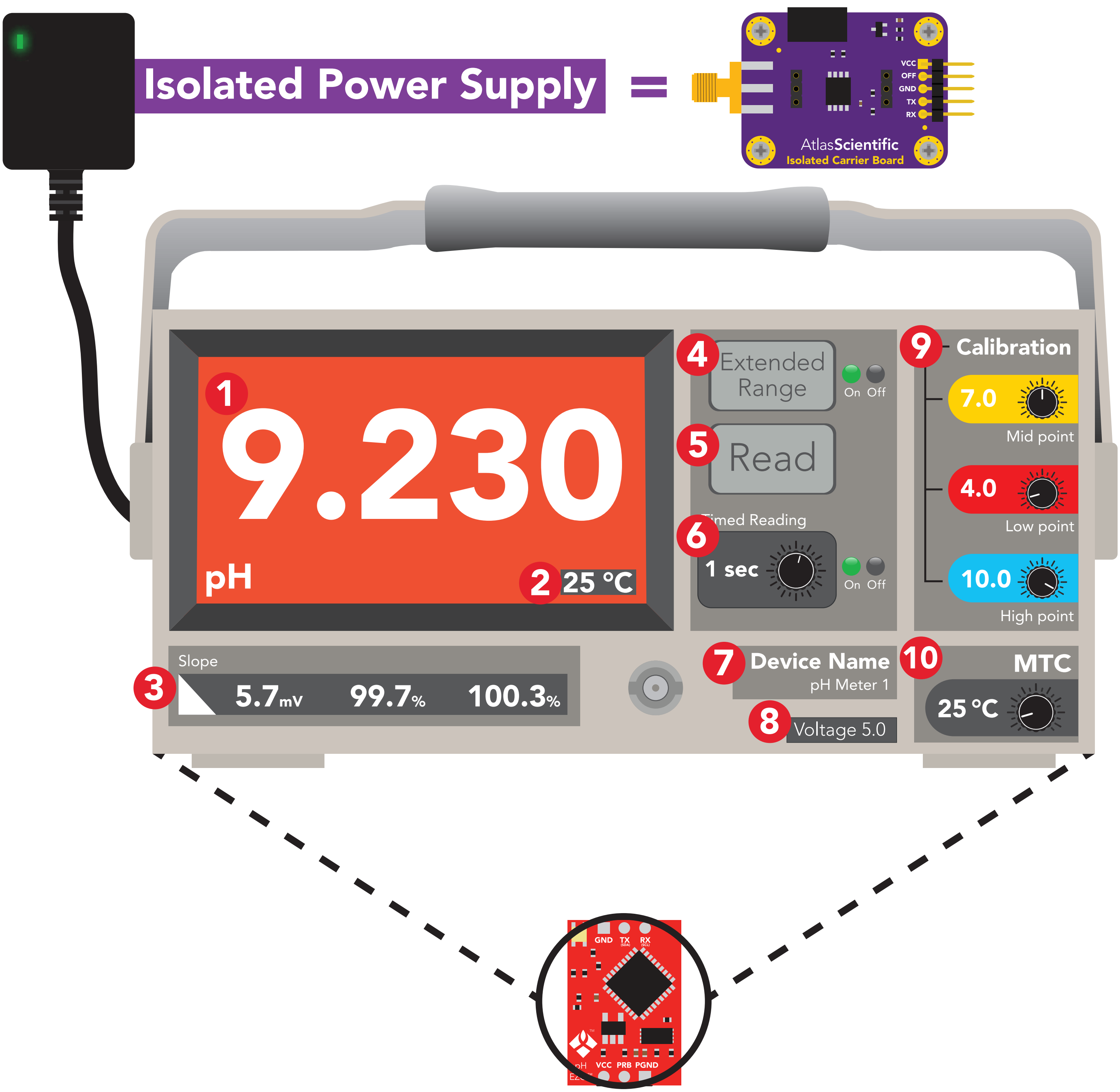


PATENT PROTECTED

Written by Jordan Press
Designed by Noah Press

This is an evolving document, check back for updates.

The EZO™ pH Circuit has all the features of this bench top meter.



1 Three decimal pH reading
2 Temperature used for reading
3 Calibration slope
4 Extended range capability
5 Immediate reading
6 Timed readings
7 Set device name
8 Voltage usage
9 Multi point calibration
10 Manual temperatue compensation

The EZO™ pH Circuit is compatible with any brand of pH probe.

Available data protocols

UART

Default

I^2C

Unavailable data protocols

SPI

Analog

RS-485

Mod Bus

4–20mA

STOP

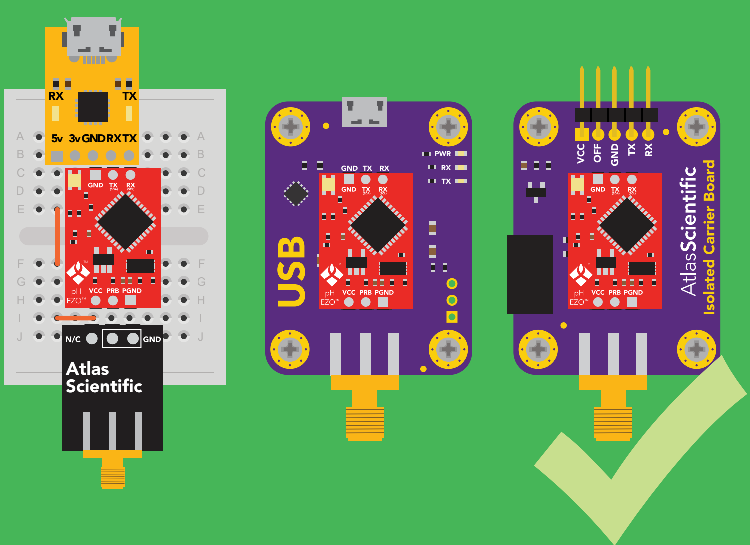
SOLDERING THIS DEVICE VOIDS YOUR WARRANTY.

Are there specific soldering instructions? **Yes, see page 71.**

Can you make a warranty claim after soldering? No.

If you have not used this product before; Observe how a properly working sensor behaves *BEFORE* embedding it into your PCB.

Get this device working using one of these methods first.



Do not embed before you have experience with this sensor.

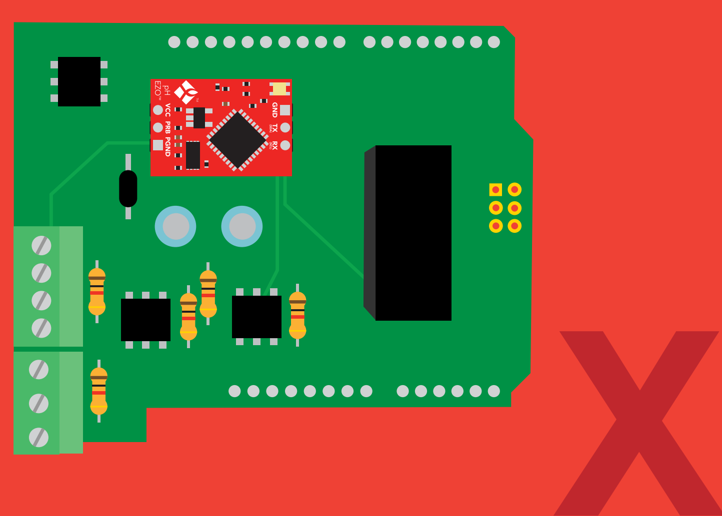
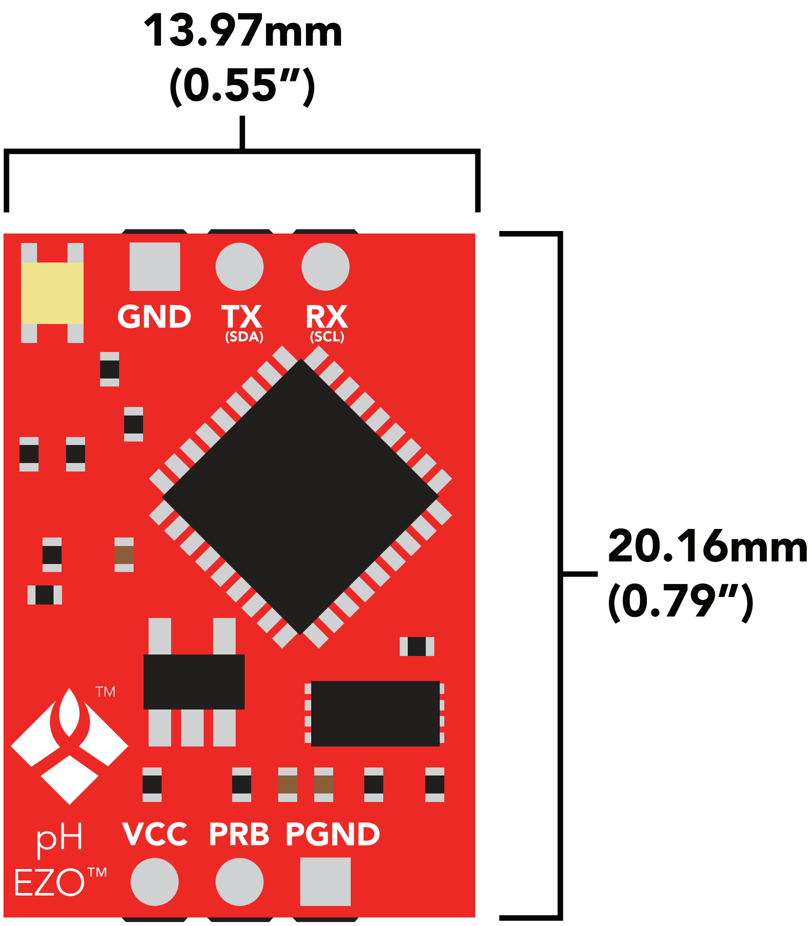


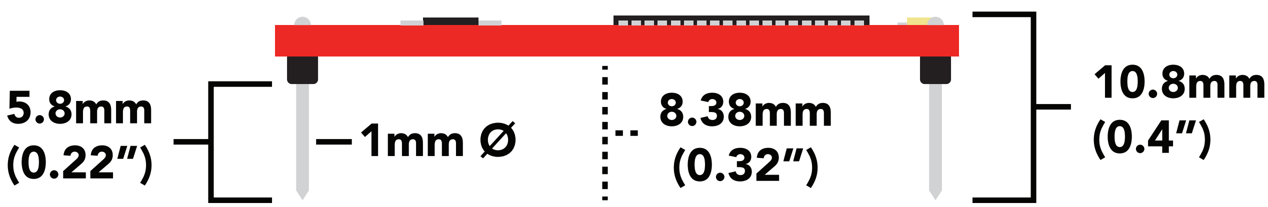
Table of contents

UART

I²C

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

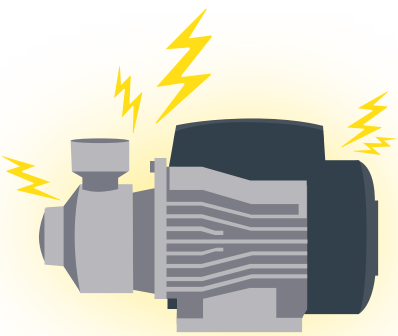
Parameter	MIN	TYP	MAX
Storage temperature (EZO™ pH)	-65 °C		125 °C
Operational temperature (EZO™ pH)	-40 °C	25 °C	85 °C
VCC	3.3V	5V	5.5V

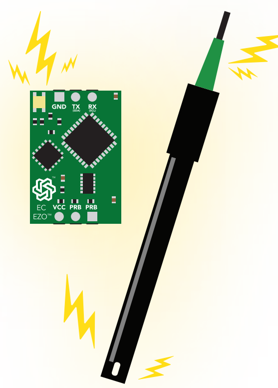
AtlasScientific™
Environmental Robotics

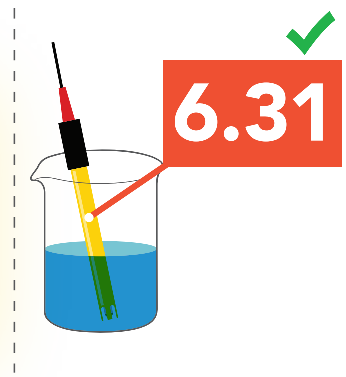
Electrical isolation

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.

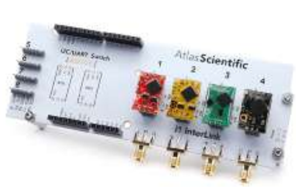
Atlas Scientific offers several different electrical isolation products that can be used in your design. Select the electrical isolation product that works best for your design.

Basic EZO™
Inline Voltage Isolator

Vertical Isolator

Electrically Isolated
EZO™ Carrier Board

Gen 2 Electrically Isolated
USB EZO™ Carrier Board



i1 InterLink

i2 InterLink

i3 InterLink

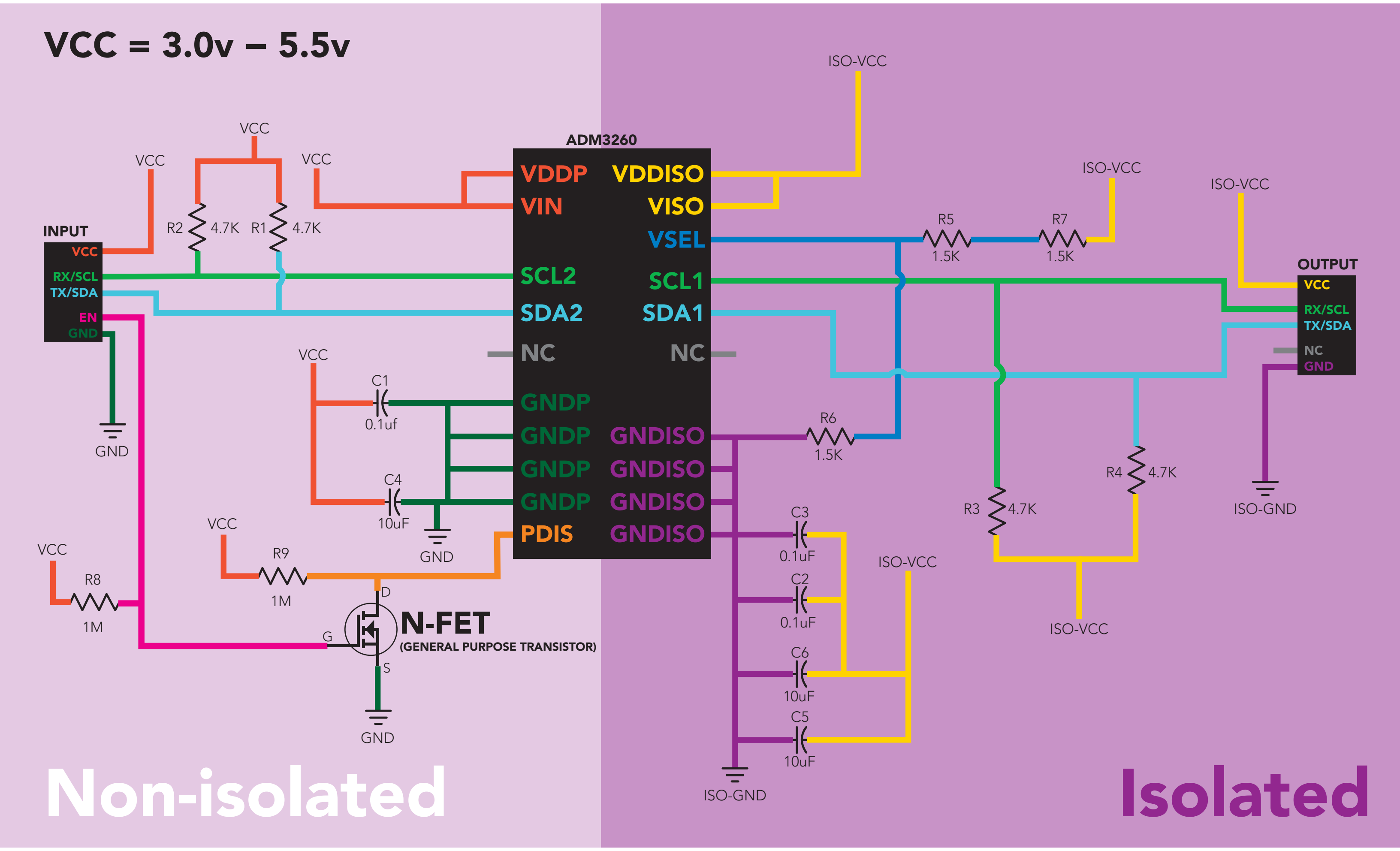
Electrically Isolated EZO™
Carrier Board (old style)

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.

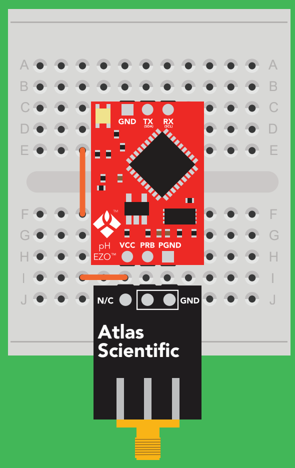
This isolation schematic is based on the ADM3260, which can output up to 150 mW of isolated power. PCB layout requires special attention for EMI/EMC and RF Control. Having good ground planes and keeping the capacitors as close to the chip as possible are crucial for proper performance.

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.

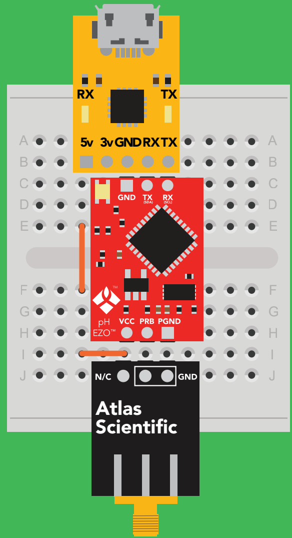
Isolated ground is different from non-isolated ground, these two lines should not be connected together.



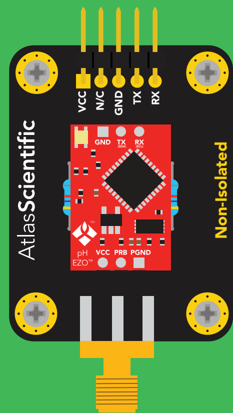
Correct wiring



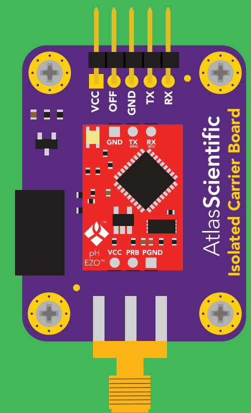
Bread board



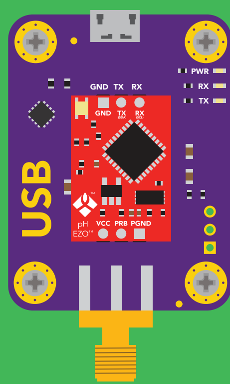
Bread board via USB



Non-Isolated
EZO™ Carrier Board



Electrically Isolated
EZO™ Carrier Board

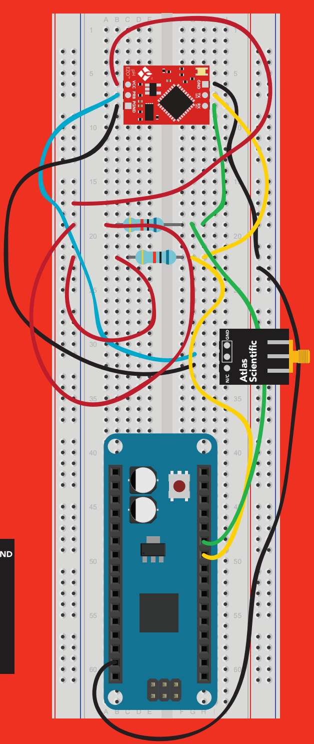


USB
carrier board

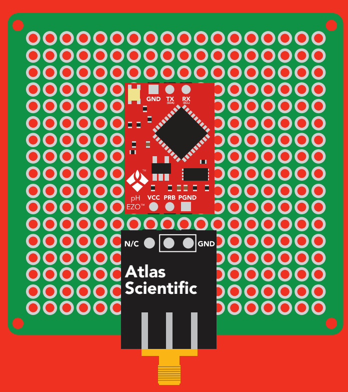
Incorrect wiring

Extended leads

Sloppy setup

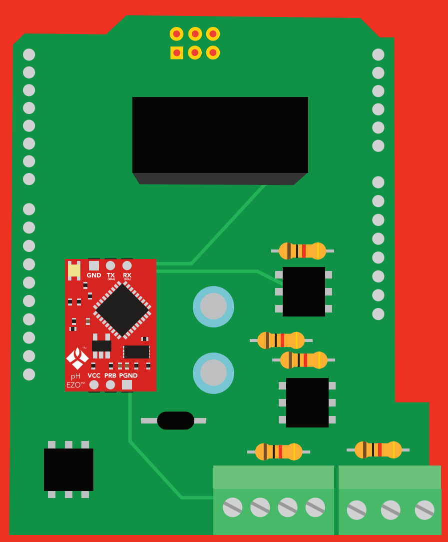


Perfboards or Protoboards



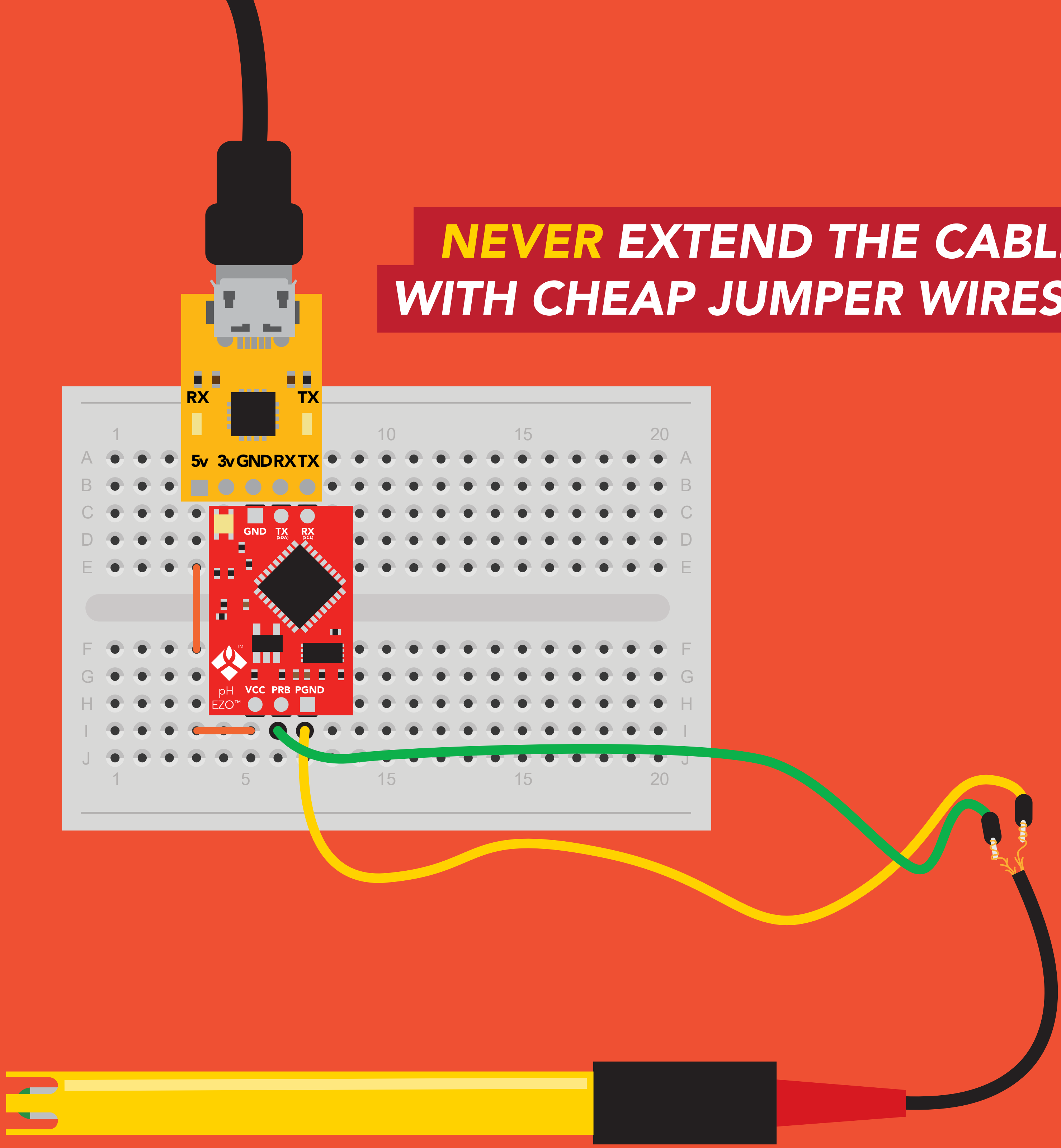
NEVER
use Perfboards or Protoboards
Flux residue and shorting wires make it very hard to get accurate readings.

*Embedded into your device



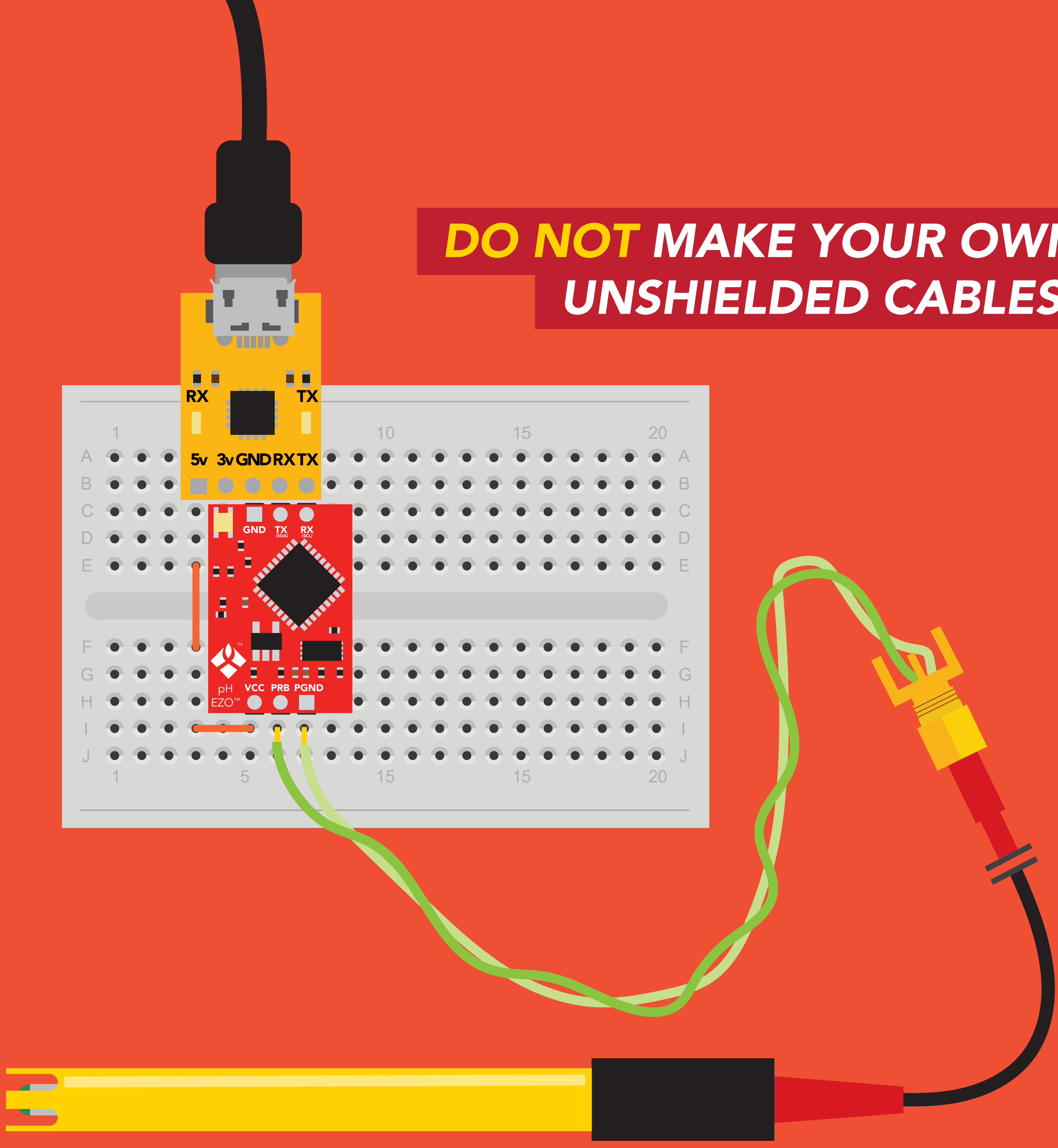
***Only after you are familar with EZO™ circuits operation**

NEVER EXTEND THE CABLE WITH CHEAP JUMPER WIRES!



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

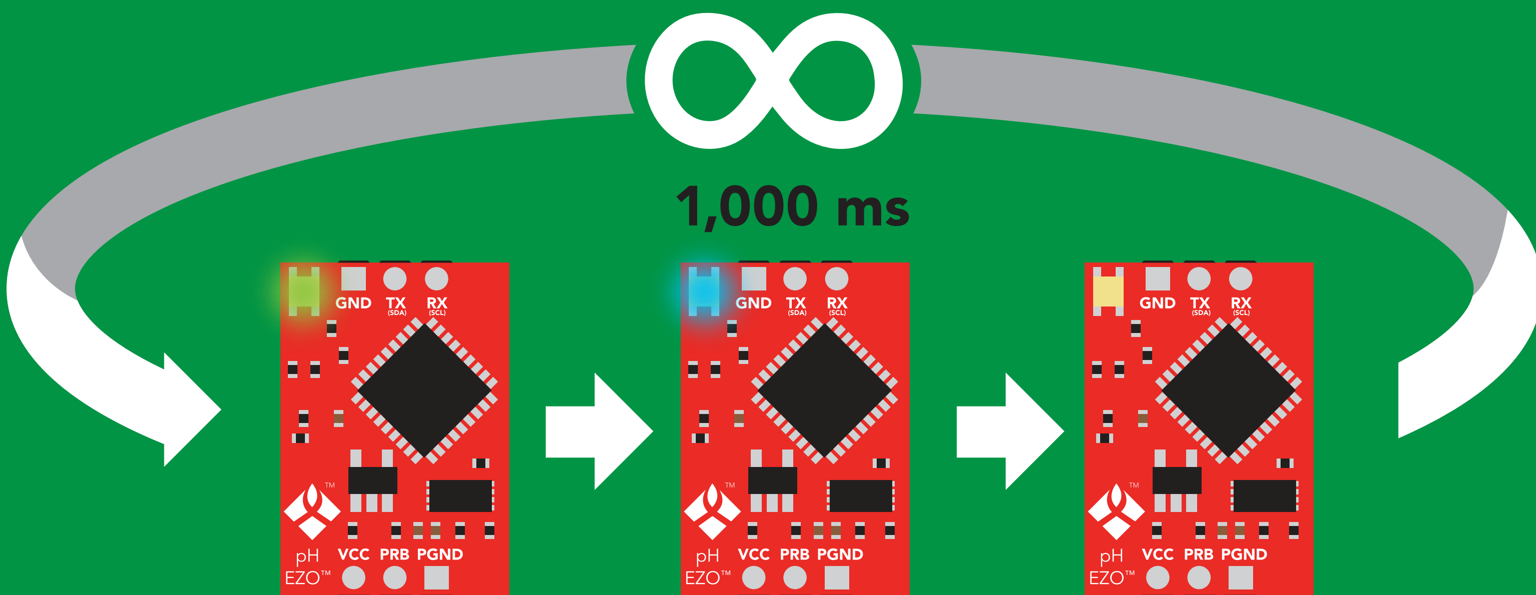
DO NOT MAKE YOUR OWN UNSHIELDED CABLES!



ONLY USE SHIELDED CABLES.

Default state
UART mode

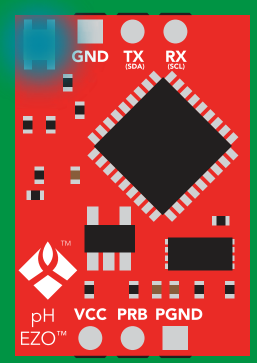
Baud	**9,600**
Readings	**continuous**
Units	**pH**
Speed	**1 reading per second**
LED	**on**



Green
Standby

Cyan
Taking reading

Transmitting



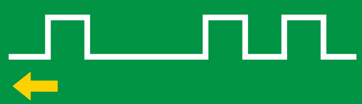
Solid Blue LED
in I²C mode
Not UART ready

UART mode

8 data bits
1 stop bit
no parity
no flow control

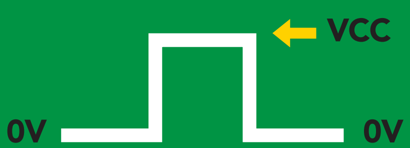
Baud
300
1,200
2,400
9,600 default
19,200
38,400
57,600
115,200

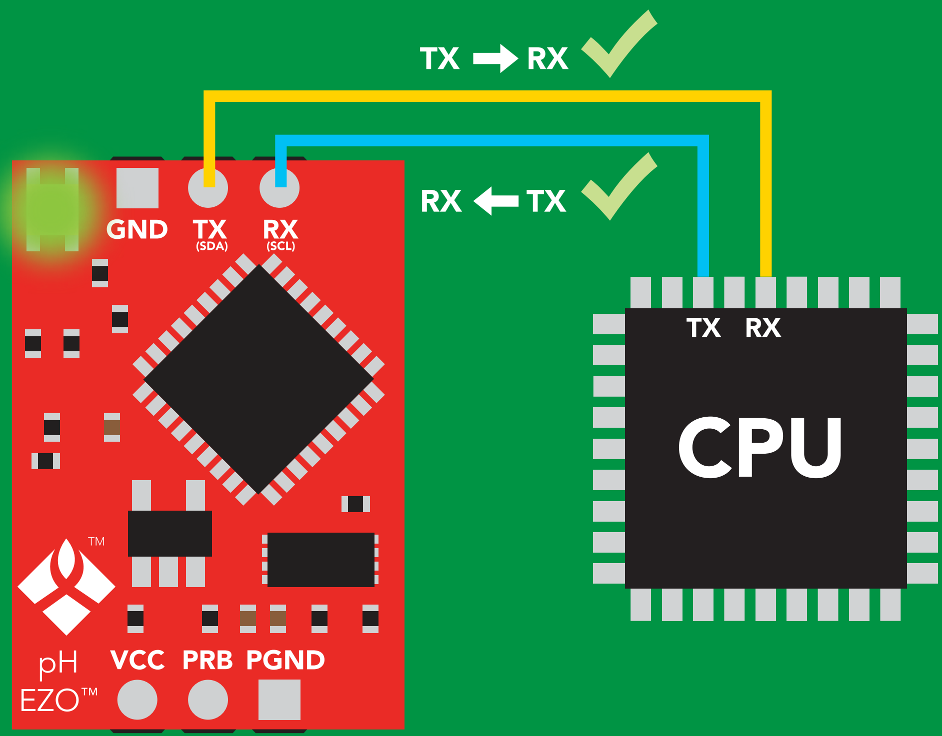
RX
Data in



TX
Data out

Vcc 3.3V – 5.5V

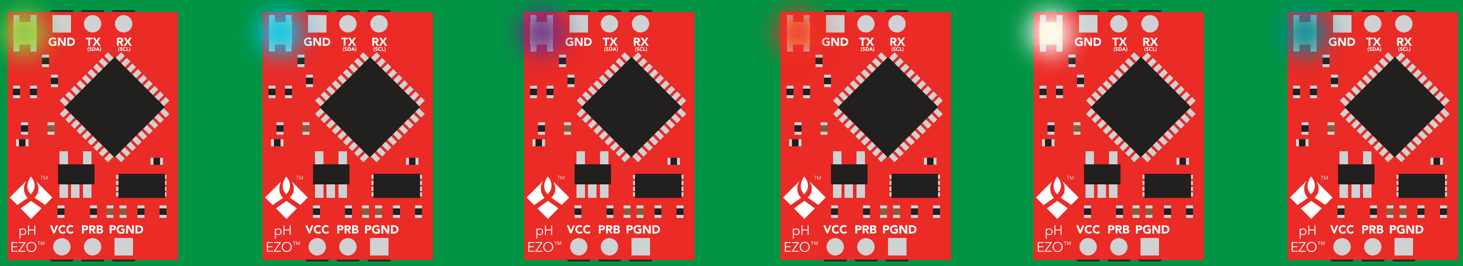




Data format

Reading	pH	**Data type**	floating point
Encoding	ASCII	**Decimal places**	3
Format	string	**Smallest string**	4 characters
Terminator	carriage return	**Largest string**	40 characters

LED color definition



Green	Cyan	Purple	Red	White	Blue
UART standby	Taking reading	Changing baud rate	Command not understood	Find	I2C standby

	LED ON
5V	**+2.2 mA**
3.3V	**+0.6 mA**

Settings that are retained if power is cut

Baud rate
Calibration
Continuous mode
Device name
Enable/disable response codes
Hardware switch to I²C mode
LED control
Protocol lock
Software switch to I²C mode

Settings that are *NOT* retained if power is cut

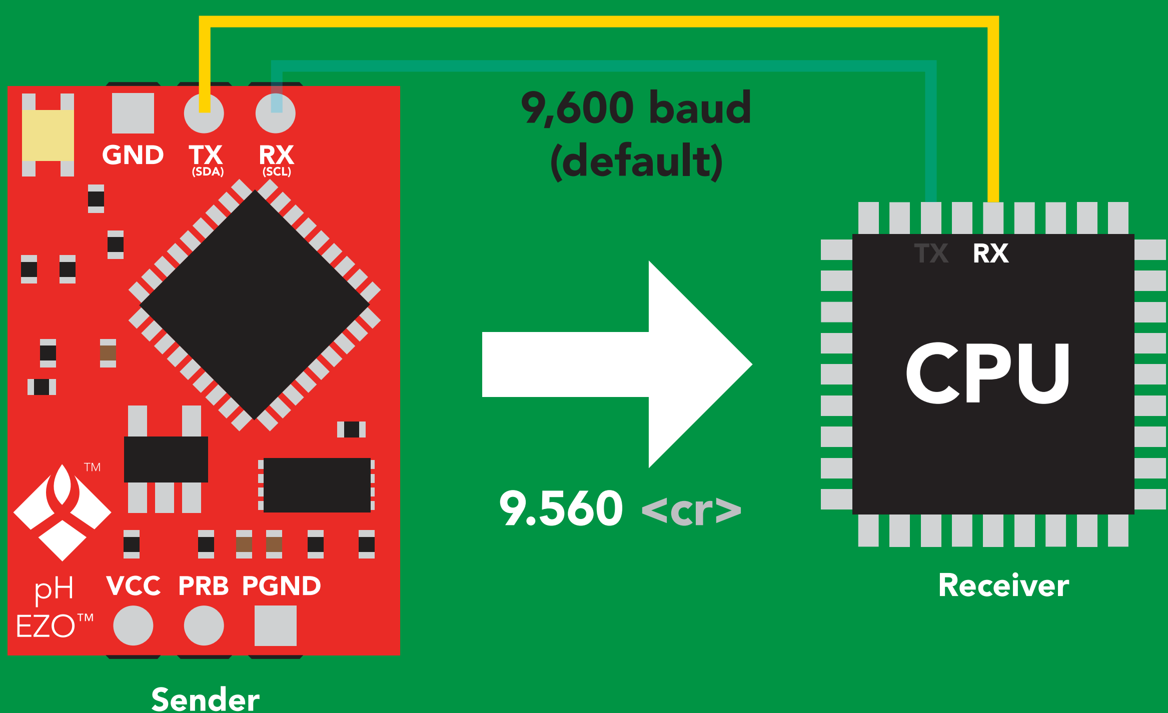
Find
Sleep mode
Temperature compensation

AtlasScientific™ Environmental Robotics

Receiving data from device

2 parts

ASCII data string	Carriage return <cr>
Command	Terminator



Advanced

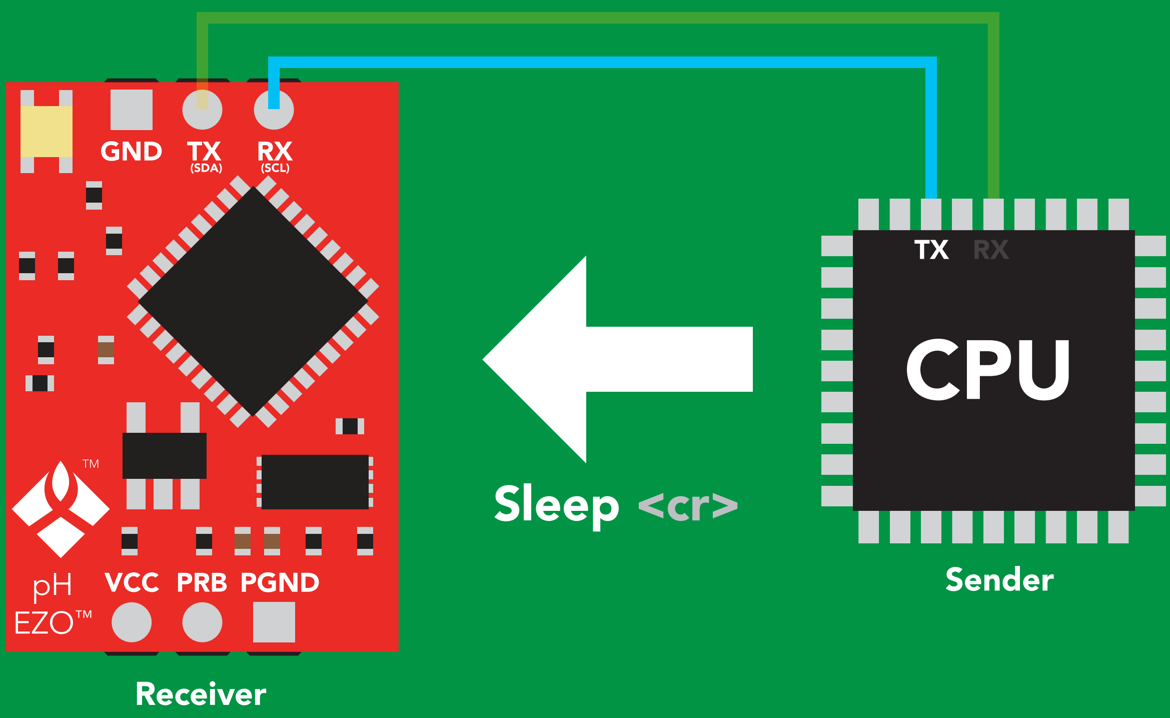
ASCII:	9	.	5	6	0	<cr>
Hex:	39	2E	35	36	30	0D
Dec:	57	46	53	54	48	13

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

AtlasScientific™ Environmental Robotics

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 I^2C 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

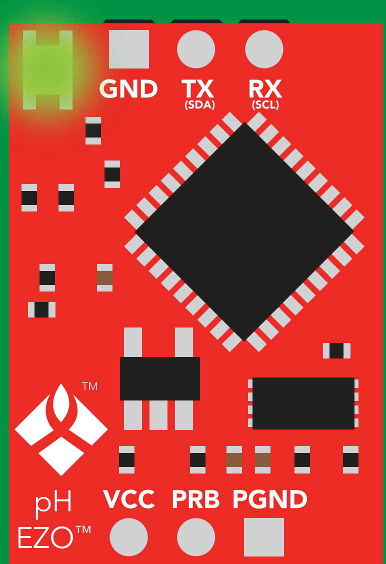
Command syntax

L,1 <cr> **LED on** **default**

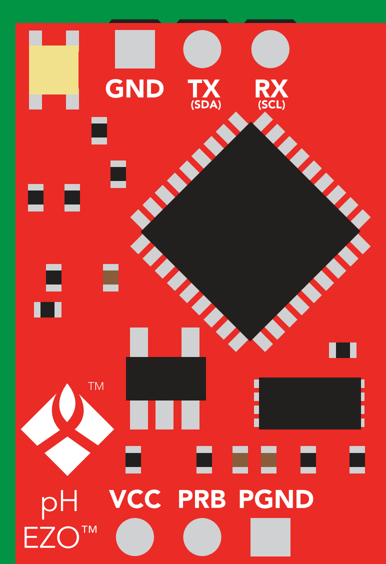
L,0 <cr> **LED off**

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> *OK <cr>



L,1



L,0

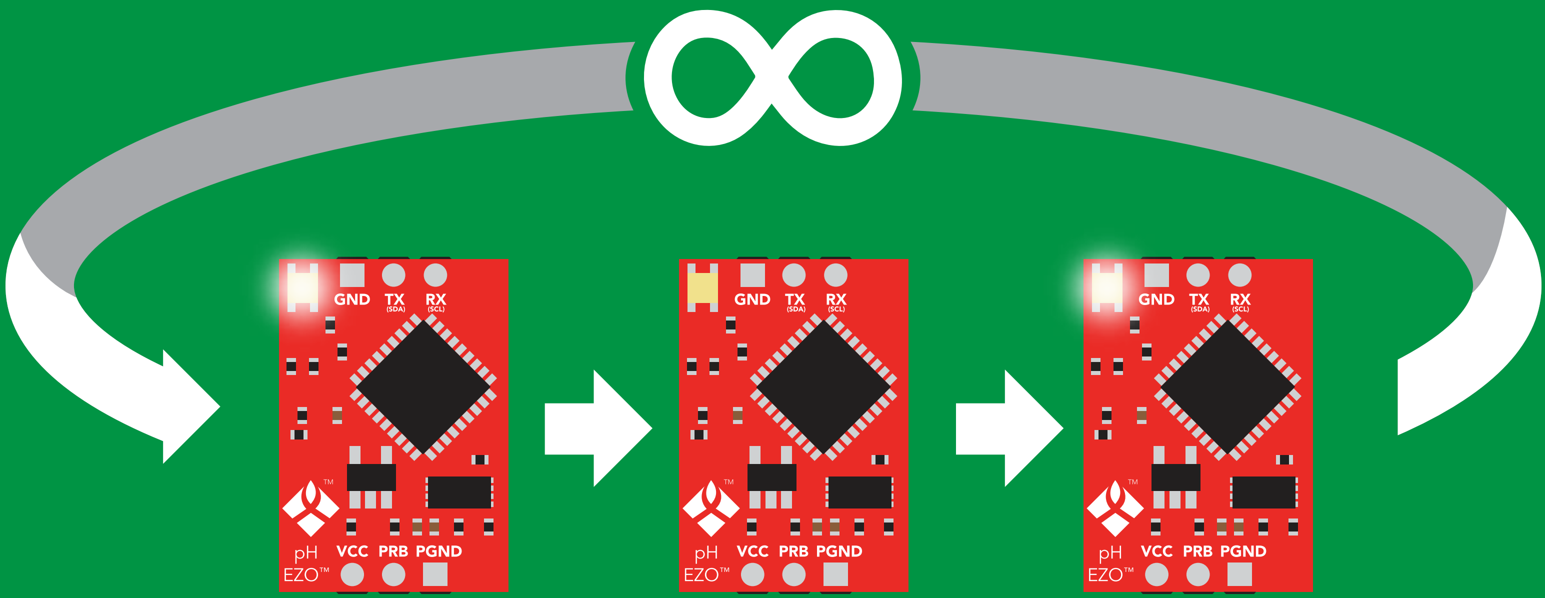
Find

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>



AtlasScientific
Environmental Robotics

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	Response
C,1 <cr>	*OK <cr> pH (1 sec) <cr> pH (2 sec) <cr> pH (n sec) <cr>
C,30 <cr>	*OK <cr> pH (30 sec) <cr> pH (60 sec) <cr> pH (90 sec) <cr>
C,0 <cr>	*OK <cr>
C,? <cr>	?C,1 <cr> or ?C,0 <cr> or ?C,30 <cr> *OK <cr>

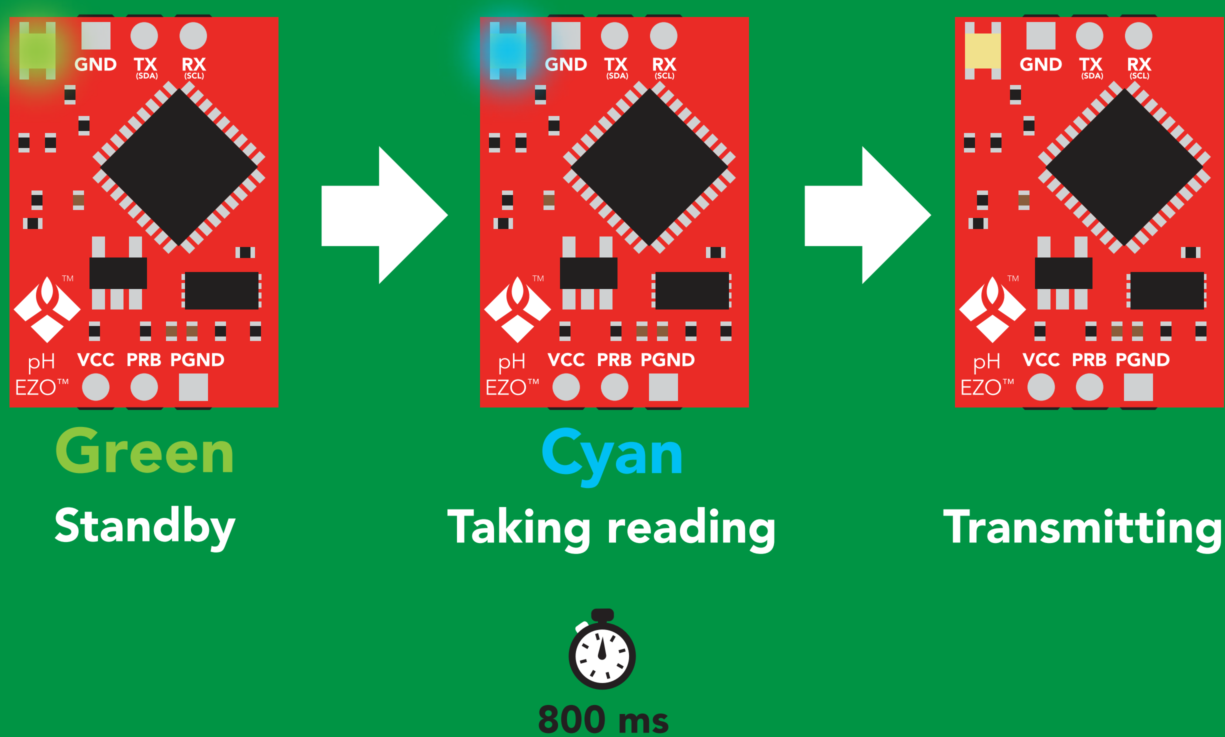
Single reading mode

Command syntax

A single reading takes 800ms

R <cr> **takes single reading**

Example	Response
R <cr>	9.560 <cr> *OK <cr>



Calibration

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	<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: Use this command to download calibration settings

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

Example	Response
Export,? <cr>	**10,120** <cr>

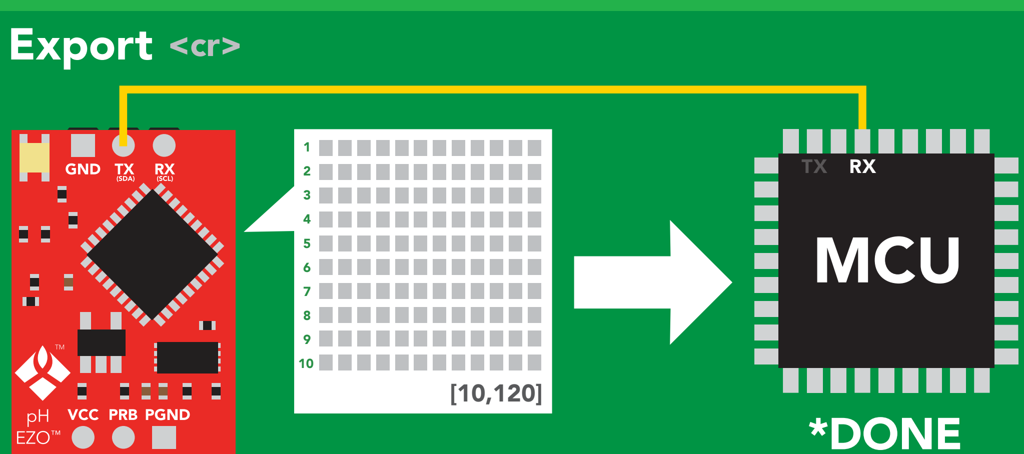
Response breakdown

10, — # of strings to export
120 — # of bytes to export

Export strings can be up to 12 characters long, and is always followed by <cr>

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

Disabling *OK simplifies this process



AtlasScientific™
Environmental Robotics

Import calibration

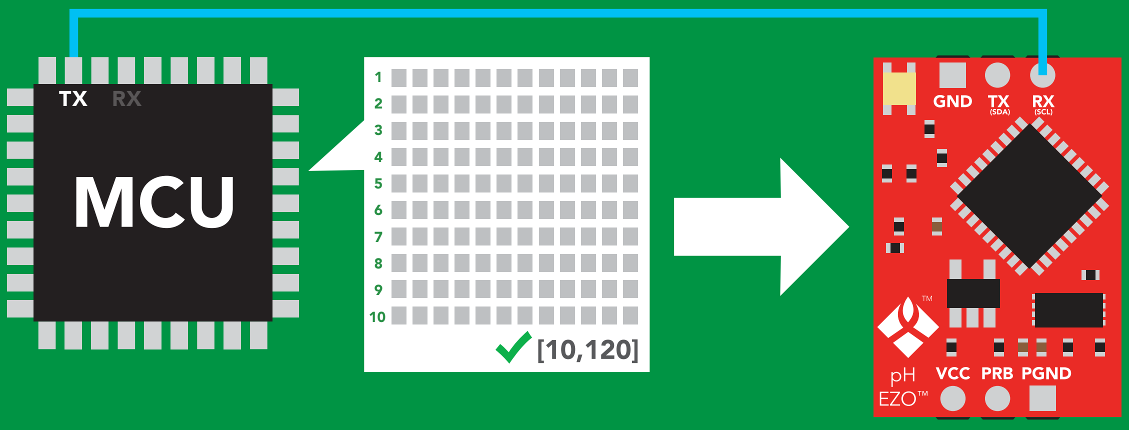
Command syntax

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

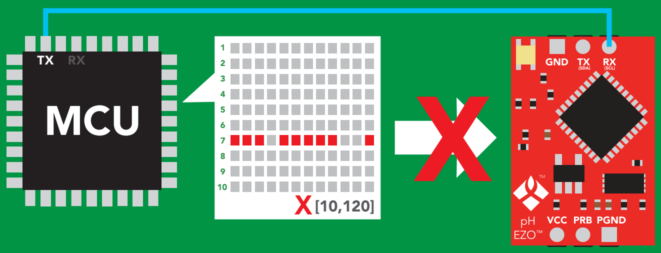
Import,n	<cr>	import calibration string to new device

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>

Import,n <cr>



*OK <cr>
system will reboot



*ER <cr>

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

AtlasScientific Environmental Robotics

Slope

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.

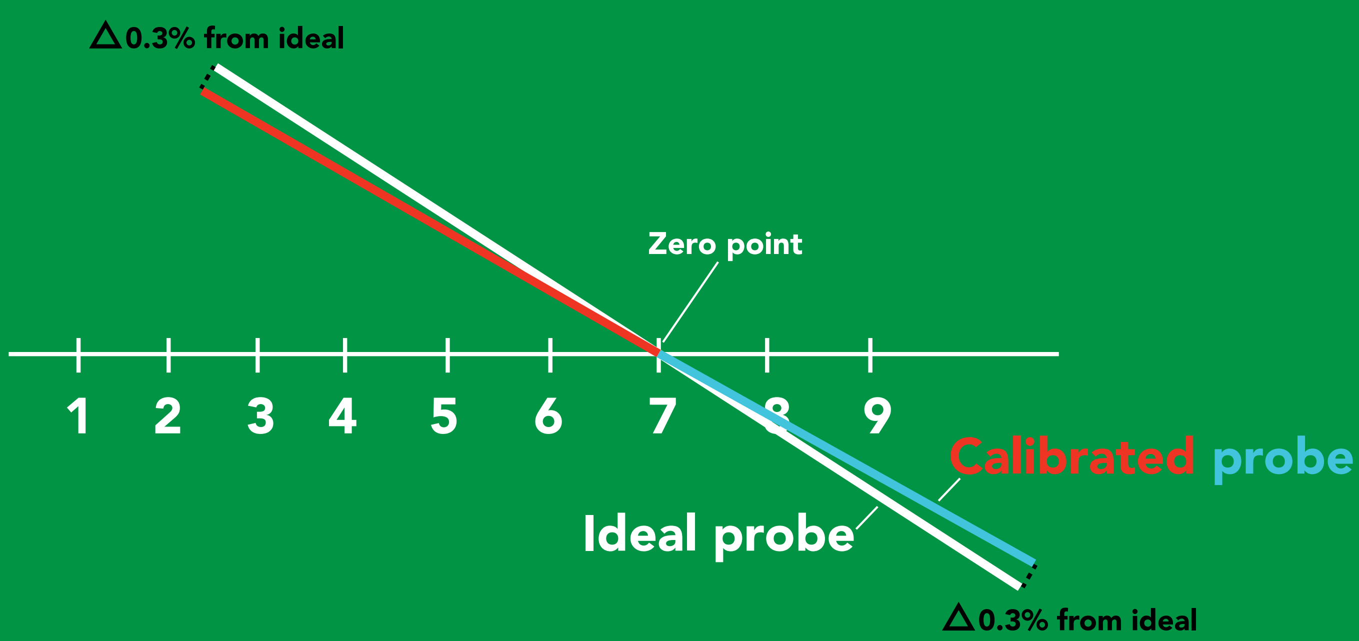
Command syntax

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

Example	Response
Slope,? <cr>	?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.



Atlas Scientific
Environmental Robotics

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>

pHext,0

pH = 0.000

pHext,1

pH = -1.220

Temperature compensation

Command syntax

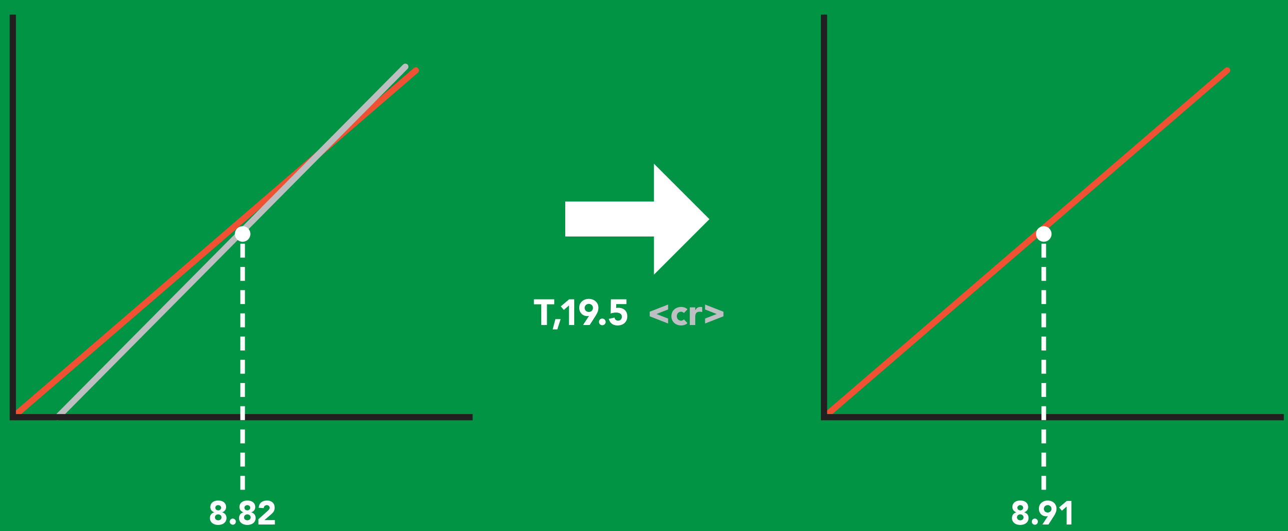
Default temperature = 25°C
Temperature is always in Celsius
Temperature is not retained if power is cut

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> 8.91 <cr>
T,? <cr>	?T,19.5 <cr> *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

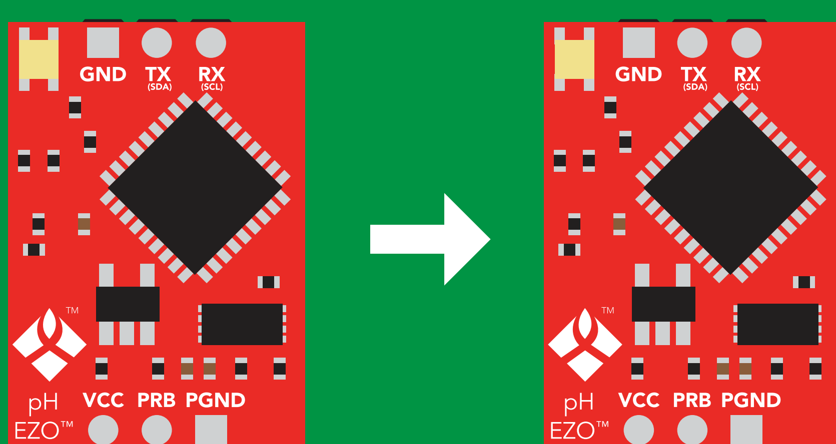
n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

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> *OK <cr>

Name,zzt

Name,?



*OK <cr>

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

Atlas Scientific
Environmental Robotics

Device information

Command syntax

i <cr> **device information**

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

Response breakdown

?i,	pH,	2.16
	Device	Firmware

AtlasScientific
Environmental Robotics

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> *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> *OK <cr>

Response breakdown

?Status,	P,	5.038
	Reason for restart	Voltage at Vcc

Restart codes

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

AtlasScientific
Environmental Robotics

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> *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

Standby
16 mA

Sleep
1.16 mA

AtlasScientific
Environmental Robotics

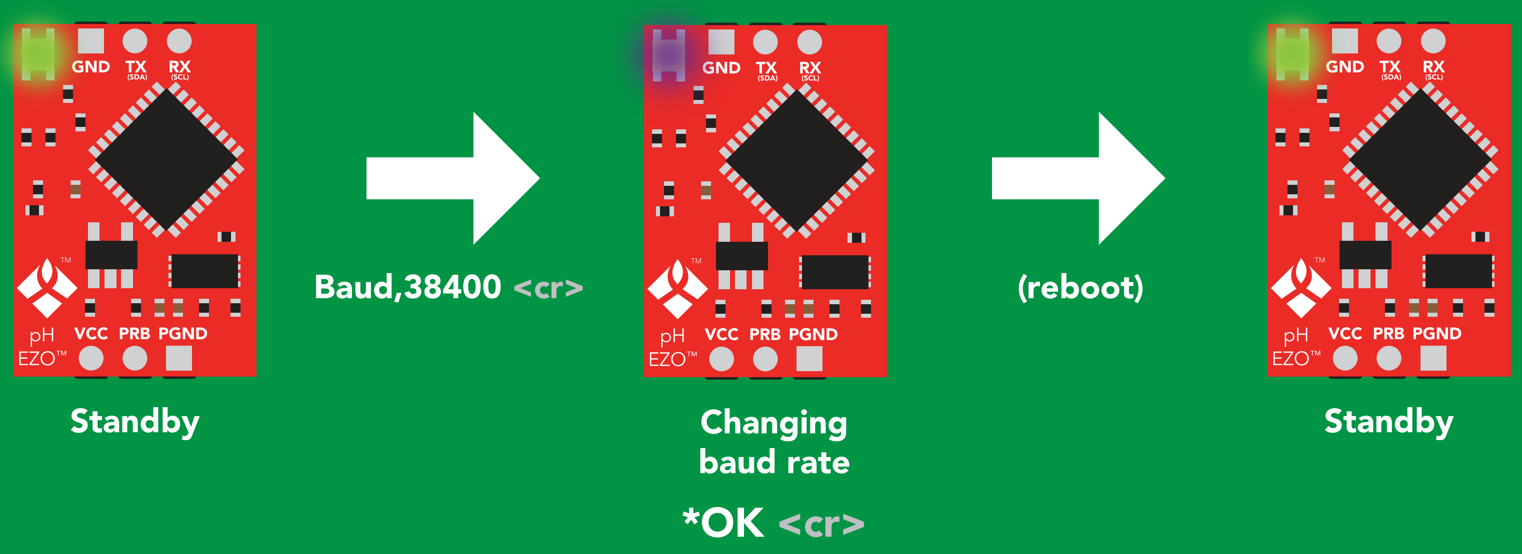
Change baud rate

Command syntax

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

Example	Response
Baud,38400 <cr>	*OK <cr>
Baud,? <cr>	?Baud,38400 <cr> *OK <cr>

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



AtlasScientific™ Environmental Robotics

Protocol lock

Command syntax

Locks device to UART mode.

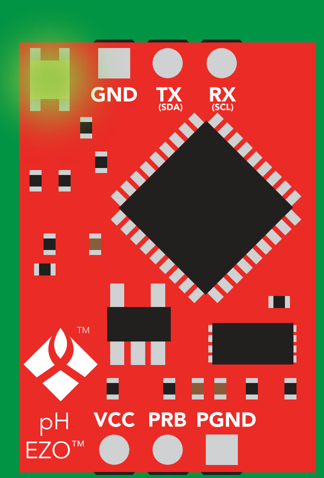
Plock,1 <cr> enable Plock

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

Plock,? <cr> Plock on/off?

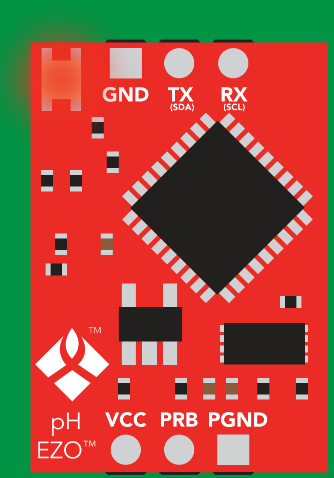
Example	Response
Plock,1 <cr>	*OK <cr>
Plock,0 <cr>	*OK <cr>
Plock,? <cr>	?Plock,1 <cr> or ?Plock,0 <cr>

Plock,1



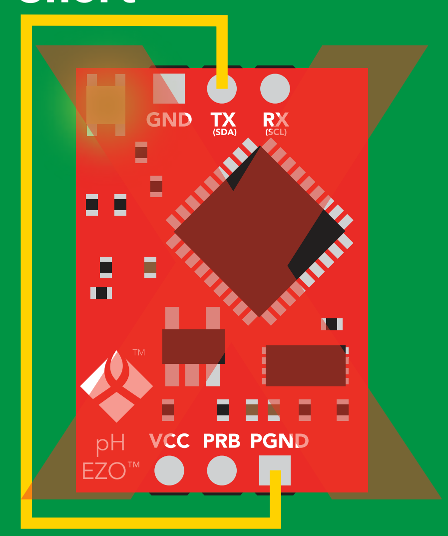
*OK <cr>

I2C,100



cannot change to I²C
*ER <cr>

Short



cannot change to I²C

AtlasScientific Environmental Robotics

Factory reset

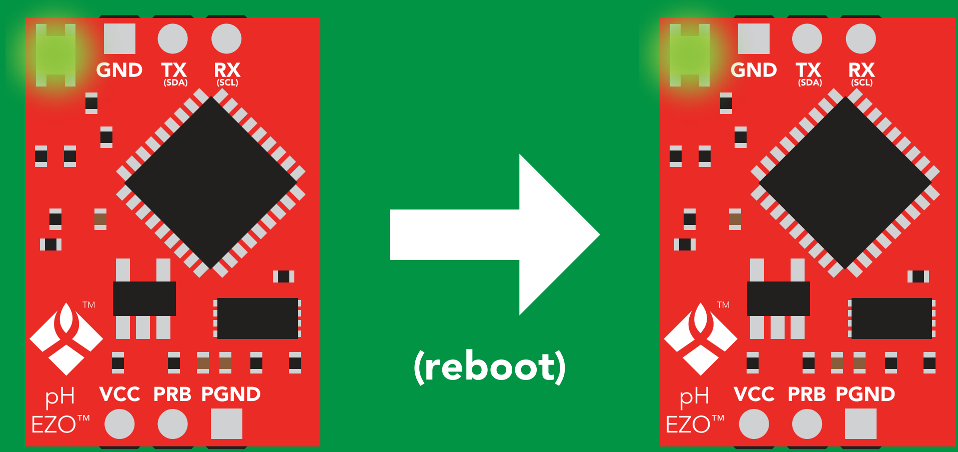
Command syntax

Clears calibration
LED on
"*OK" enabled

Factory	<cr>	enable factory reset

Example	Response
Factory <cr>	*OK <cr>

Factory <cr>



*OK <cr>

*RS <cr>
*RE <cr>

Baud rate will not change

Change to I²C mode

Command syntax

Default I²C address 99 (0x63)

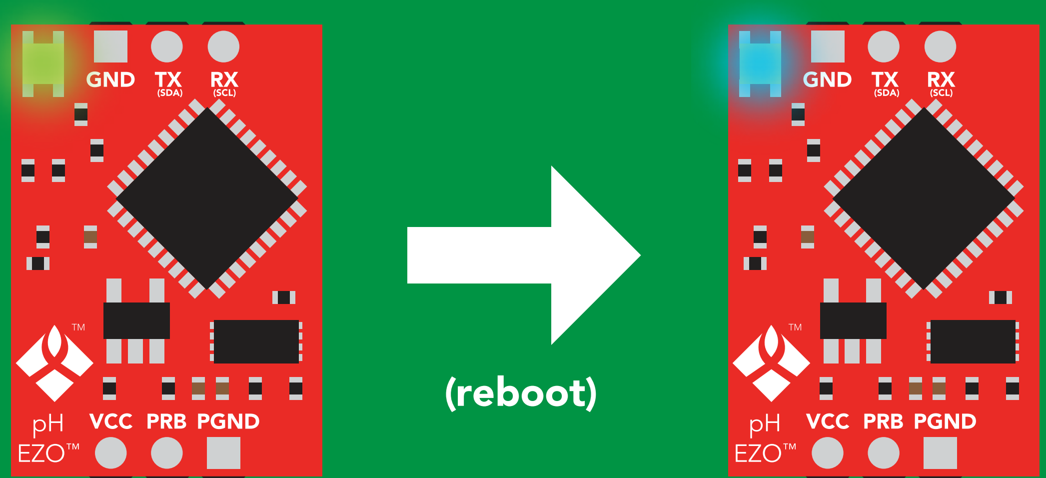
I2C,n <cr> sets I²C address and reboots into I²C mode

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



Green

*OK <cr>

Blue

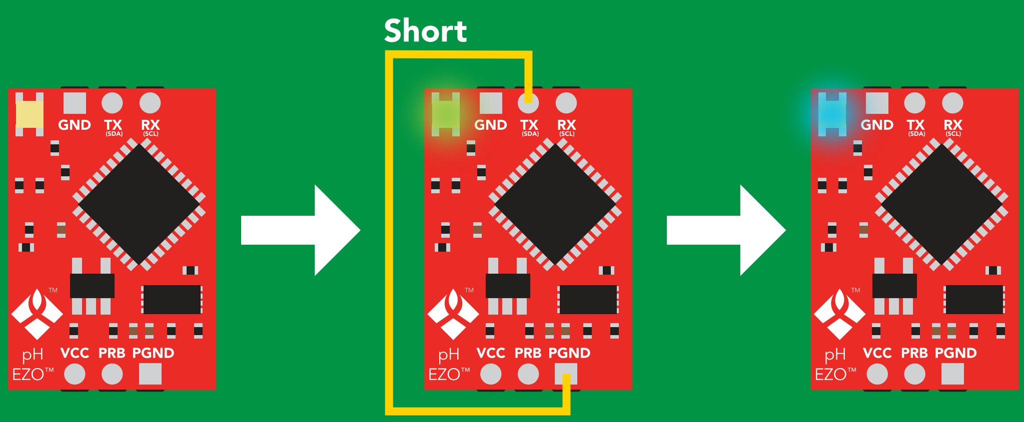
now in I²C mode

Manual switching to I²C

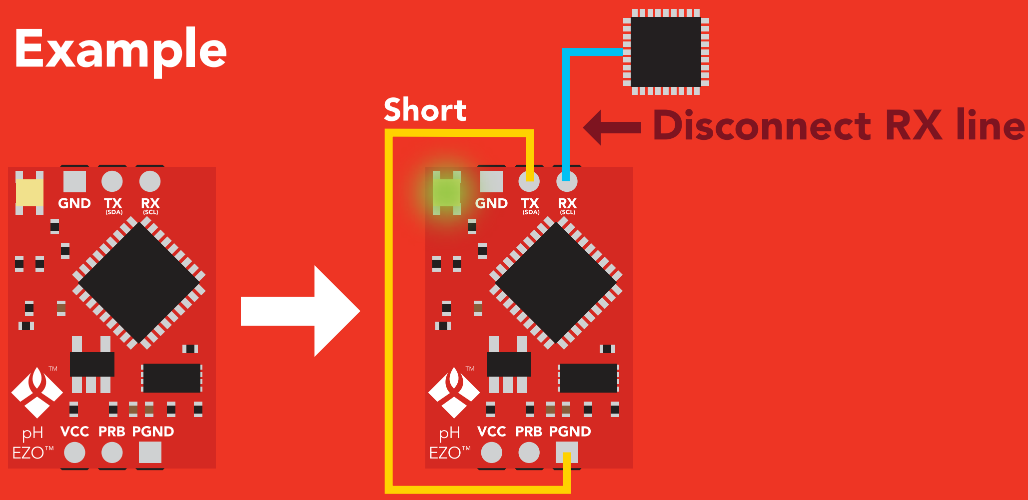
- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to PGND
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from Green to Blue
- Disconnect ground (power off)
- Reconnect all data and power

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

Example



Wrong Example



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™ device into I²C mode click here

Settings that are retained if power is cut

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

Settings that are *NOT* retained if power is cut

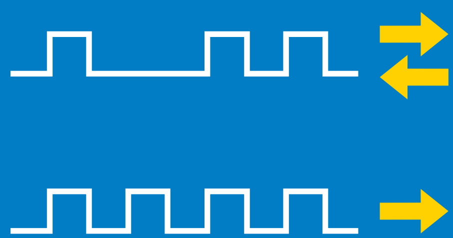
Find
Sleep mode
Temperature compensation

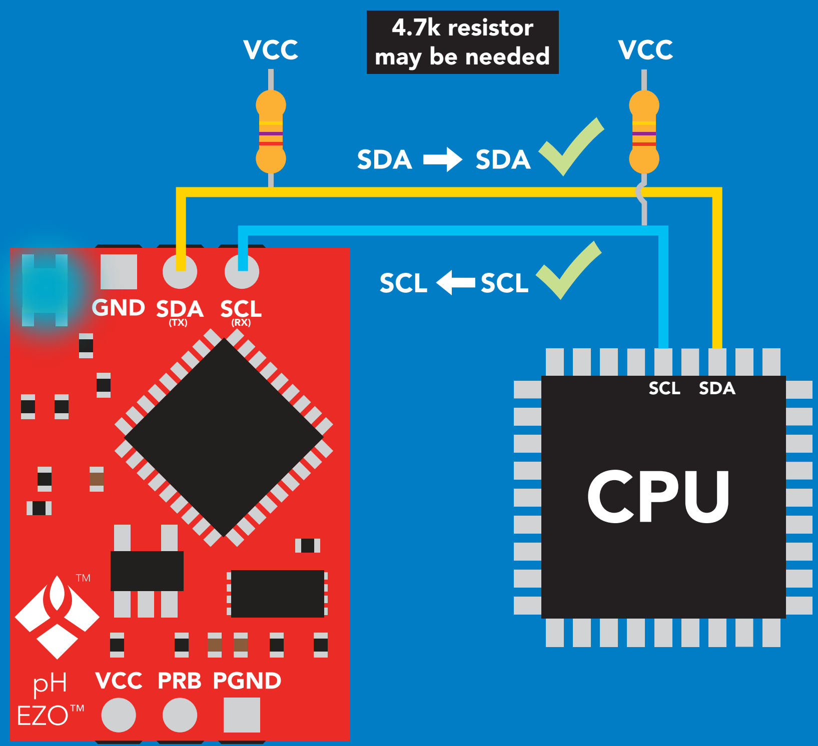
I^{2}C mode

I^{2}C address	(0x01 – 0x7F) 99 (0x63) default
Vcc	3.3V – 5.5V
Clock speed	100 – 400 kHz

SDA

SCL





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

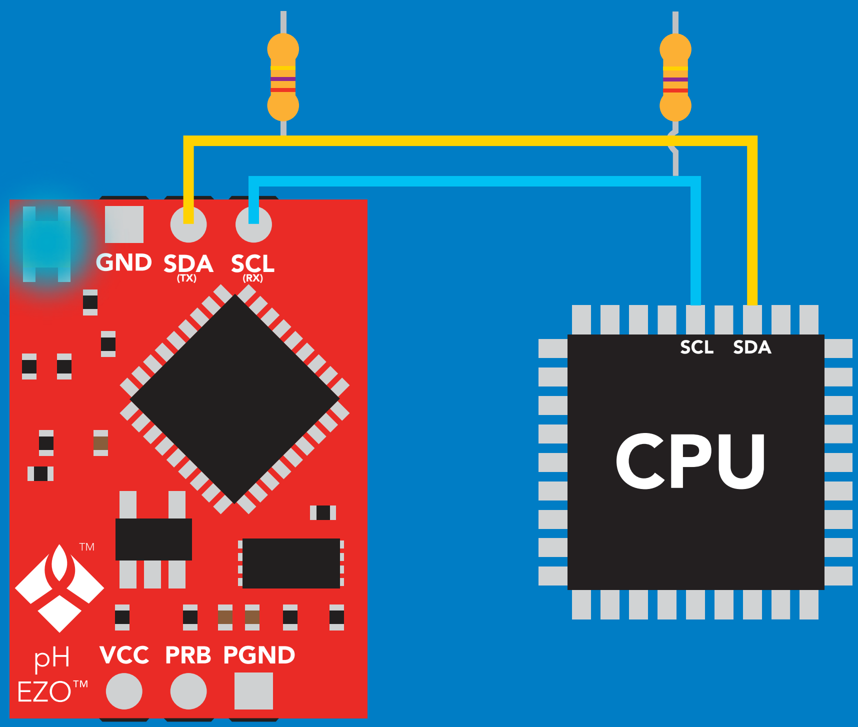
Sending commands to device

5 parts

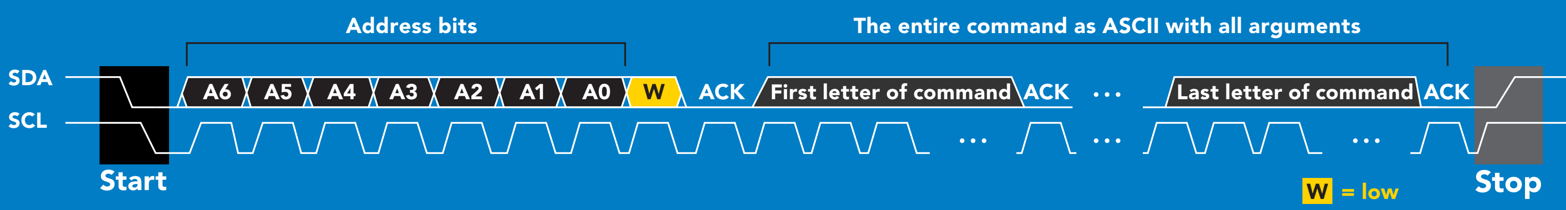
Start	I^2C address	Write	Command (not case sensitive)	Stop
	99 (0x63)		ASCII command string	

Example

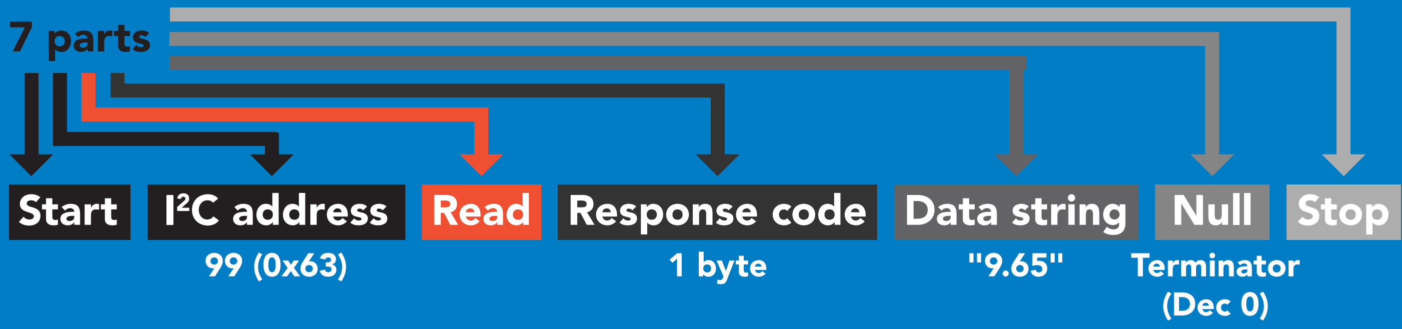
Start	99 (0x63)	Write	Sleep	Stop
	I^2C address		Command	

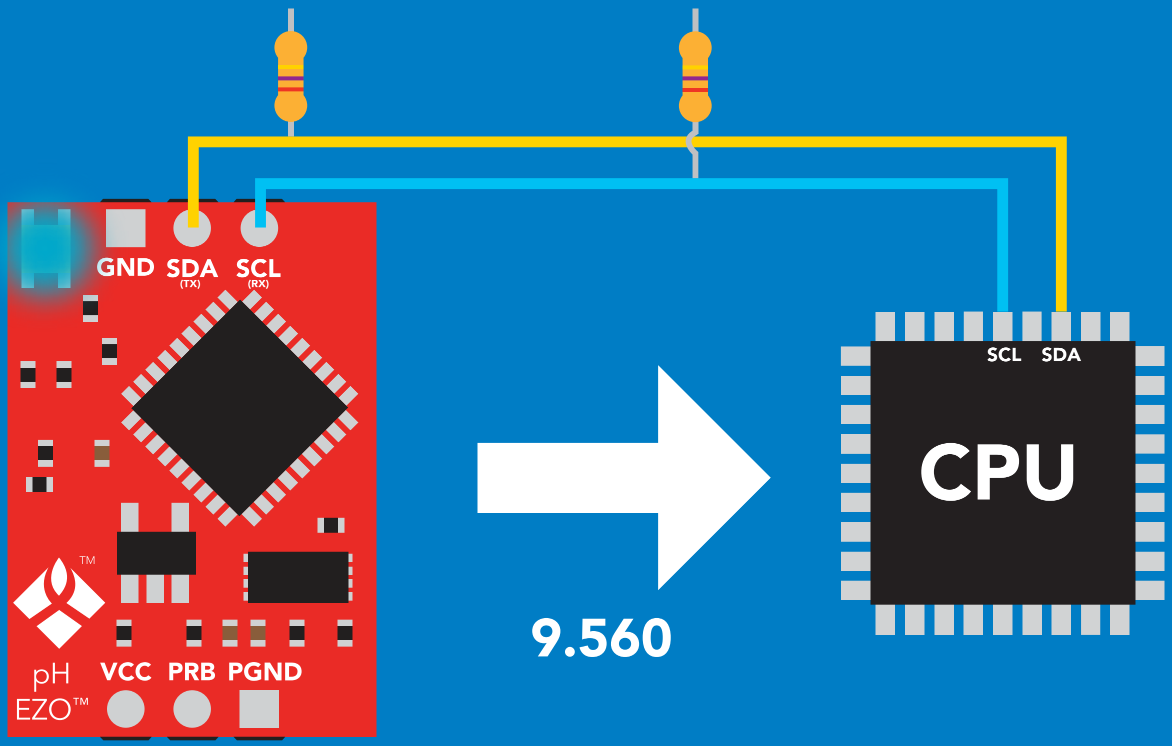


Advanced

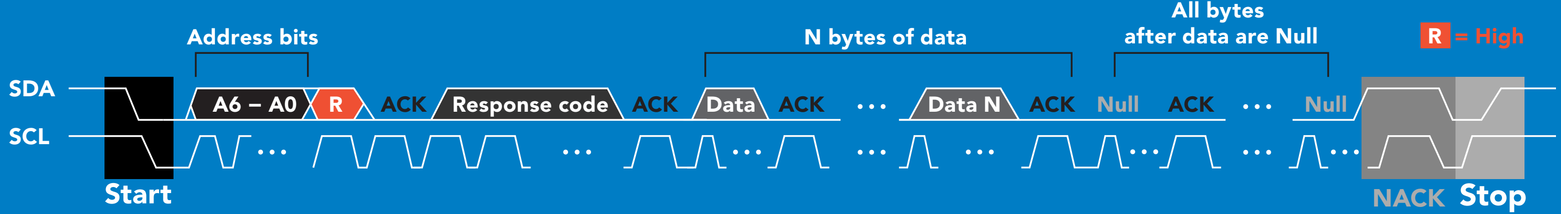


Requesting data from device





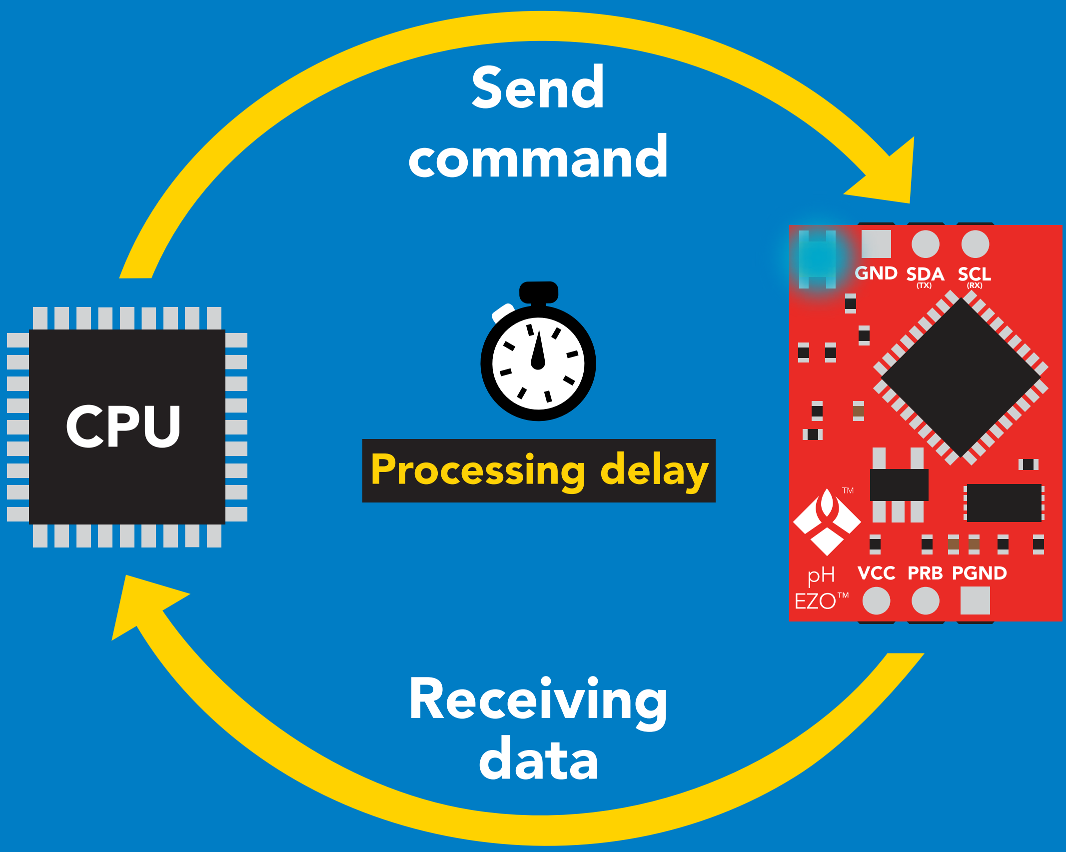
Advanced



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;
```

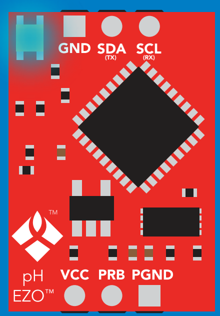
If there is no processing delay or the processing delay is too short, the response code will always be 254.

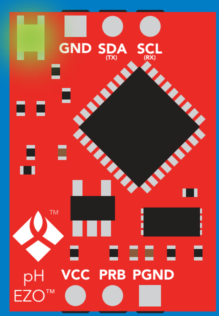
Response codes

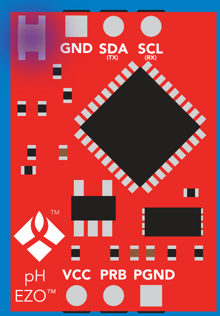
Single byte, not string

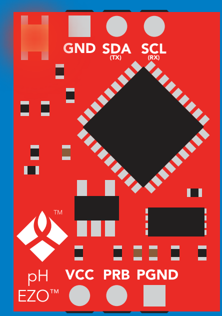
255	no data to send
254	still processing, not ready
2	syntax error
1	successful request

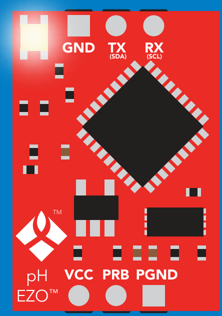
LED color definition











Blue — I²C standby

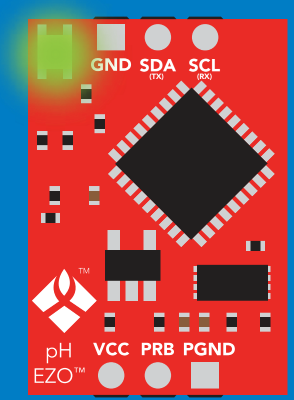
Green — Taking reading

Purple — Changing I²C address

Red — Command not understood

White — Find

	LED ON
5V	**+2.2 mA**
3.3V	**+0.6 mA**



Solid Green LED

in UART mode
Not I²C ready

AtlasScientific™
Environmental Robotics

I^2C mode
command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function	
Baud	switch back to UART mode	pg. 61
Cal	performs calibration	pg. 48
Export	export calibration	pg. 49
Factory	enable factory reset	pg. 60
Find	finds device with blinking white LED	pg. 46
i	device information	pg. 55
I2C	change I^2C address	pg. 59
Import	import calibration	pg. 50
L	enable/disable LED	pg. 45
Name	set/show name of device	pg. 54
pHext	enable/disable extended pH scale	pg. 52
Plock	enable/disable protocol lock	pg. 58
R	returns a single reading	pg. 47
Sleep	enter sleep mode/low power	pg. 57
Slope	returns the slope of the pH probe	pg. 51
Status	retrieve status information	pg. 56
T	temperature compensation	pg. 53

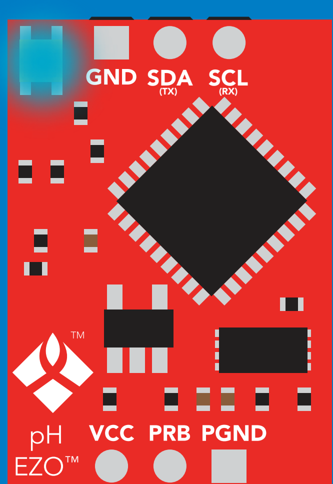
LED control

Command syntax

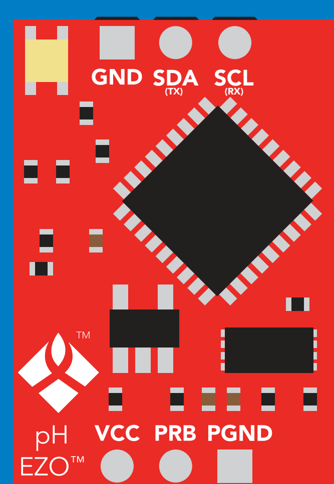
300ms processing delay

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

Example	Response
L,1	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 1 (Dec) ?L,0 (ASCII) 0 (Null)



L,1



L,0

AtlasScientific™ Environmental Robotics

Find

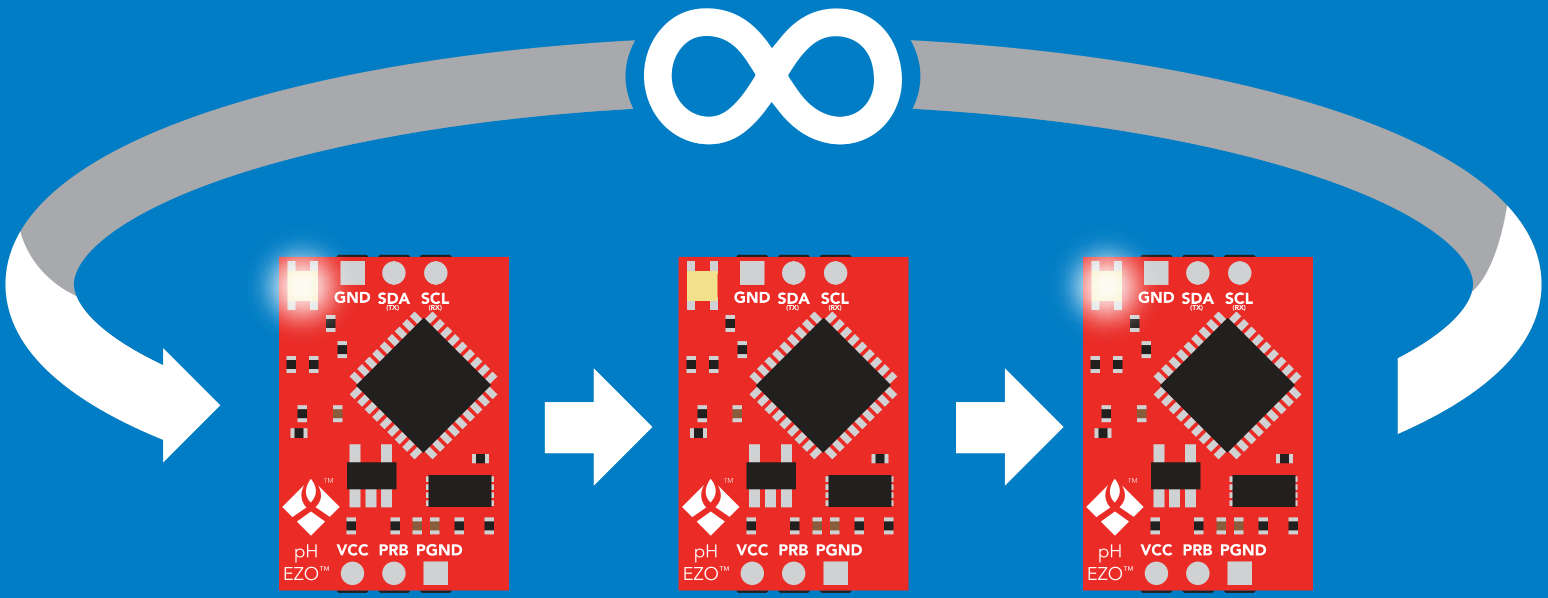
300ms processing delay

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	



AtlasScientific™ Environmental Robotics

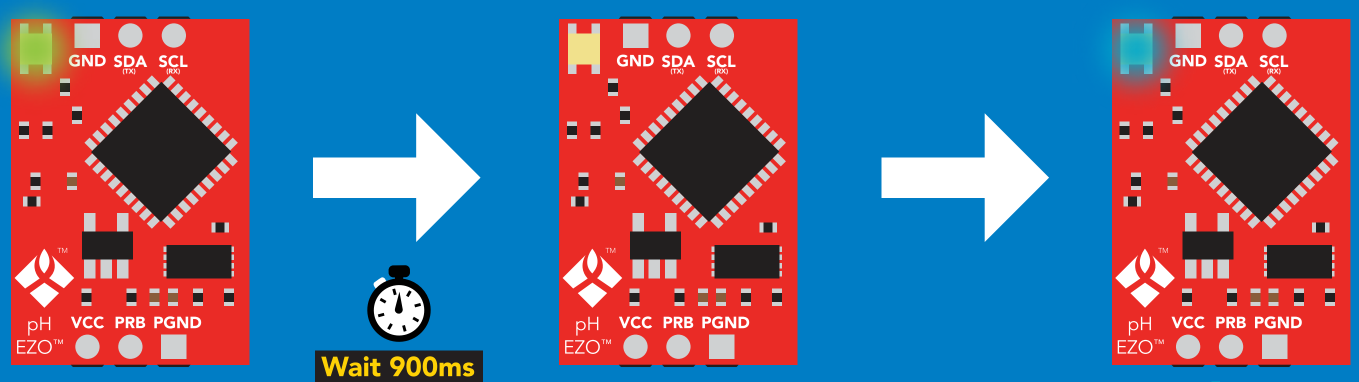
Taking reading

Command syntax

900ms processing delay

R return 1 reading

Example	Response
R	Wait 900ms · 1 Dec · 9.560 ASCII · 0 Null



Green
Taking reading

Transmitting

Blue
Standby

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); 1 (Dec) ?Cal,2 (ASCII, two point) 0 (Null) or 1 (Dec) ?Cal,3 (ASCII, three point) 0 (Null)

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,?	Wait 300ms — 1 (Dec) 10,120 (ASCII) 0 (Null)
Export	Wait 300ms — 1 (Dec) 59 6F 75 20 61 72 (ASCII) 0 (Null) (1 of 10)
Export	Wait 300ms — 1 (Dec) 65 20 61 20 63 6F (ASCII) 0 (Null) (2 of 10)
(7 more)	⋮
Export	Wait 300ms — 1 (Dec) 6F 6C 20 67 75 79 (ASCII) 0 (Null) (10 of 10)
Export	Wait 300ms — 1 (Dec) *DONE (ASCII) 0 (Null)

Response breakdown

10, 120

10: # of strings to export

120: # of bytes to export

Export strings can be up to 12 characters long

AtlasScientific™ Environmental Robotics

Import calibration

300ms processing delay

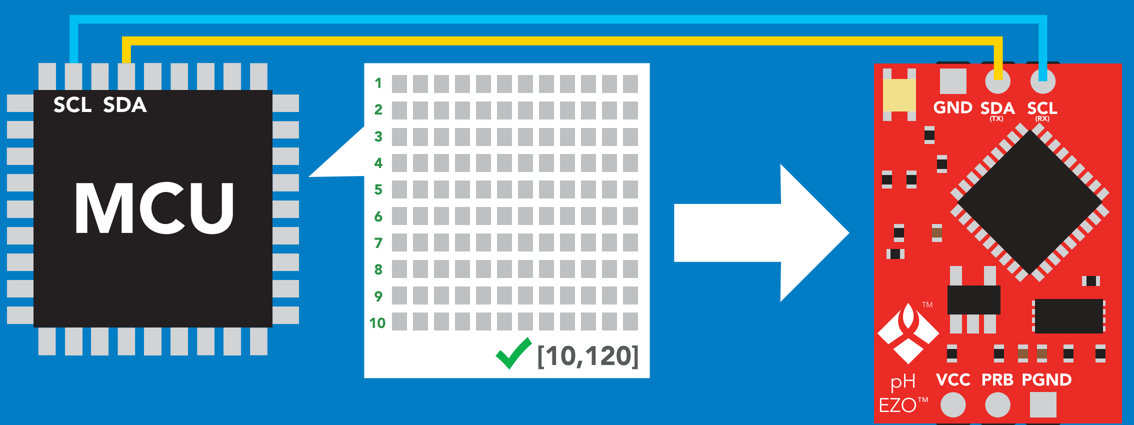
Command syntax

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

Import,n	**import calibration string to new device**

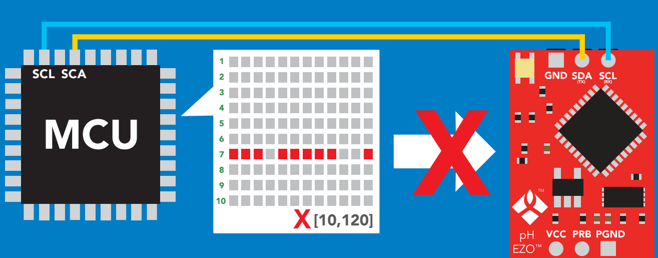
Example		Response
Import, 59 6F 75 20 61 72	(1 of 10)	Wait 300ms 1 Dec 0 Null
Import, 65 20 61 20 63 6F	(2 of 10)	Wait 300ms 1 Dec 0 Null
⋮		⋮
Import, 6F 6C 20 67 75 79	(10 of 10)	Wait 300ms 1 Dec 0 Null

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.**

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,?

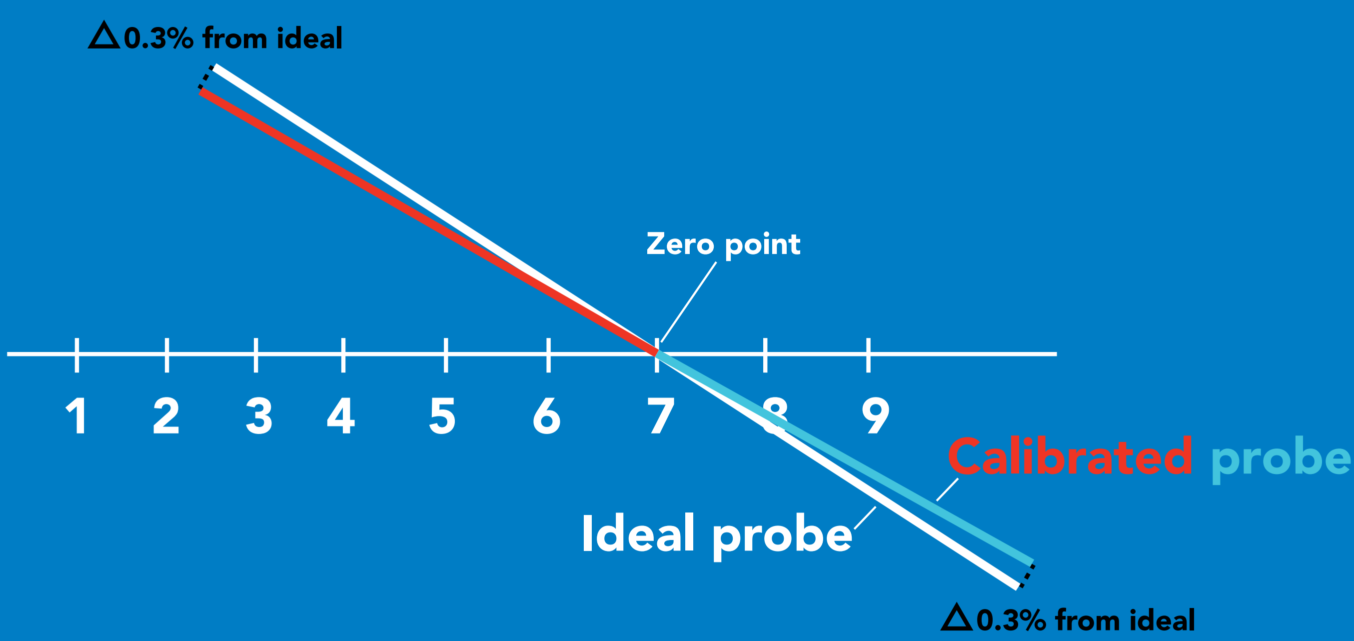
Response

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.



AtlasScientific 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

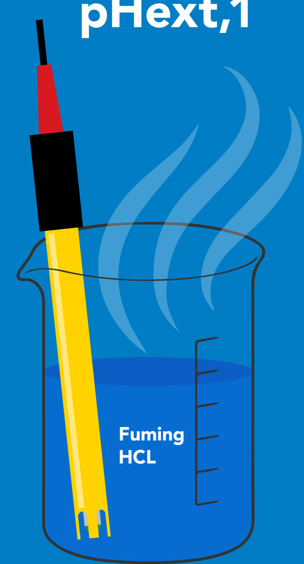
pHext,0	extended pH scale off (0 – 14)	**default**
pHext,1	extended pH scale on (-1.6 – 15.6)	
pHext,?	extended pH scale on/off?	

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

AtlasScientific Environmental Robotics

Temperature compensation

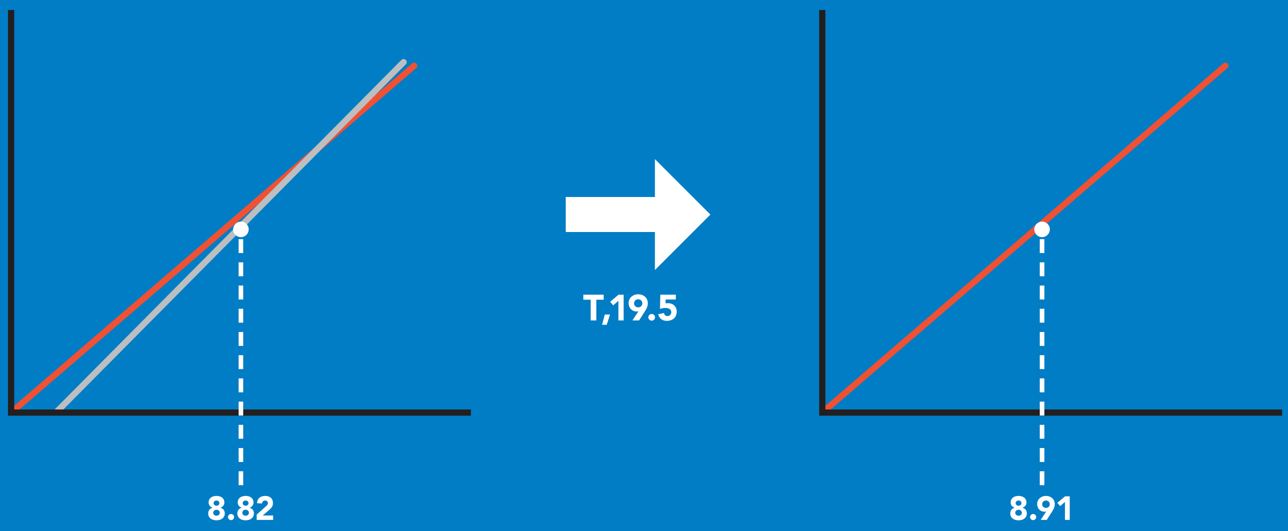
Command syntax

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
T,?	compensated temperature value?
RT,n	set temperature compensation and take a reading

300ms processing delay

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)



AtlasScientific
Environmental Robotics

Naming device

300ms processing delay

Command syntax

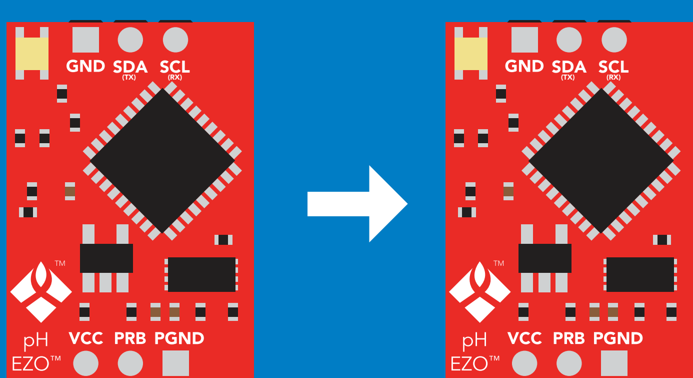
Do not use spaces in the name

Name,n	set name	n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ (1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16)
Name,	clears name	Up to 16 ASCII characters
Name,?	show name	

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

Name,?



1 0

1 ?Name,zzt 0

AtlasScientific Environmental Robotics

Device information

Command syntax

300ms processing delay

i	device information

Example	Response
i	Wait 300ms · 1 (Dec) · ?i,pH,1.98 (ASCII) · 0 (Null)

Response breakdown

?i, pH, 1.98

pH, → Device

1.98 → Firmware

Reading device status

Command syntax

300ms processing delay

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

Example	Response
Status	Wait 300ms · 1 (Dec) · ?Status,P,5.038 (ASCII) · 0 (Null)

Response breakdown

?Status,	P,	5.038
	Reason for restart	Voltage at Vcc

Restart codes

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

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**

Protocol lock

Command syntax

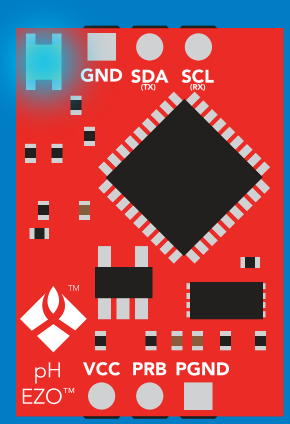
300ms processing delay

Plock,1	enable Plock	
Plock,0	disable Plock	default
Plock,?	Plock on/off?	

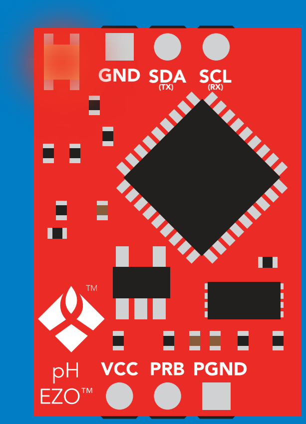
Locks device to I²C mode.

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)

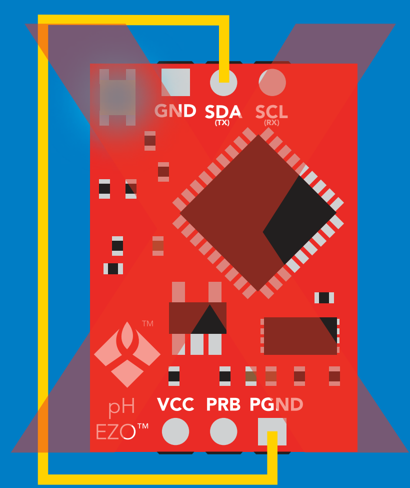
Plock,1



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 (no response given)

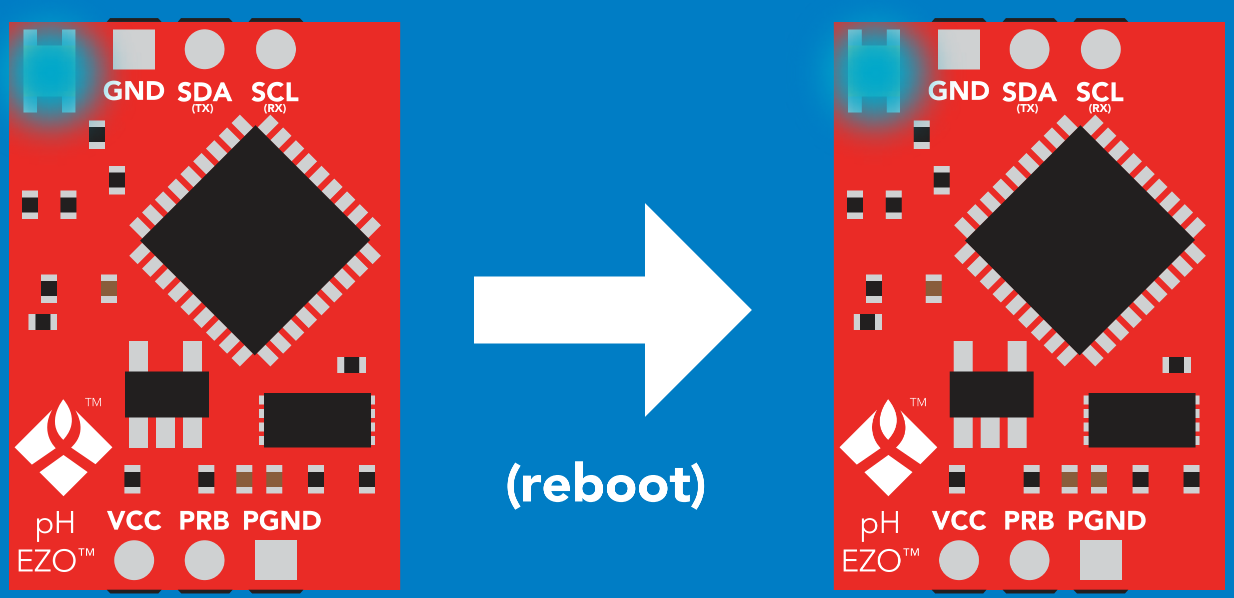
Warning!

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



AtlasScientific
Environmental Robotics

Factory reset

Command syntax

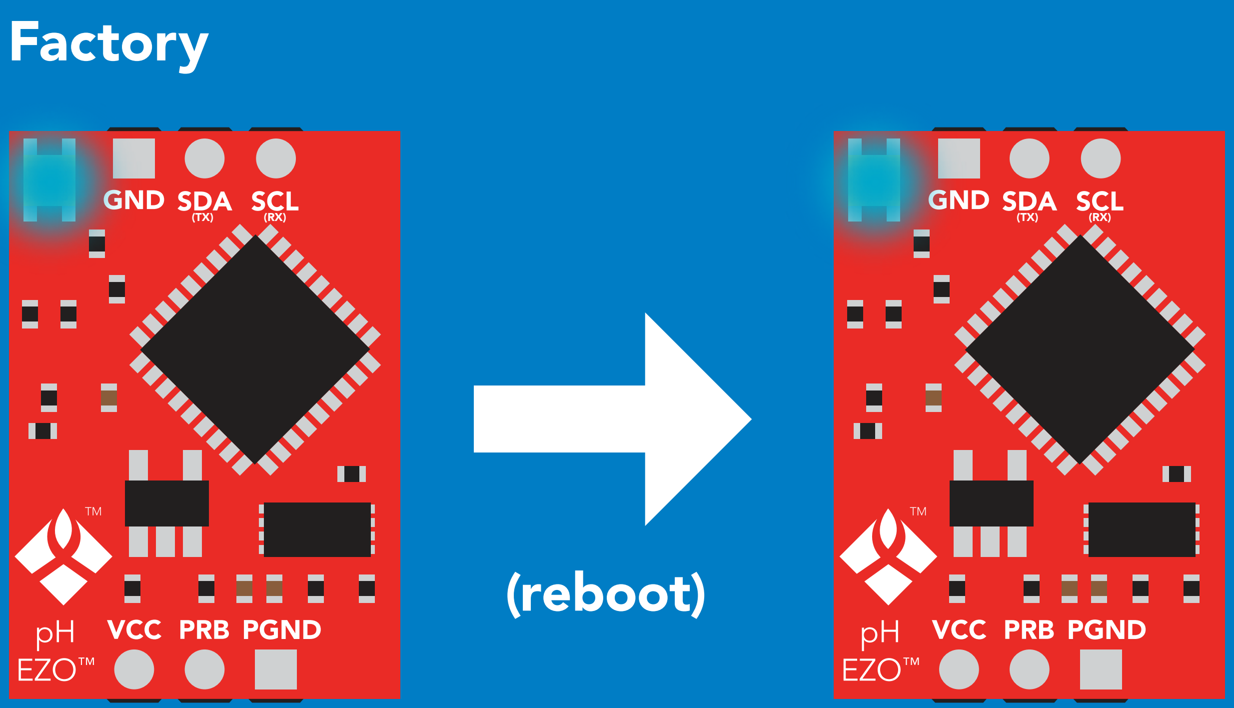
Factory reset will not take the device out of I^2C mode.

Factory	enable factory reset

I^2C address will not change

Example	Response
Factory	device reboot (no response given)

Clears calibration
LED on
Response codes enabled



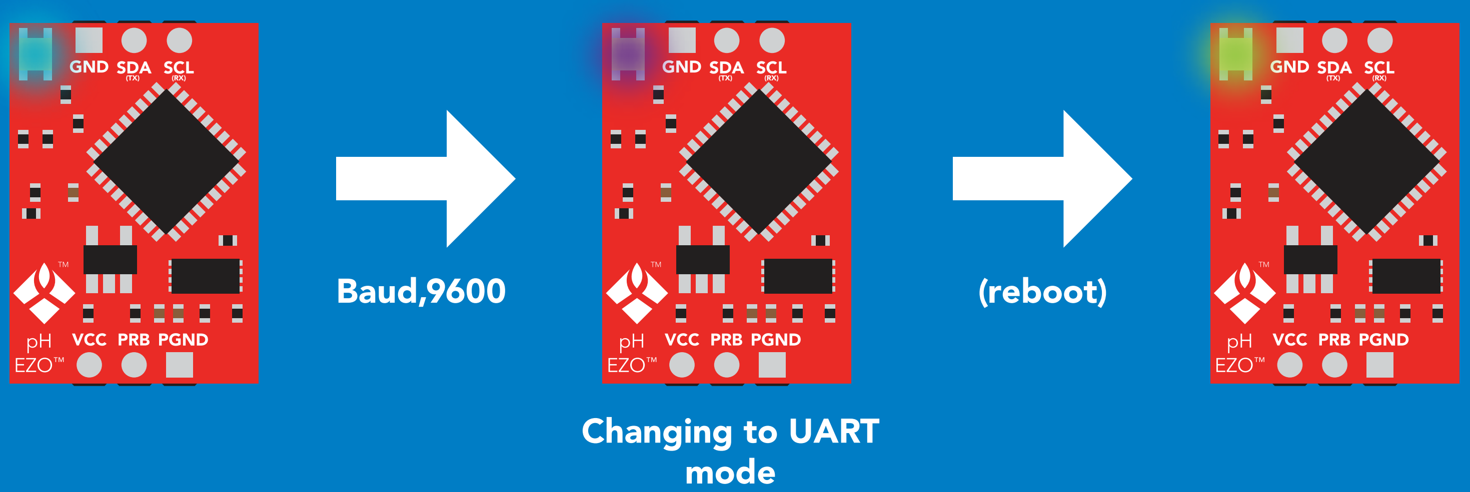
Change to UART mode

Command syntax

Baud,n switch from I²C to UART

Example	Response
Baud,9600	reboot in UART mode (no response given)

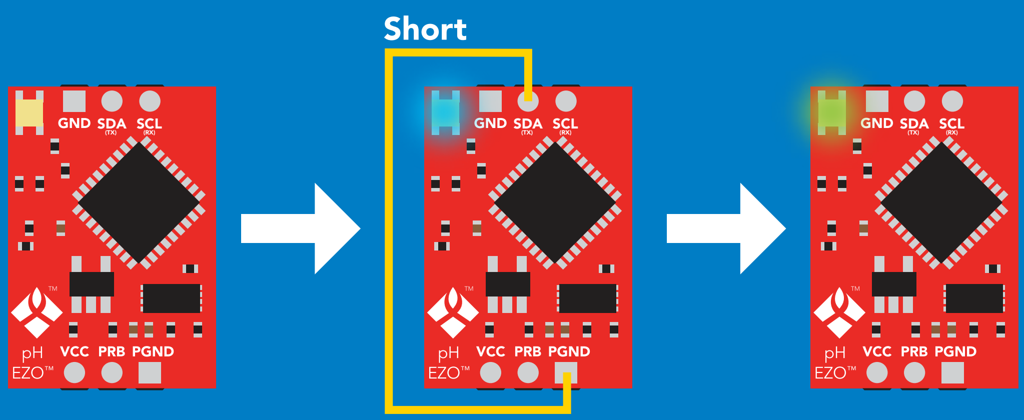
n =
- 300
- 1200
- 2400
- 9600
- 19200
- 38400
- 57600
- 115200



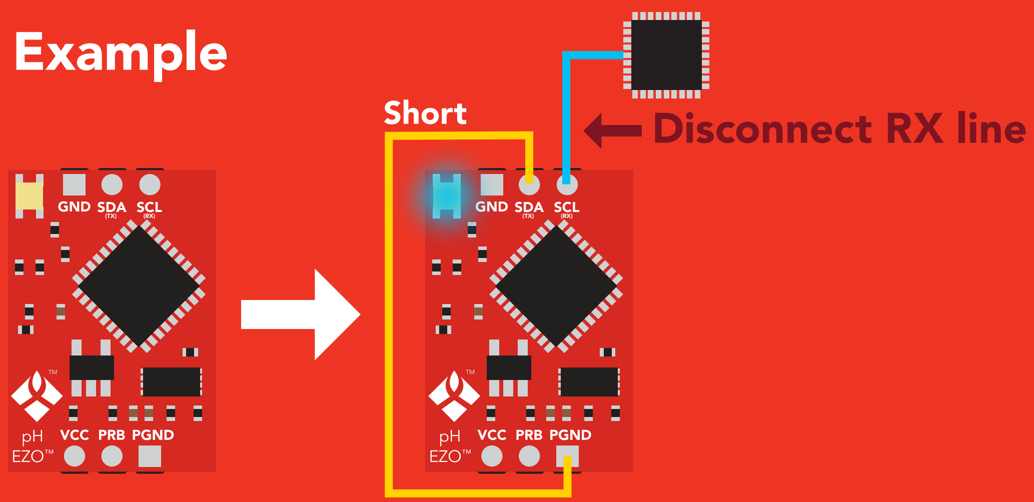
Manual switching to UART

- 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

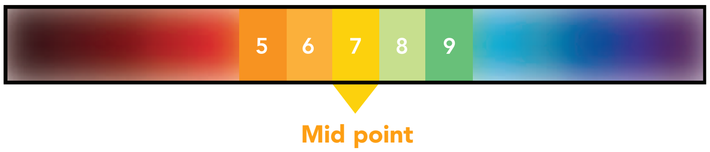


Calibration theory

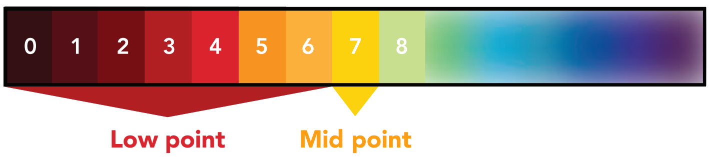
The accuracy of your readings is directly related to the quality of your calibration.
(Calibration is not difficult, and a little bit of care goes a long way).

Single, Two point, or Three point calibration accuracy

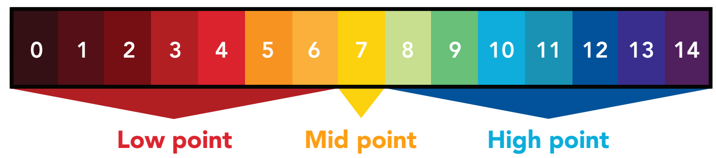
Single point calibration



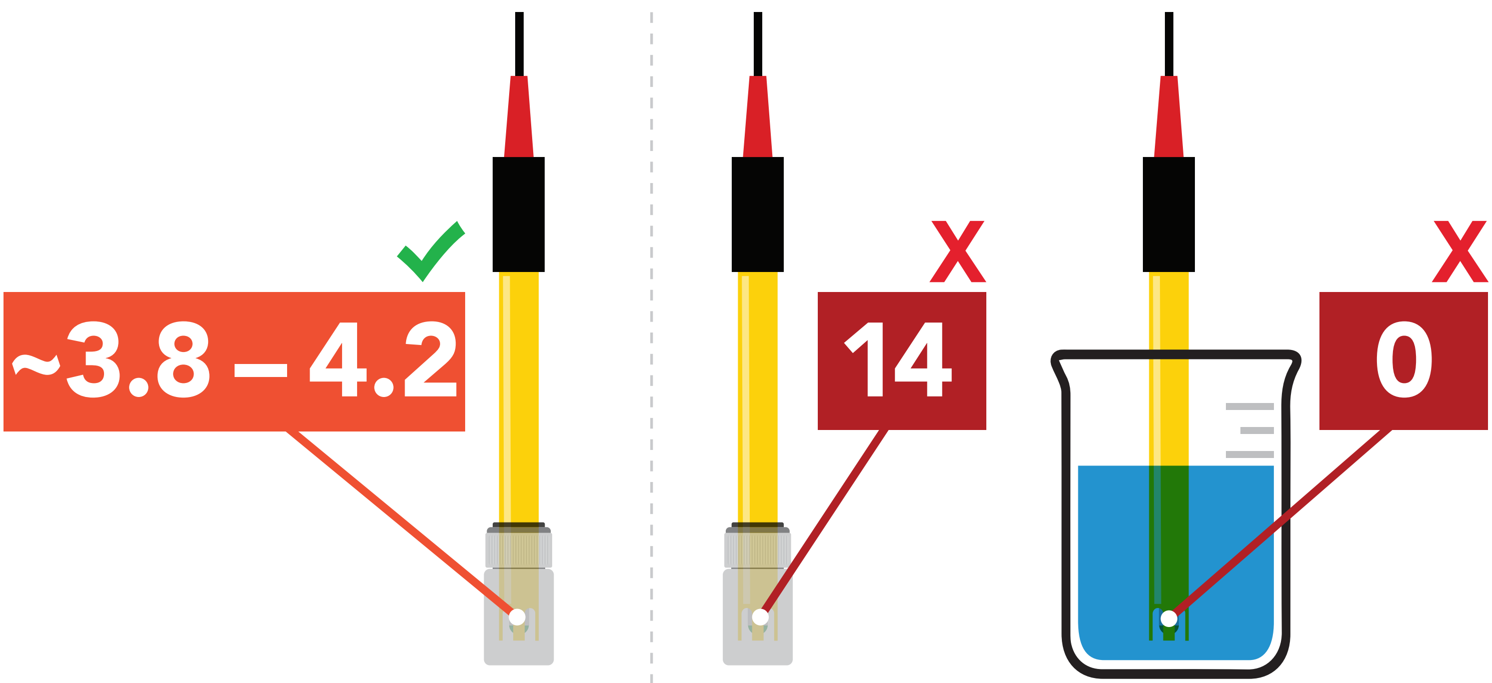
Two point calibration



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 ***~3.8 – 4.2***

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.

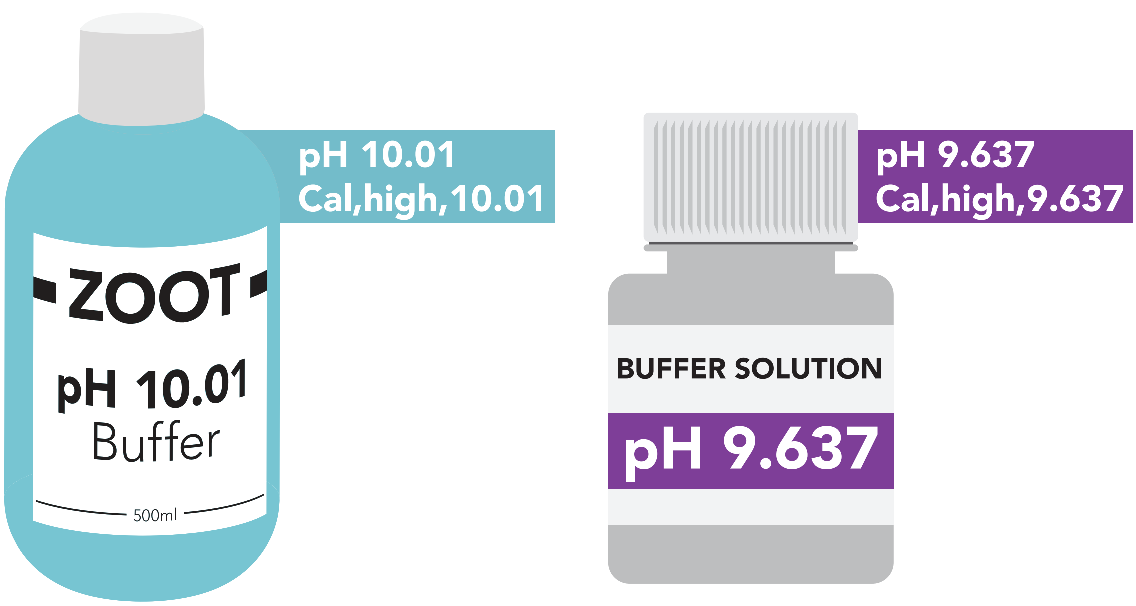
1 Mid point

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

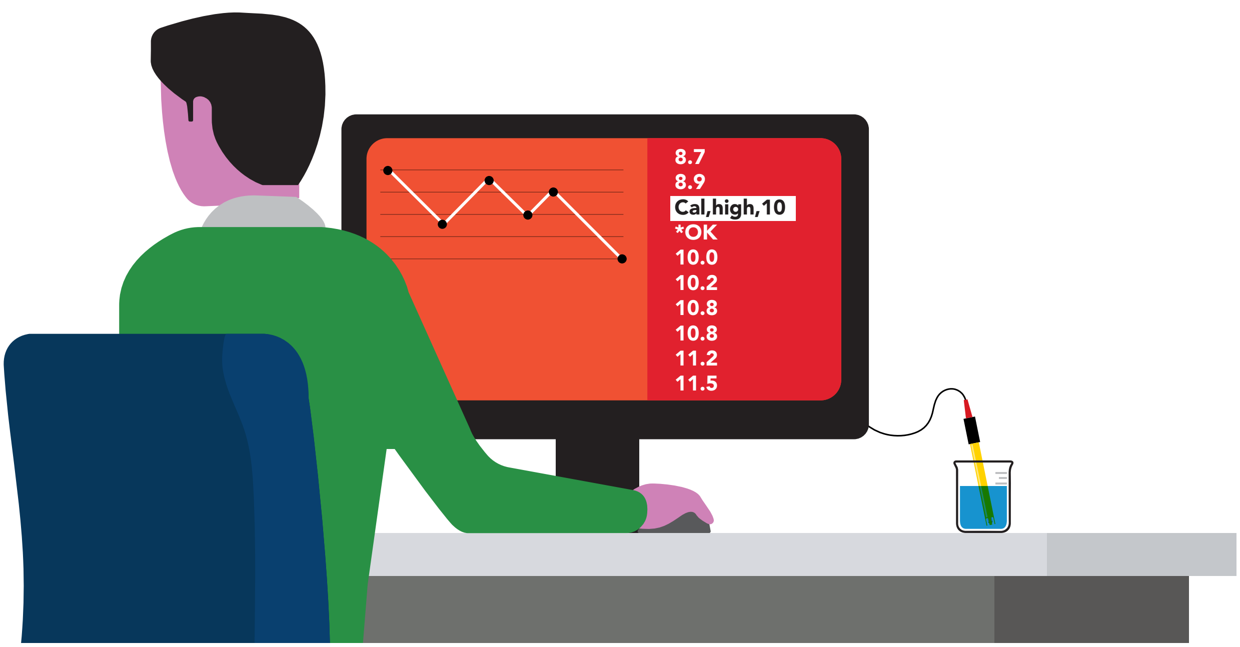
While you can use calibration solutions that have complex values, we recommend avoiding unnecessary complexity. ***Unusually specific calibration values should be treated with suspicion.***

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.

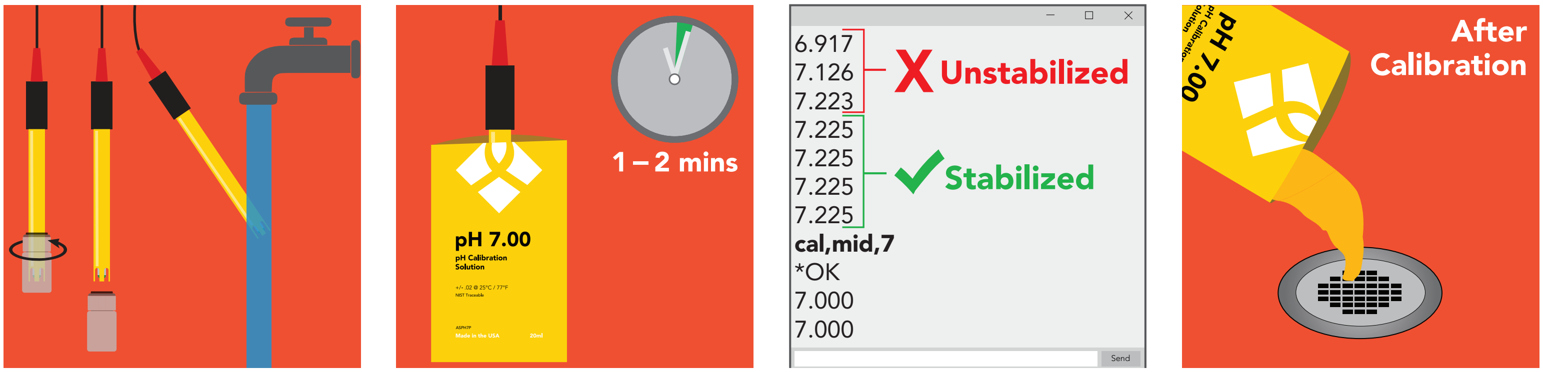
Letting the probes pre-calibrtion readings stabilize over an extended period will cause your calibrated readings to take a long time to stabilize.

Avoid frequent recalibrations.

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

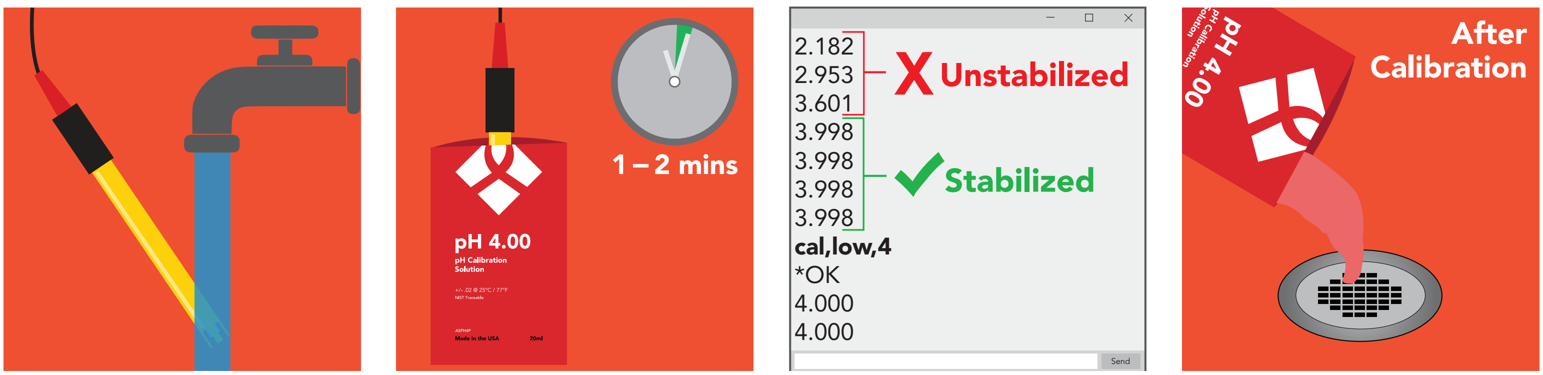
pH probes loose accuracy slowly. Frequent recalibrations to insure high accuracy will often have the opposite effect. It is far more llikly that you will misscalibrate the probe rather then improve its accuracy.

1. Mid point calibration



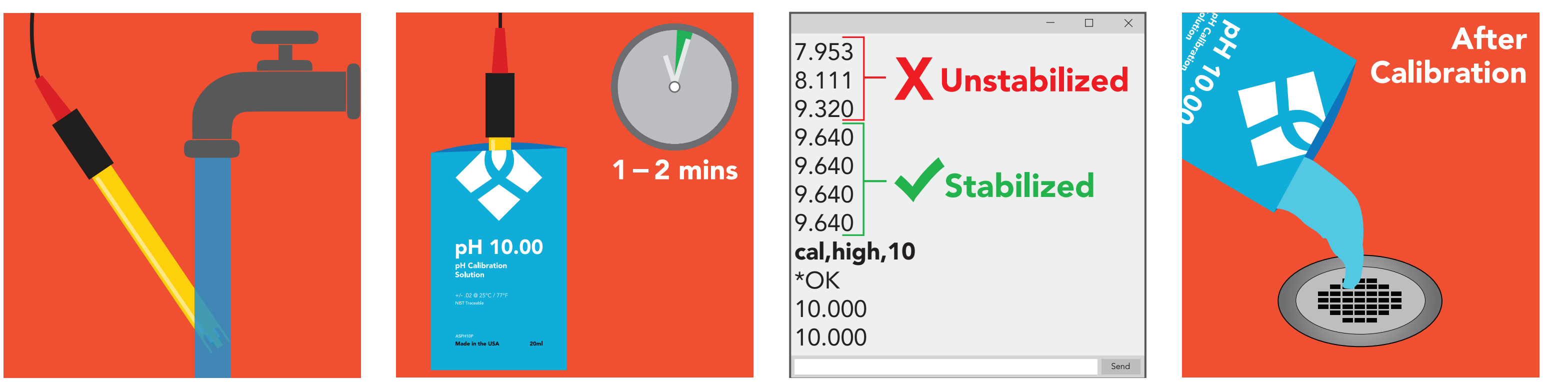
Mid point calibrated

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.

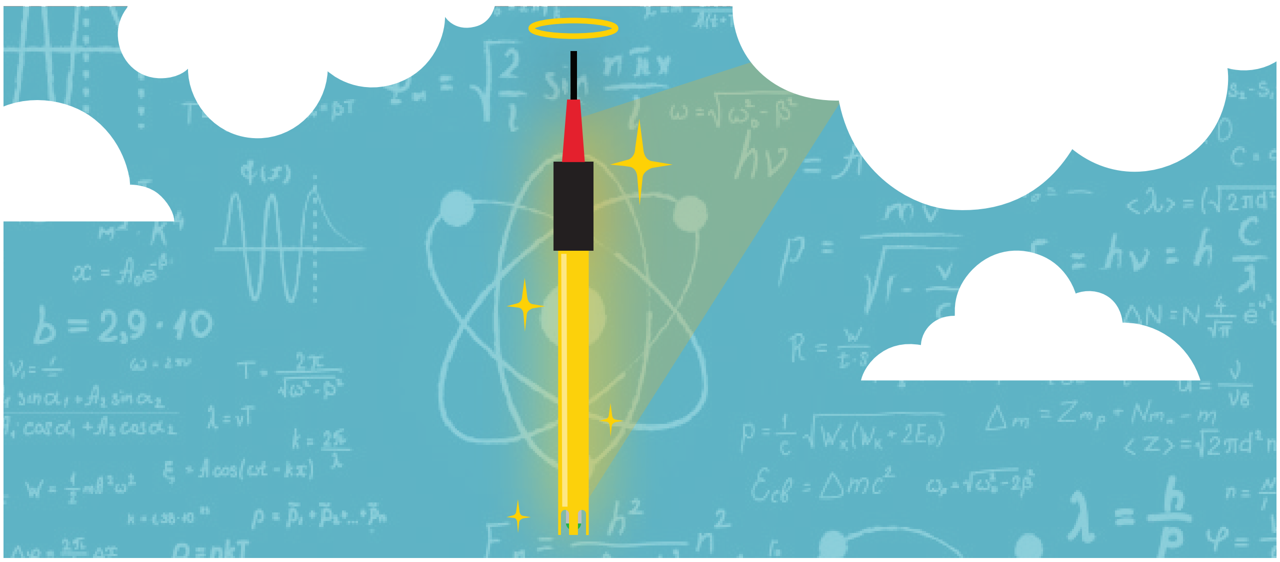
AtlasScientific
Environmental Robotics

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)

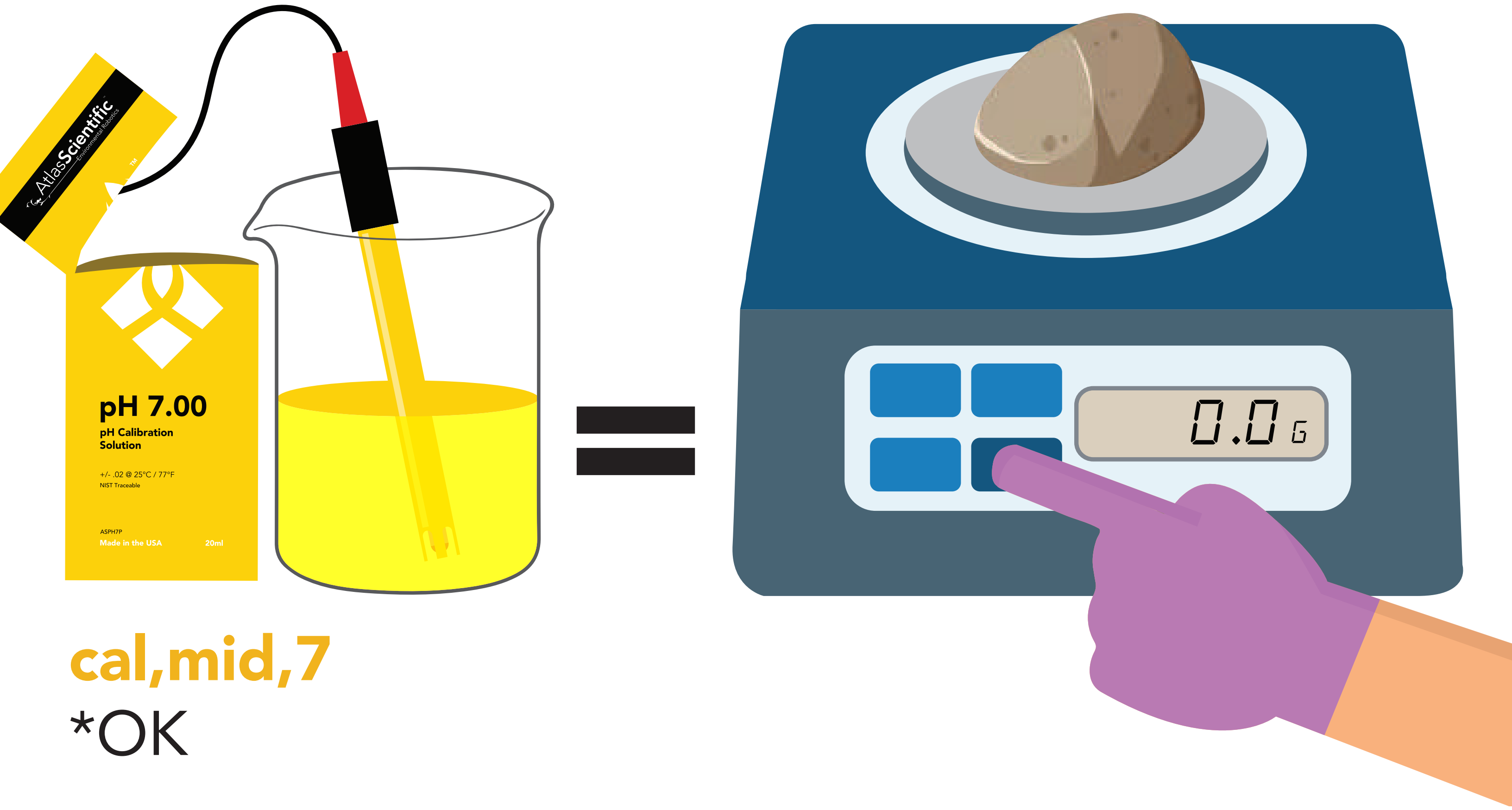
An uncalibrated pH probe will have a mathematically perfect slope. Because no pH probe is mathematically perfect, the slope can be used to determine if the pH was calibrated.

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%.

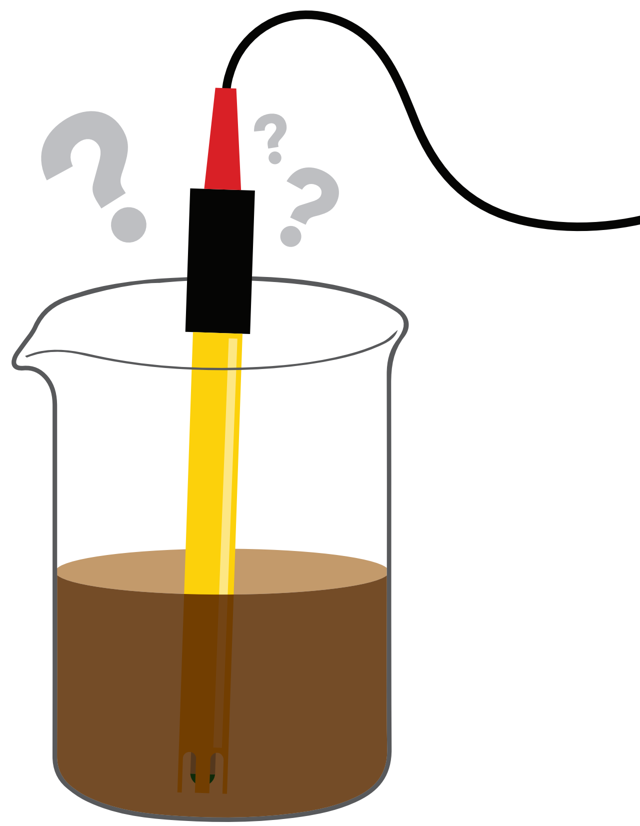
The slope after pH 4 calibration: 98.2, 100, -1.2

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.

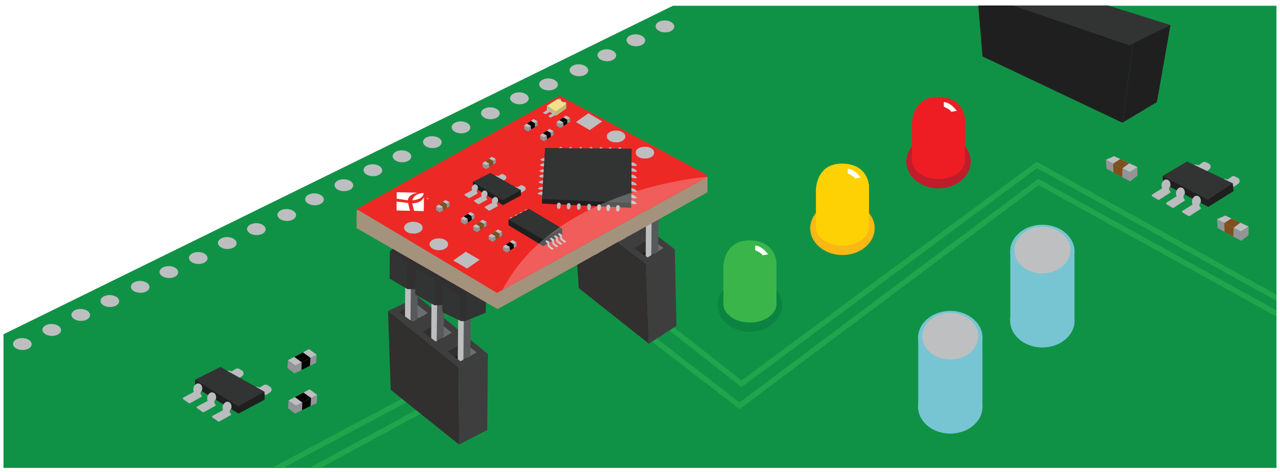
To gain a deeper understanding of how slope affects the stability and accuracy of a pH probe, intentionally miscalibrate the probe and see how it affects the slope.



Soldering

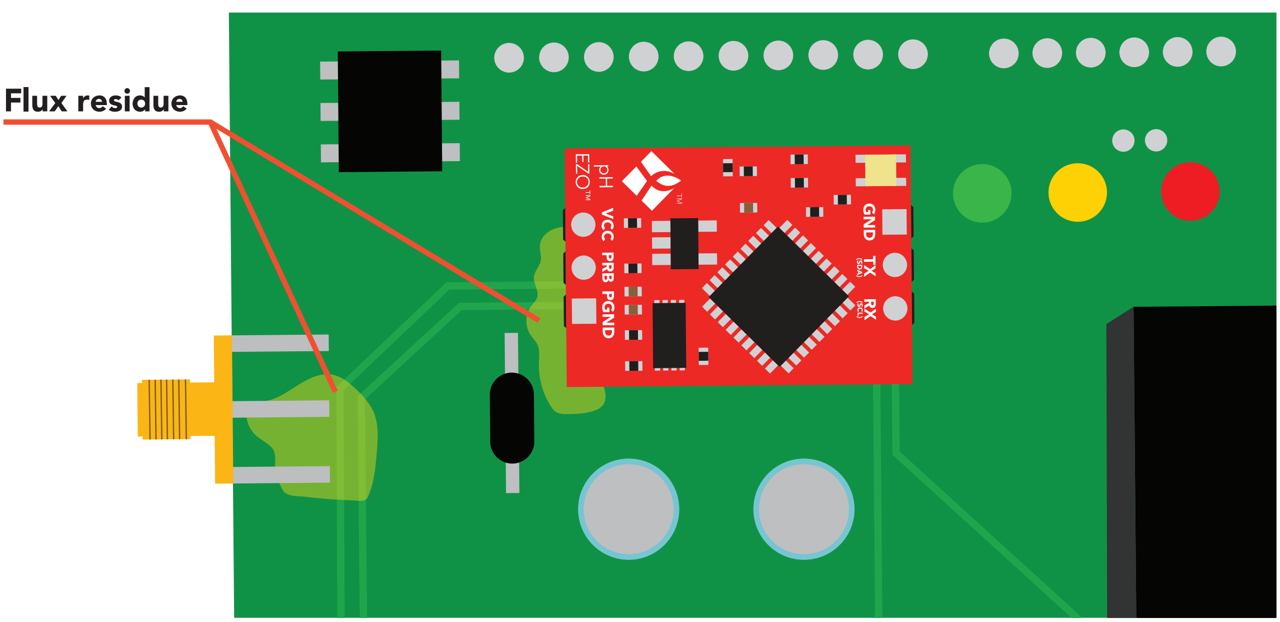
Do not directly solder an EZO circuit to your PCB. If something goes wrong during the soldering process it may become impossible to correct the problem. It is simply not worth the risk.

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.

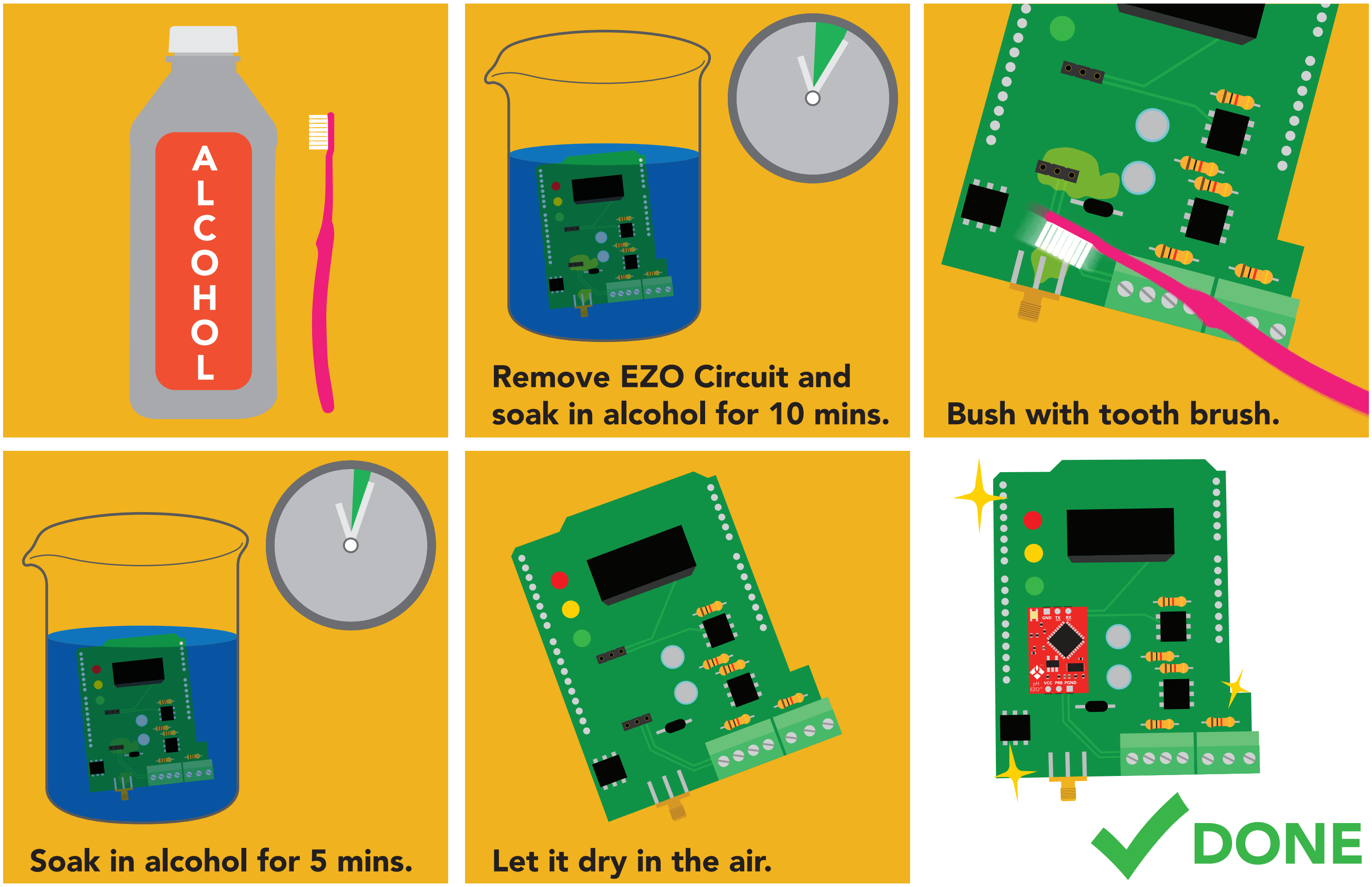
Flux residue will severely affect your readings. Any Flux residue that comes in contact with the PRB pins or your probes connector will cause a "flux short".



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

Soldering

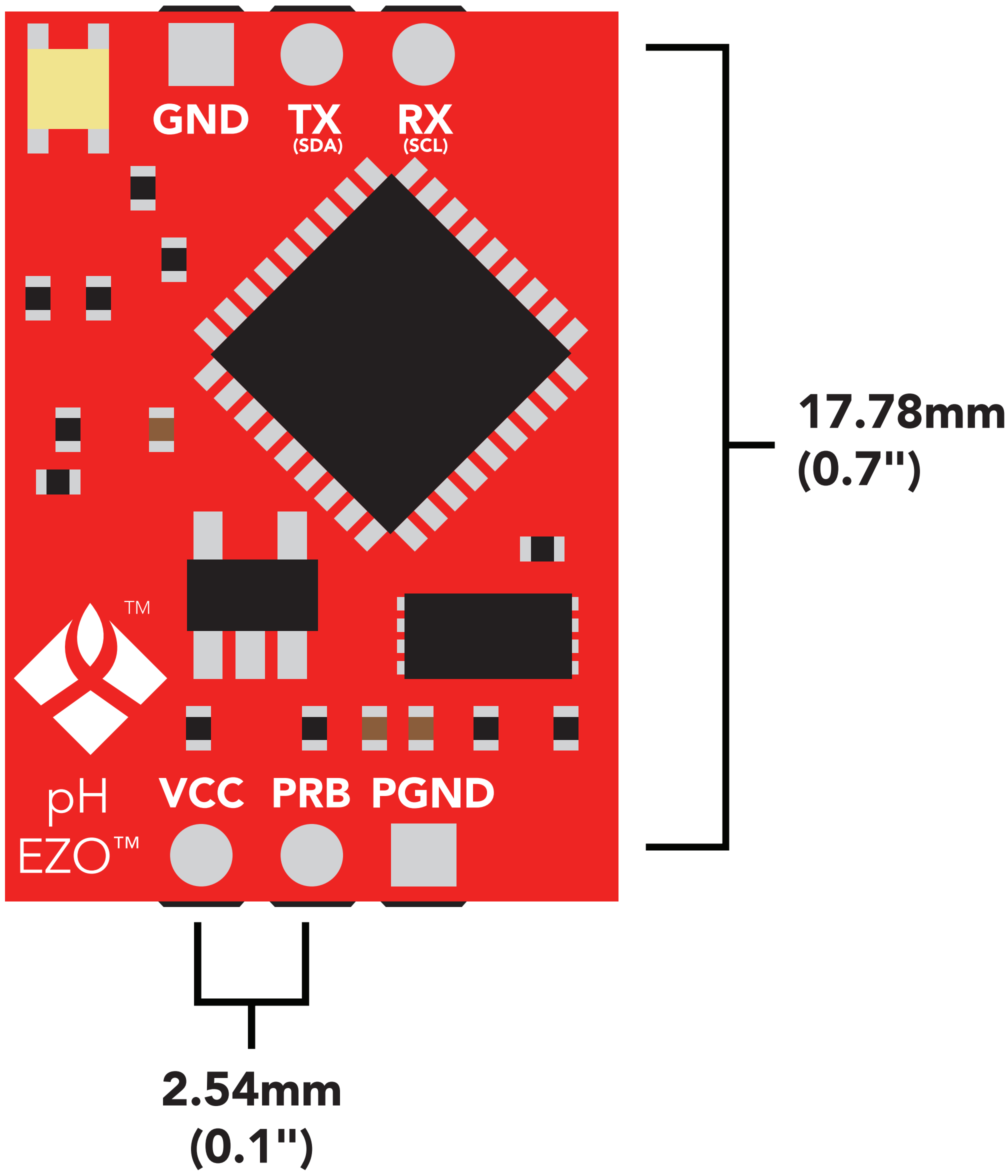
Removing flux residue can be done with commercially available products such as flux off or you can use alcohol and a tooth brush.



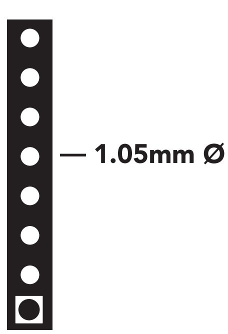
What does a flux short look like?

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

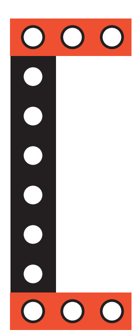
EZO™ circuit footprint



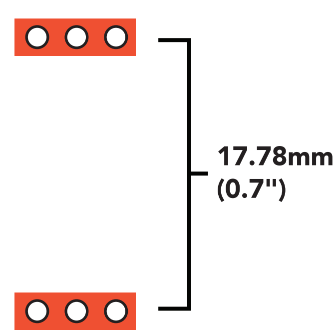
1 In your CAD software place a 8 position header.



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.

AtlasScientific™
Environmental Robotics

V 6.5

Revised 5/24

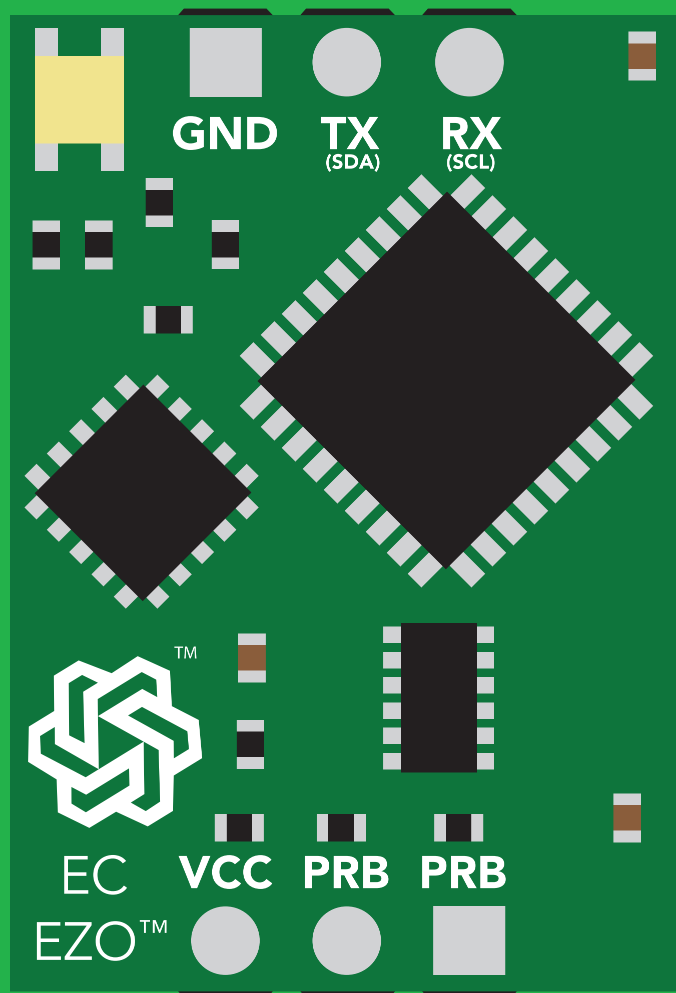
EZO-EC™

Embedded Conductivity Circuit

ISO 7888 Compliant

(determination of electrical conductivity)

Reads	**Conductivity = μS/cm** **Total dissolved solids = ppm** **Salinity = PSU (ppt) 0.00 – 42.00** **Specific gravity** ***(sea water only)* = 1.00 – 1.300**
Range	**0.07 – 500,000+ μS/cm**
Accuracy	**+/– 2%**
EC reading time	**600ms**
Supported probes	**K 0.01 – K 10.2 any brand**
Calibration	**2 or 3 point**
Temp compensation	**Yes**
Data protocol	**UART & I²C**
Default I²C address	**100 (0x64)**
Operating voltage	**3.3V – 5V**
Data format	**ASCII**

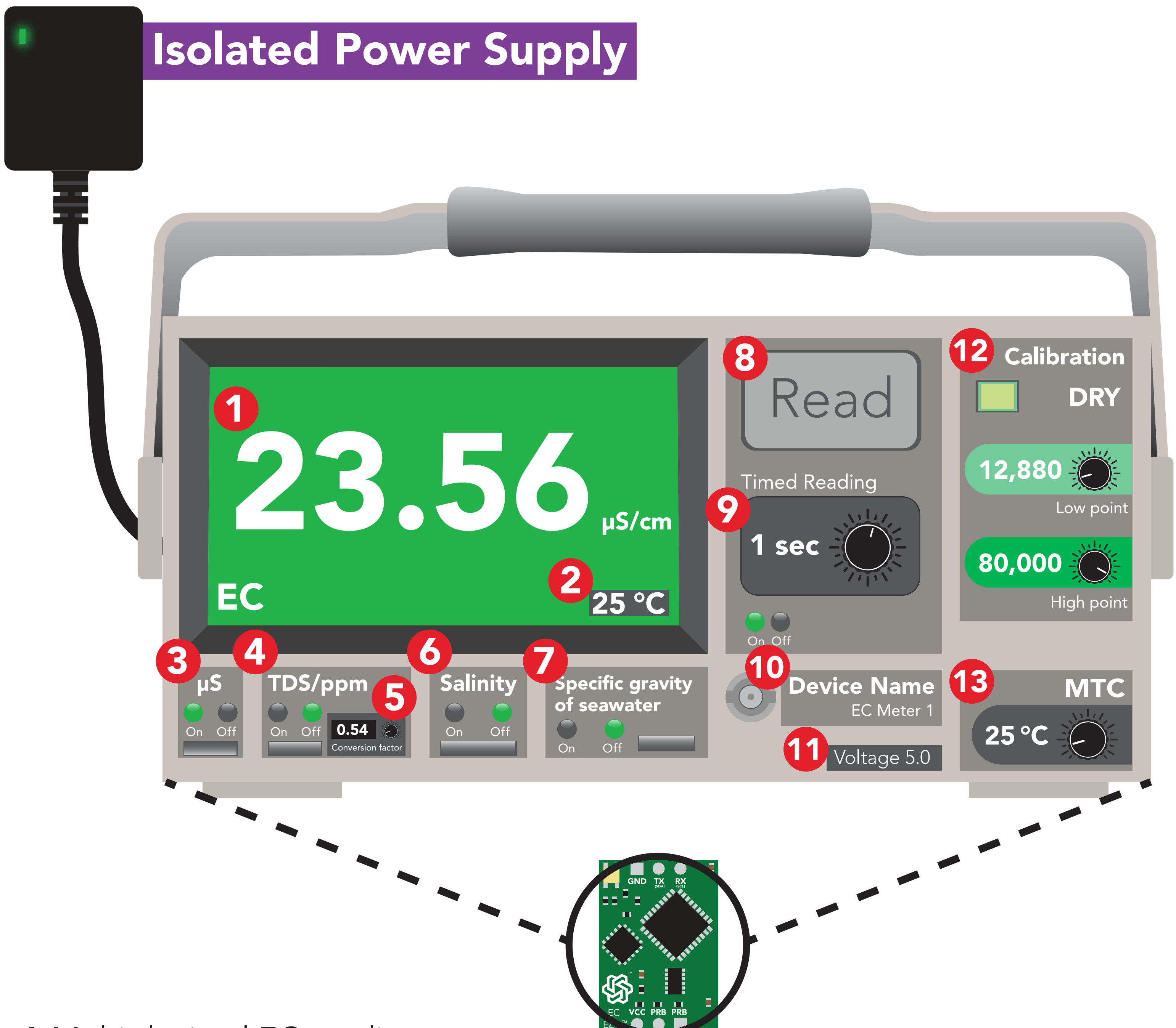


PATENT PROTECTED

Written by Jordan Press
Designed by Noah Press

This is an evolving document, check back for updates.

The EZO™ EC Circuit has all the features of this bench top meter.



1 Multi decimal EC reading

2 Temperature used for reading

3 Enable EC readings

4 Enable TDS/ ppm readings

5 Variable TDS conversion factor

6 Enable salinity readings

7 Enable specific gravity readings

8 Immediate reading

9 Timed readings

10 Set device name

11 Voltage usage

12 Multi-point variable calibration

13 Temperature compensation

The EZO Complete-EC™ is compatible with any brand of EC probe from K 0.01 – K10.2

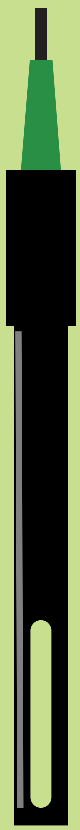
Conductivity probe range

The EZO™ Conductivity circuit is compatible with any brand of two-conductor conductivity probe, ranging from:

K 0.01 → K 10.2

Atlas Scientific™ has tested three different K value probe types:

K 0.1



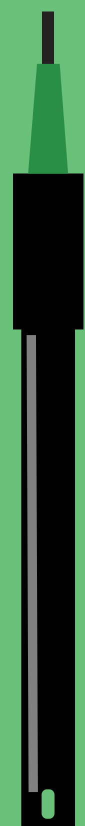
accurate reading range

0.07μS/cm – 50,000μS/cm

TDS *(ppm)* 0 – 25,000

Salinity *(ppt)* 0 – 33

K 1.0



accurate reading range

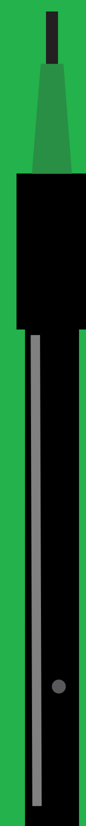
5μS/cm – 200,000+μS/cm

TDS *(ppm)* 2 – 100,000

Salinity *(ppt)* 0 – 42*

****salinity scale cannot go any higher***

K 10



accurate reading range

10μS/cm – 1S/cm

TDS *(ppm)* 5 – 500,000

Salinity *(ppt)* 0 – 42*

****salinity scale cannot go any higher***

Atlas Scientific™ does not know what the accurate reading range would be for conductivity probes, other than the above mentioned values. Determining the accurate reading range of such probes, i.e. **K 2.6**, or **K 0.66**, is the responsibility of the embedded systems engineer.

Resolution

The EZO™ Conductivity circuit, employs a method of scaling resolution. As the conductivity increases the resolution between readings decreases.

The EZO™ Conductivity circuit will output conductivity readings where the first **4 digits** are valid and the others are set to 0. This excludes conductivity readings that are less than 9.99. In that case, only 3 conductivity digits will be output.

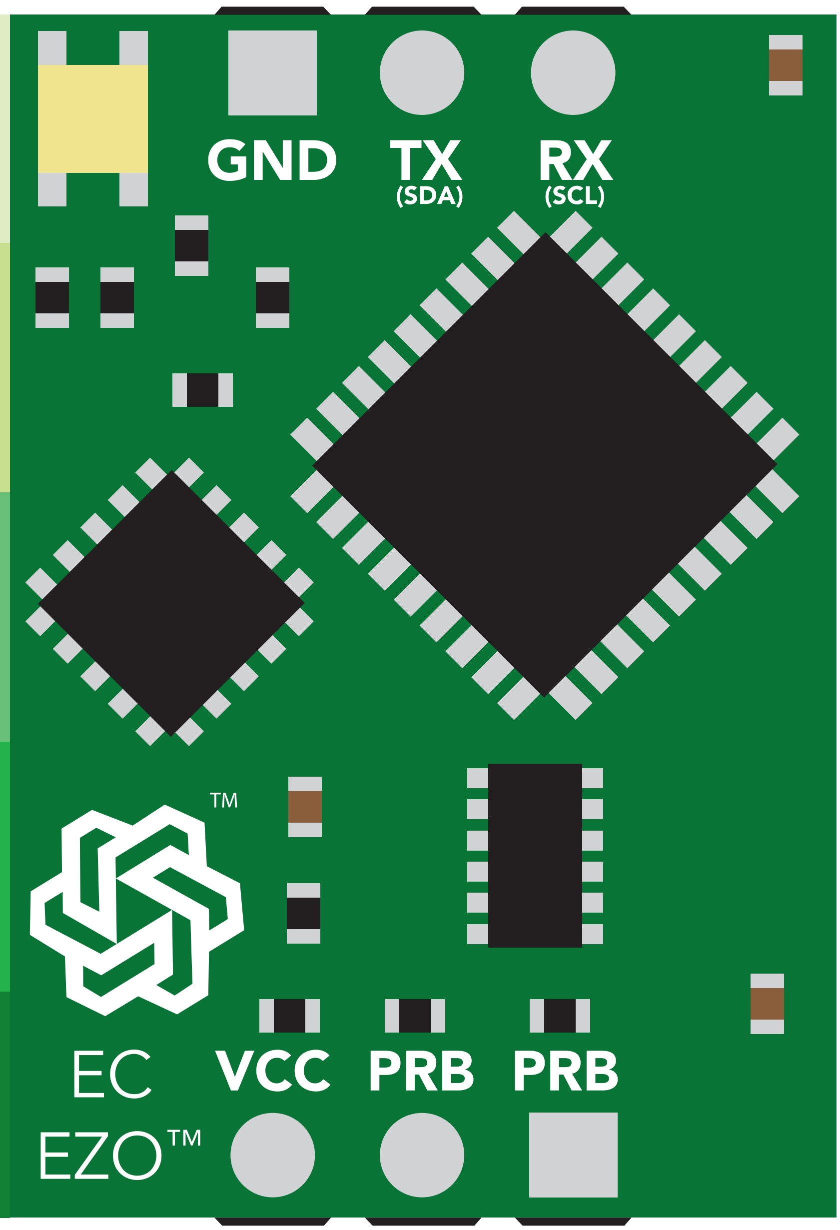
0.07 – 99.99
Resolution = **0.01µS/cm**

100.1 – 999.9
Resolution = **0.1µS/cm**

1,000 – 9,999
Resolution = **1.0µS/cm**

10,000 – 99,990
Resolution = **10µS/cm**

100,000 – 999,900
Resolution = **100µS/cm**



Available data protocols

UART

Default

I^2C

Unavailable data protocols

SPI

Analog

RS-485

Mod Bus

4–20mA

STOP

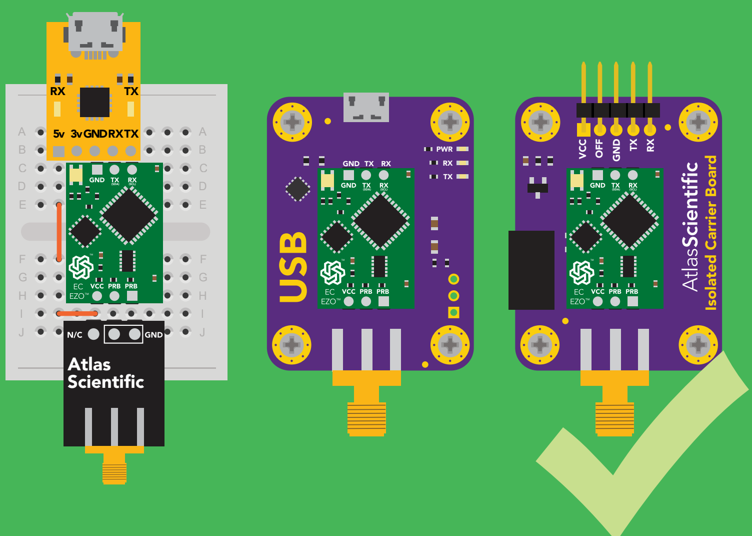
SOLDERING THIS DEVICE VOIDS YOUR WARRANTY.

Are there specific soldering instructions? Yes, see page 73.

Can you make a warranty claim after soldering? No.

If you have not used this product before; Observe how a properly working sensor behaves *BEFORE* embedding it into your PCB.

Get this device working using one of these methods first.



Do not embed before you have experience with this sensor.

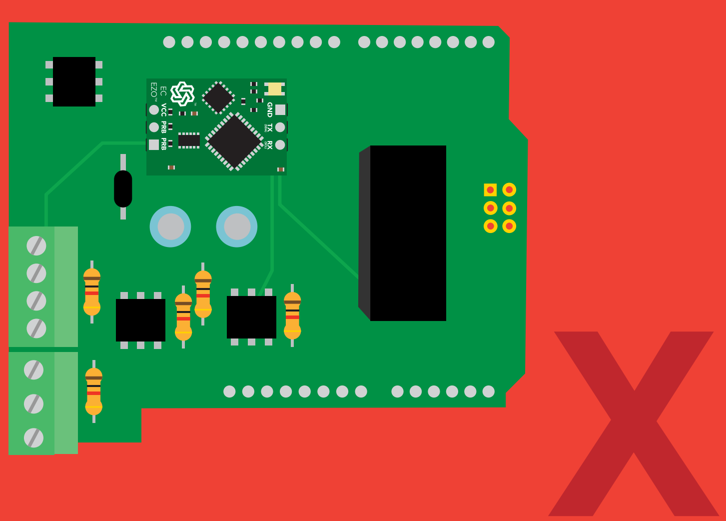
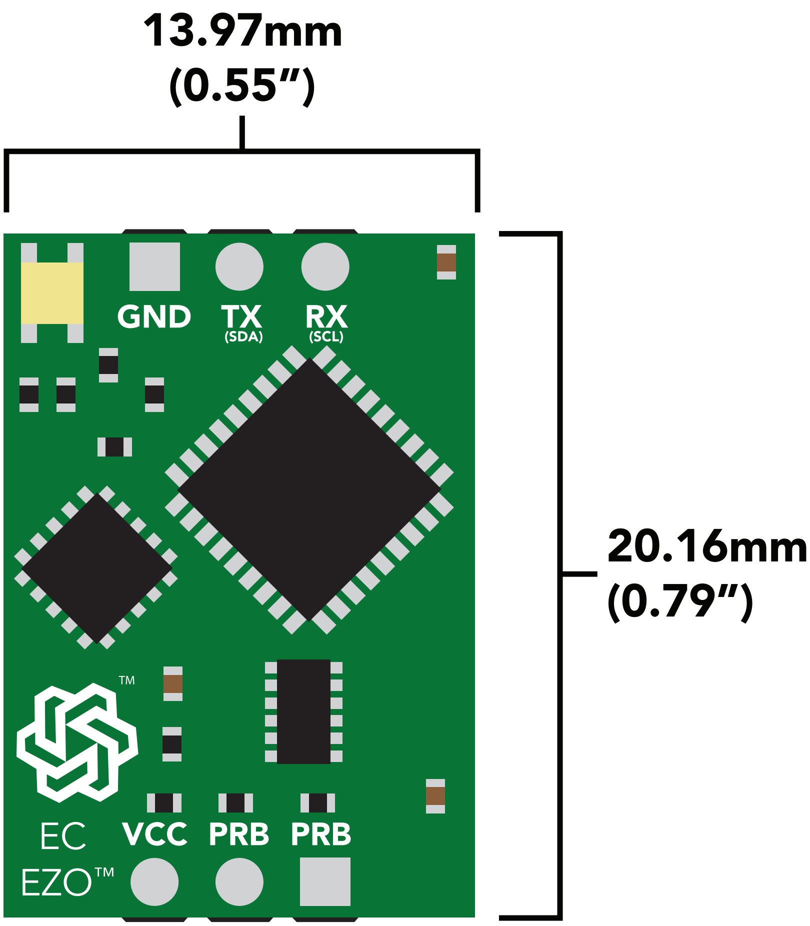


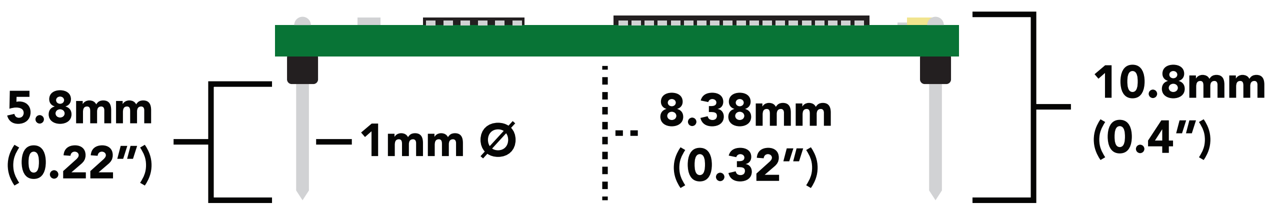
Table of contents

UART

I²C

EZO™ circuit dimensions





Power consumption

	LED	MAX	STANDBY	SLEEP
5V	ON	50 mA	18.14 mA	0.7 mA
	OFF	45 mA	15.64 mA	
3.3V	ON	35 mA	16.85 mA	0.4 mA
	OFF	34 mA	15.85 mA	

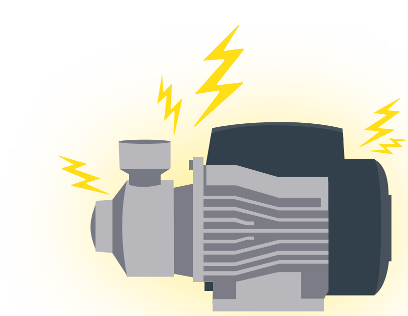
Absolute max ratings

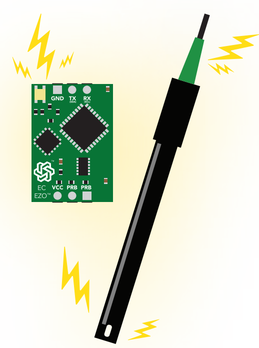
Parameter	MIN	TYP	MAX
Storage temperature (EZO™ Conductivity)	-60 °C		150 °C
Operational temperature (EZO™ Conductivity)	-40 °C	25 °C	125 °C
VCC	3.3V	5V	5.5V

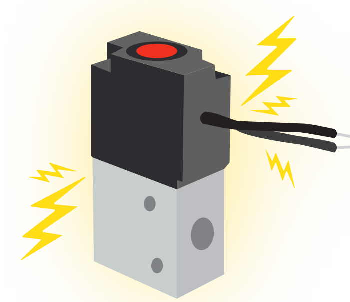
Electrical isolation

Conductivity readings will introduce significant electrical interference into your water. This electrical interference will affect other sensors, such as pH, ORP, and dissolved oxygen. Electrical isolation is 100% effective in preventing this electrical interference.

Unlike other probes, a conductivity probe provides a low-resistance pathway from your water to your electronics. If an accidental electrical surge passes through your water, it will travel up your conductivity probe and into your electronics. Electrical isolation is 100% effective at stopping an accidental electrical surge from destroying your computer system.







Advice:

When reading conductivity along with other sensors, electrical isolation is strongly recommended.
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)

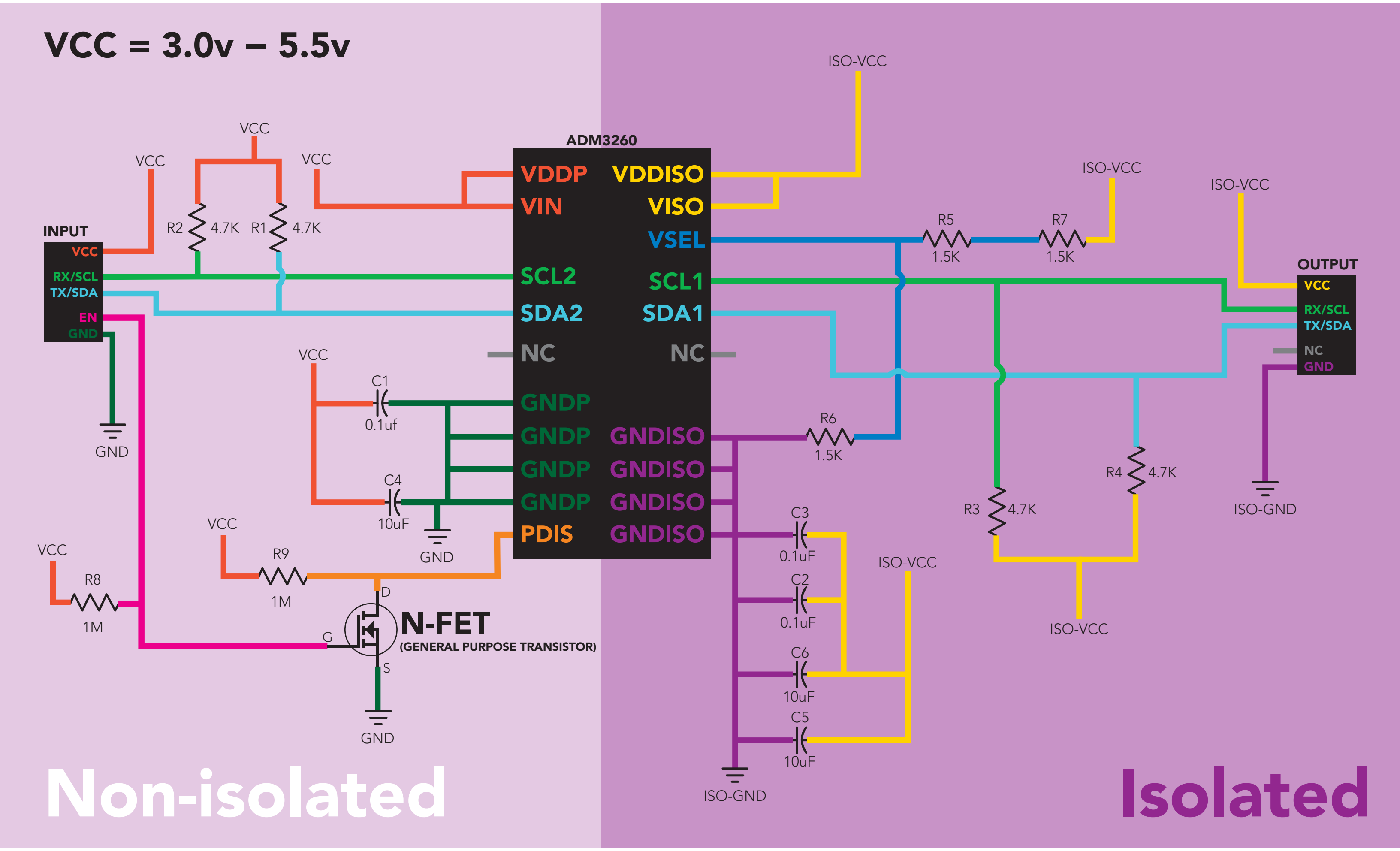
AtlasScientific
Environmental Robotics

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.

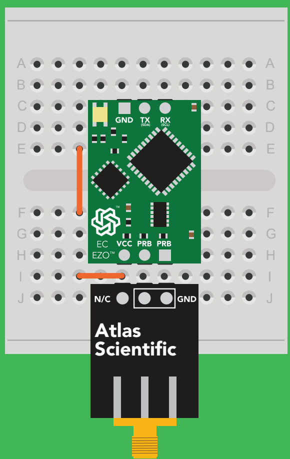
This isolation schematic is based on the ADM3260, which can output up to 150 mW of isolated power. PCB layout requires special attention for EMI/EMC and RF Control. Having good ground planes and keeping the capacitors as close to the chip as possible are crucial for proper performance.

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.

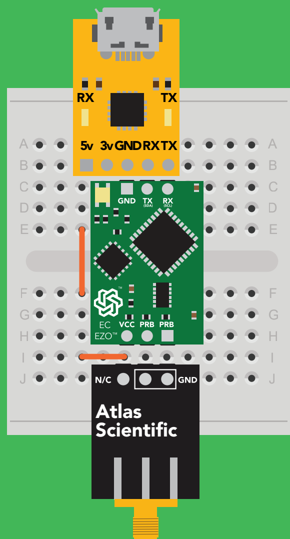
Isolated ground is different from non-isolated ground, these two lines should not be connected together.



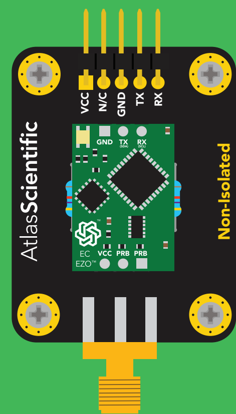
✓ Correct wiring



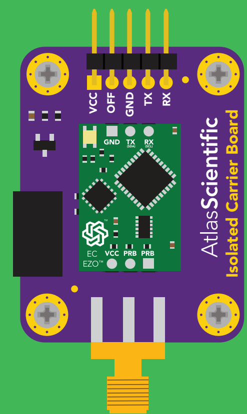
Bread board



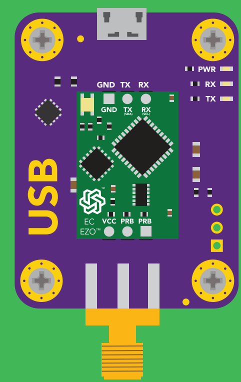
Bread board via USB



Non-Isolated
EZO™ Carrier Board



Electrically Isolated
EZO™ Carrier Board

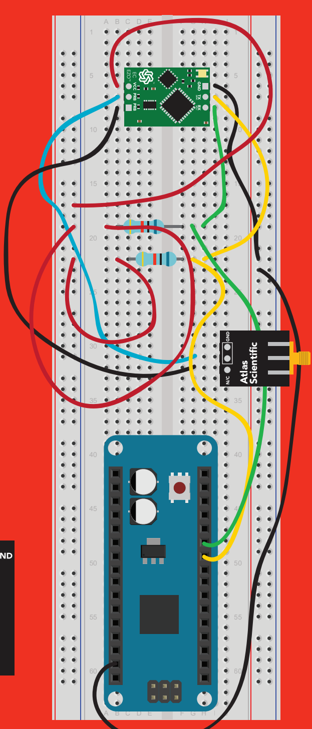


USB
carrier board

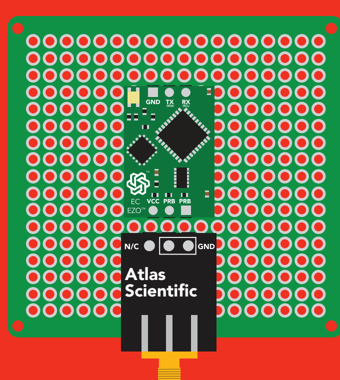
X Incorrect wiring

Extended leads

Sloppy setup

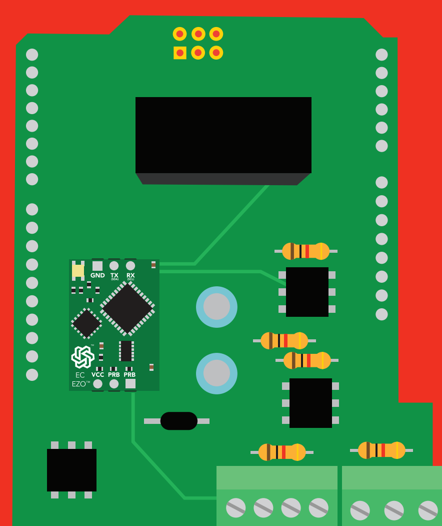


Perfboards or Protoboards



NEVER
use Perfboards or Protoboards
Flux residue and shorting wires make it very hard to get accurate readings.

*Embedded into your device

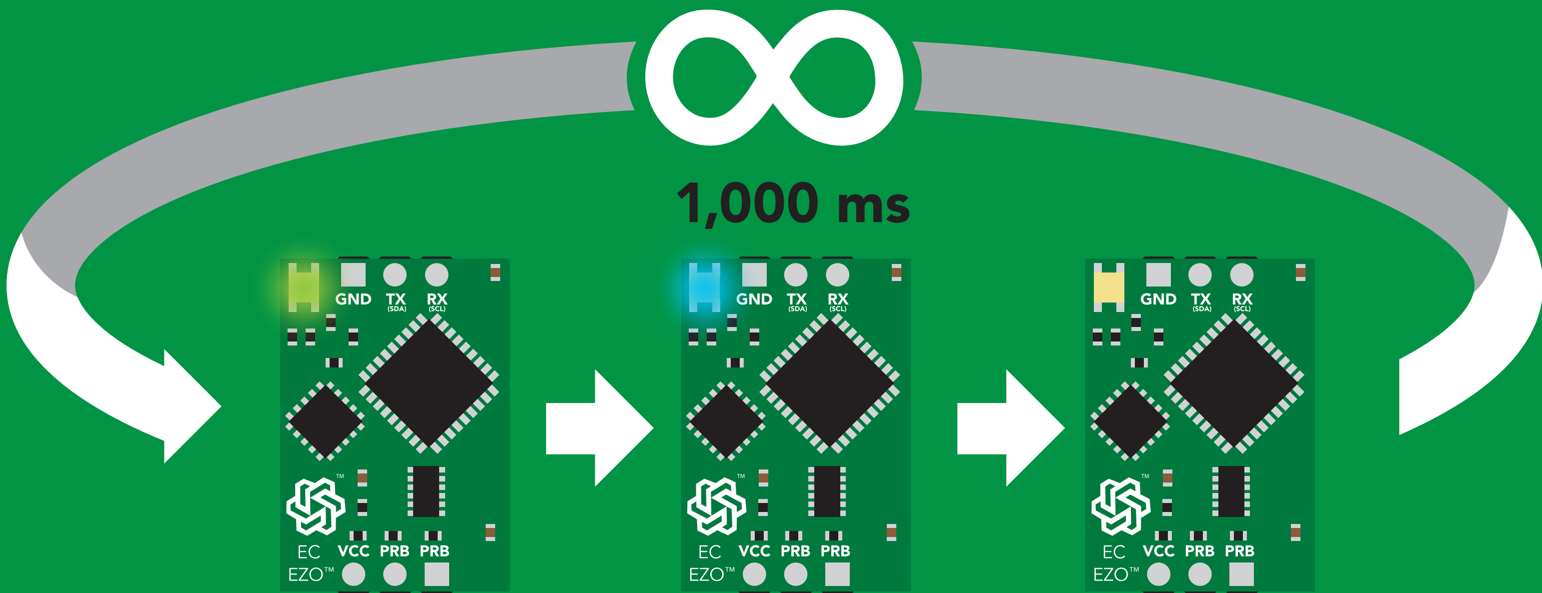


***Only after you are familar with EZO™ circuits operation**

AtlasScientific
Environmental Robotics

Default state
UART mode

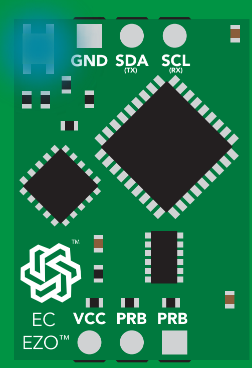
Baud	9,600
Readings	continuous
Units	μS/cm
Speed	1 reading per second
LED	on



Green
Standby

Cyan
Taking reading

Transmitting



Solid Blue LED
in I²C mode
Not UART ready

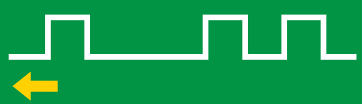
AtlasScientific™
Environmental Robotics

UART mode

8 data bits	no parity
1 stop bit	no flow control

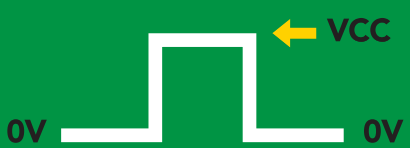
Baud
- 300
- 1,200
- 2,400
- 9,600 default
- 19,200
- 38,400
- 57,600
- 115,200

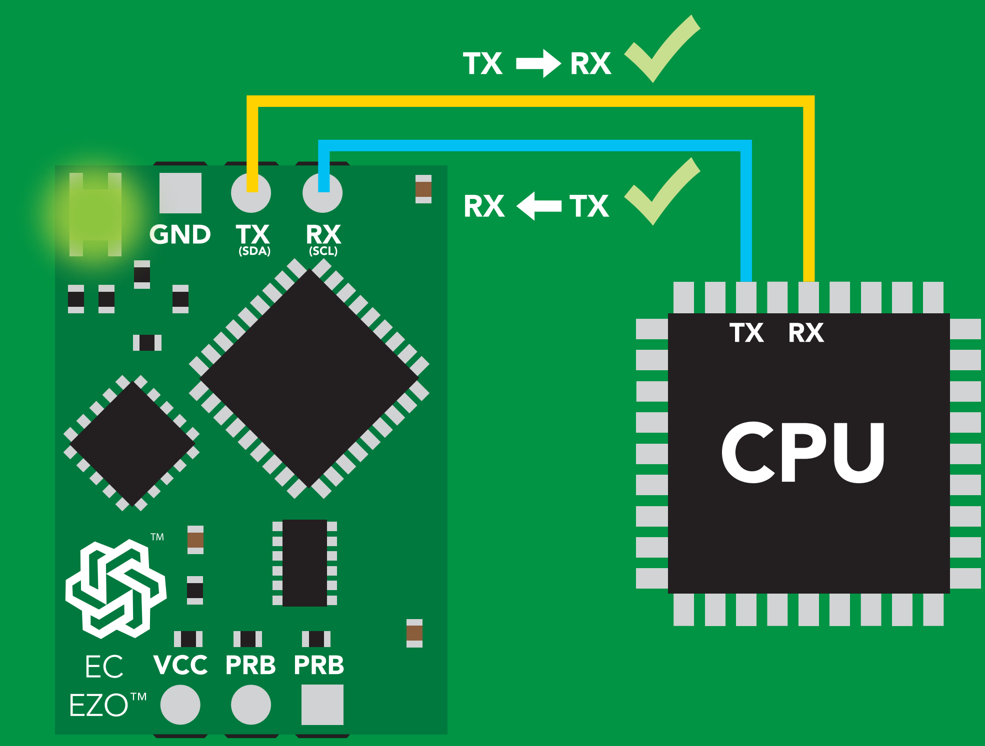
RX
Data in



TX
Data out

Vcc 3.3V – 5.5V





Data format

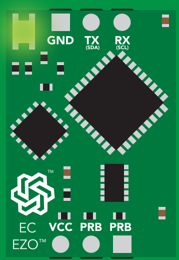
Reading

Conductivity = Deafult

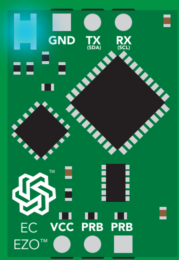
Total dissolved solids
Salinity
Specific gravity
} = Must be enabled

Order	EC,TDS,SAL,SG
Encoding	ASCII
Format	string
Terminator	carriage return
Data type	floating point
Decimal places	3
Smallest string	3 characters
Largest string	40 characters

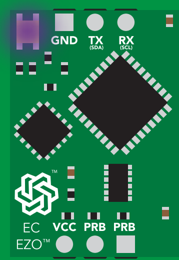
LED color definition



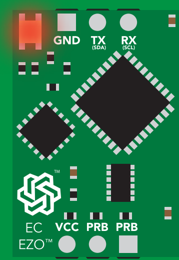
Green
UART standby



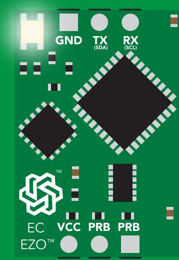
Cyan
Taking reading



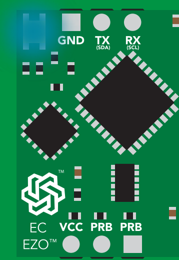
Purple
Changing baud rate



Red
Command not understood



White
Find



Blue
I2C standby

	LED ON
5V	**+2.5 mA**
3.3V	**+1 mA**

Settings that are retained if power is cut

Baud rate
Calibration
Continuous mode
Device name
Enable/disable parameters
Enable/disable response codes
Hardware switch to I²C mode
LED control
Protocol lock
Software switch to I²C mode

Settings that are *NOT* retained if power is cut

Find
Sleep mode
Temperature compensation

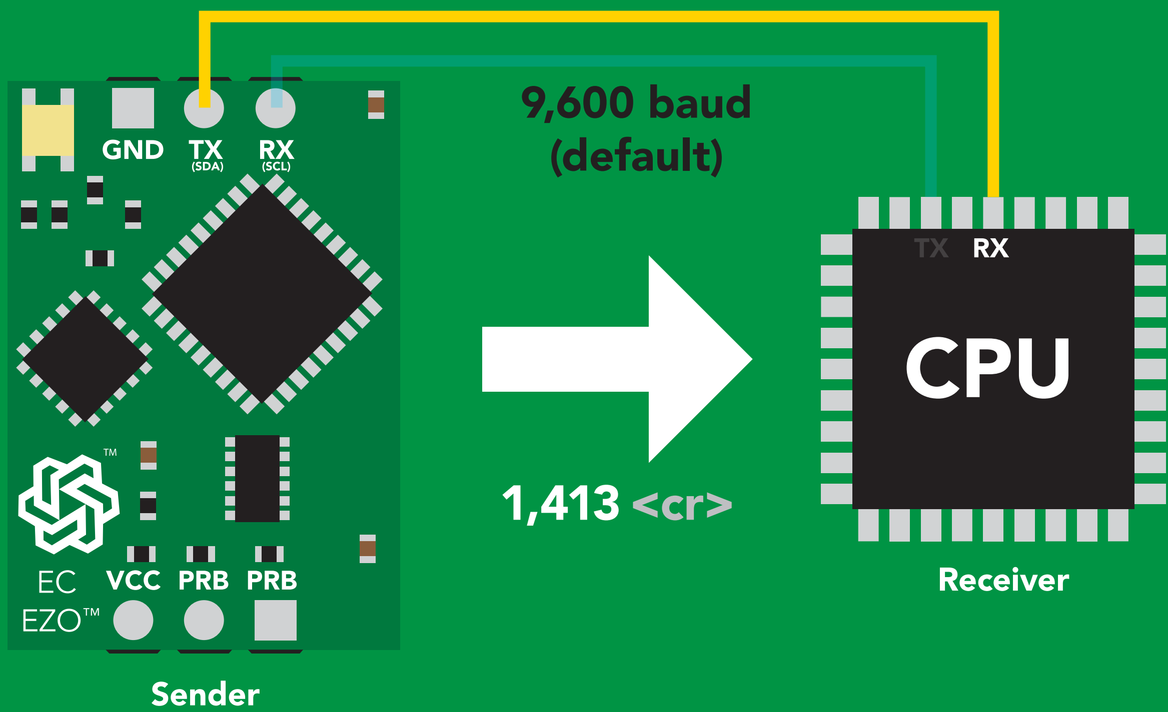
AtlasScientific™
Environmental Robotics

Receiving data from device

2 parts

ASCII data string — Command

Carriage return <cr> — Terminator



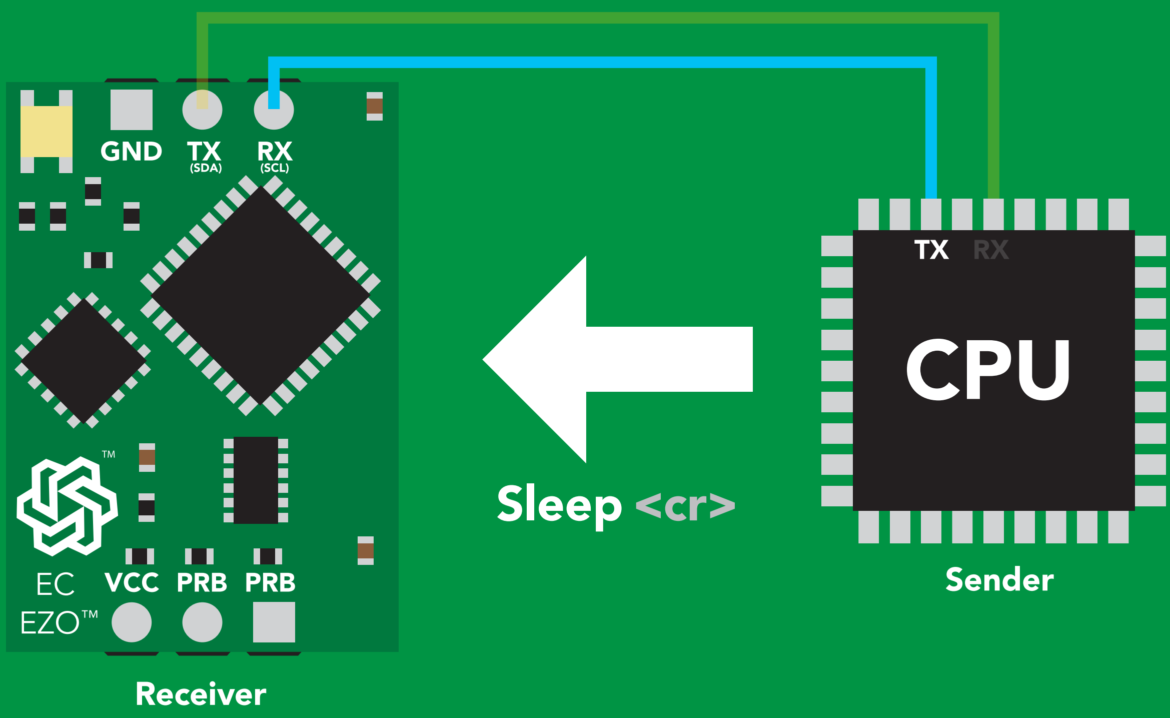
Advanced

ASCII:	1	,	4	1	3	<cr>
Hex:	31	2C	34	31	33	0D
Dec:	49	44	52	49	51	13

Sending commands to device

2 parts

Command (not case sensitive)	Carriage return <cr>
ASCII data string	Terminator



Advanced

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

UART mode

command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function		Default state
Baud	change baud rate	pg. 34	9,600
C	enable/disable continuous reading	pg. 20	enabled
Cal	performs calibration	pg. 22	n/a
Export	export calibration	pg. 24	n/a
Factory	enable factory reset	pg. 36	n/a
Find	finds device with blinking white LED	pg. 19	n/a
i	device information	pg. 30	n/a
I2C	change to I^2C mode	pg. 37	not set
Import	import calibration	pg. 25	n/a
K	Set probe type	pg. 26	K 1.0
L	enable/disable LED	pg. 18	enabled
Name	set/show name of device	pg. 29	not set
O	enable/disable parameters	pg. 28	all enabled
Plock	enable/disable protocol lock	pg. 35	disabled
R	returns a single reading	pg. 21	n/a
Sleep	enter sleep mode/low power	pg. 33	n/a
Status	retrieve status information	pg. 32	enable
T	temperature compensation	pg. 27	25°C
TDS	change the TDS conversion factor	pg. 23	0.54
*OK	enable/disable response codes	pg. 31	enable

LED control

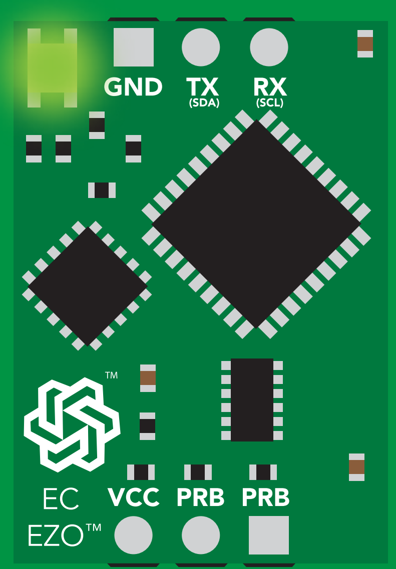
Command syntax

L,1 <cr> LED on **default**

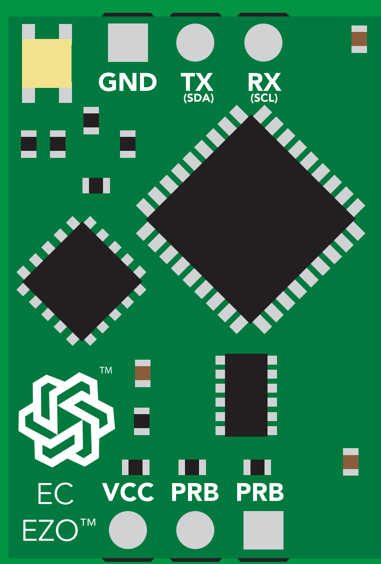
L,0 <cr> LED off

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> *OK <cr>



L,1



L,0

AtlasScientific
Environmental Robotics

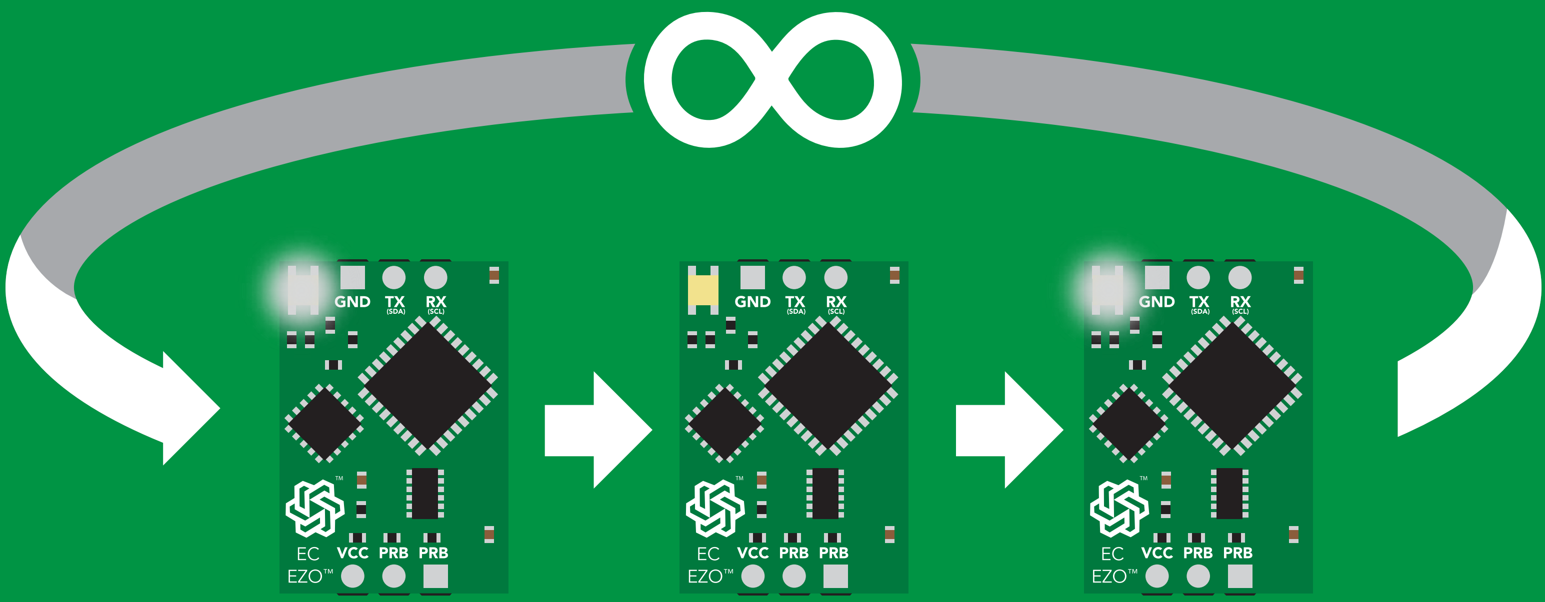
Find

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?	

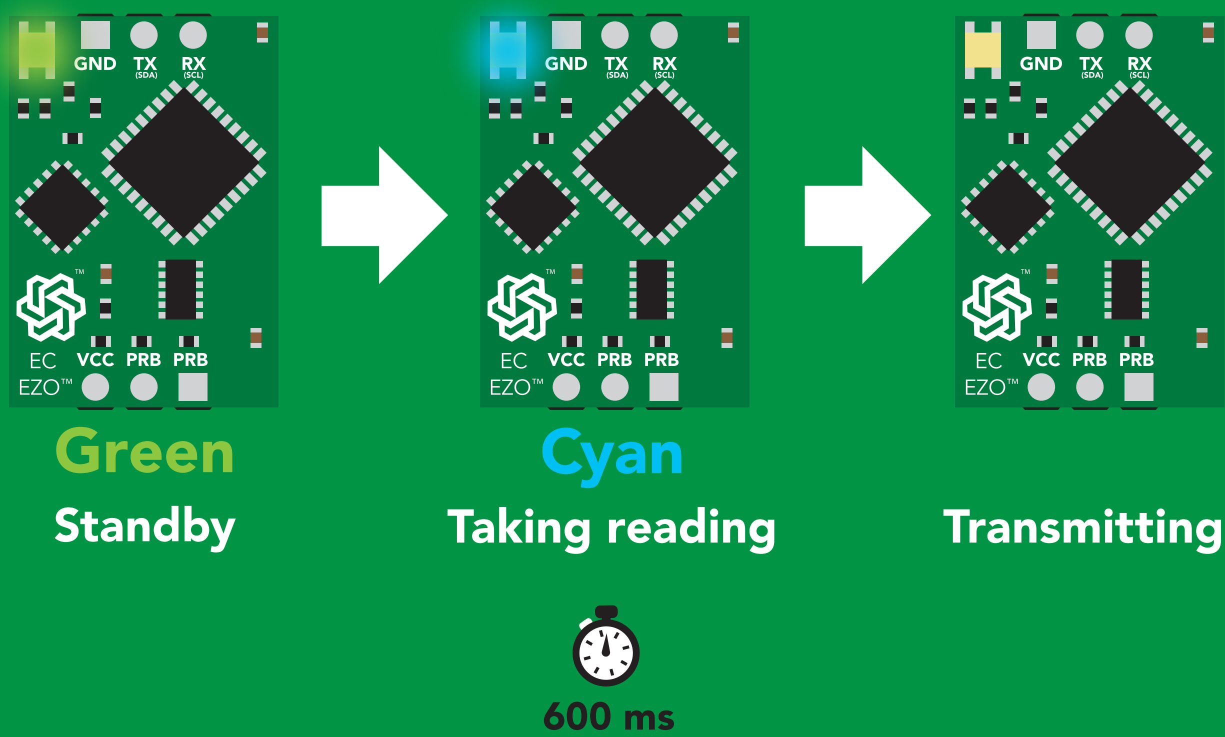
Example	Response
C,1 <cr>	*OK <cr> EC,TDS,SAL,SG (1 sec) <cr> EC,TDS,SAL,SG (2 sec) <cr> EC,TDS,SAL,SG (3 sec) <cr>
C,30 <cr>	*OK <cr> EC,TDS,SAL,SG (30 sec) <cr> EC,TDS,SAL,SG (60 sec) <cr> EC,TDS,SAL,SG (90 sec) <cr>
C,0 <cr>	*OK <cr>
C,? <cr>	?C,1 <cr> or ?C,0 <cr> or ?C,30 <cr> *OK <cr>

Single reading mode

Command syntax

R <cr> **takes single reading**

Example	Response
R <cr>	1,413 <cr> *OK <cr>



Calibration

Command syntax

Dry calibration must always be done first!

Cal,dry	<cr>	dry calibration
Cal,n	<cr>	single point calibration, where n = any value
Cal,low,n	<cr>	low end calibration, where n = any value
Cal,high,n	<cr>	high end calibration, where n = any value
Cal,clear	<cr>	delete calibration data
Cal,?	<cr>	device calibrated?

Example	Response
Cal,dry <cr>	*OK <cr>
Cal,84 <cr>	*OK <cr>
Cal,low,12880 <cr>	*OK <cr>
Cal,high,80000 <cr>	*OK <cr>
Cal,clear <cr>	*OK <cr>
Cal,? <cr>	?CAL,0 <cr> or ?CAL,1 <cr> (two point) or ?CAL,2 (three point) *OK <cr>

Two point calibration:
Step 1. "cal,dry"
Step 2. "cal,n"
Calibration complete!

Three point calibration:
Step 1 "cal,dry"
Step 2 "cal,low,n"
Step 3 "cal,high.n"
Calibration complete!

AtlasScientific
Environmental Robotics

Changing the TDS (ppm) conversion factor

There are several different conversion factors used to read TDS(ppm). For some applications, it may be necessary to use a conversion factor other than the default value of 0.54

Command syntax

TDS,n	<cr>	set custom conversion factor, n = any value between 0.01 – 1.00
TDS,?	<cr>	conversion factor being used

Example	Response
TDS,? <cr>	?TDS,0.54 <cr> *OK <cr>
R <cr>	EC TDS 100,54 <cr> *OK <cr>
TDS,0.46 <cr>	*OK <cr>
R <cr>	EC TDS 100,46 <cr> *OK <cr>

Common conversion factors

NaCl	0.47 – 0.50
KCL	0.50 - 0.57
"442"	0.65 – 0.85

Formula

EC x conversion factor = TDS

Export calibration

Command syntax

Export: Use this command to download calibration settings

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

Example	Response
Export,? <cr>	10,120 <cr>

Response breakdown

10, 120

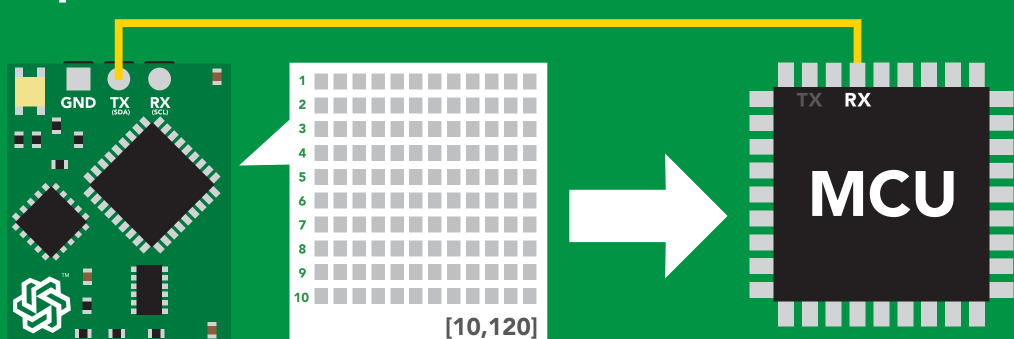
# of strings to export / # of bytes to export

Export strings can be up to 12 characters long, and is always followed by <cr>

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

Disabling *OK simplifies this process

Export <cr>



*DONE

AtlasScientific™ Environmental Robotics

Import calibration

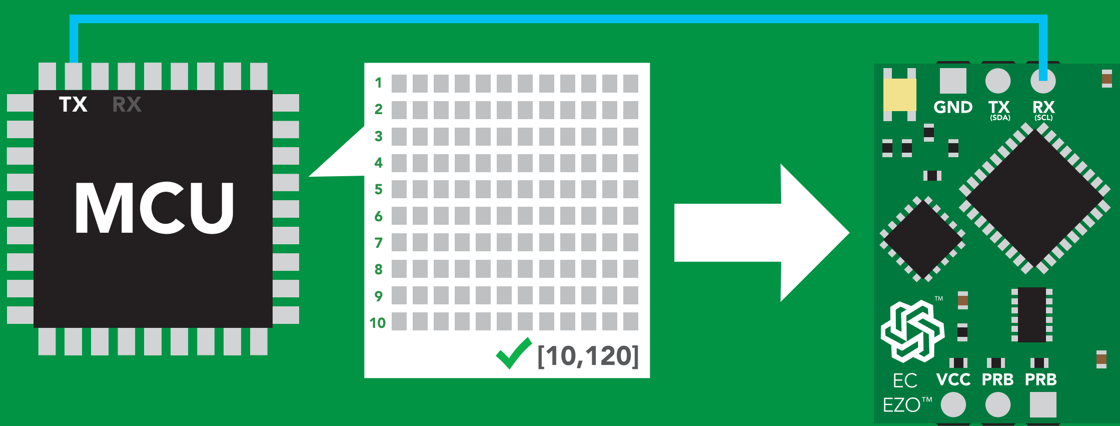
Command syntax

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>

Import,n <cr>



*OK <cr>
system will reboot

*ER <cr>

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

Setting the probe type

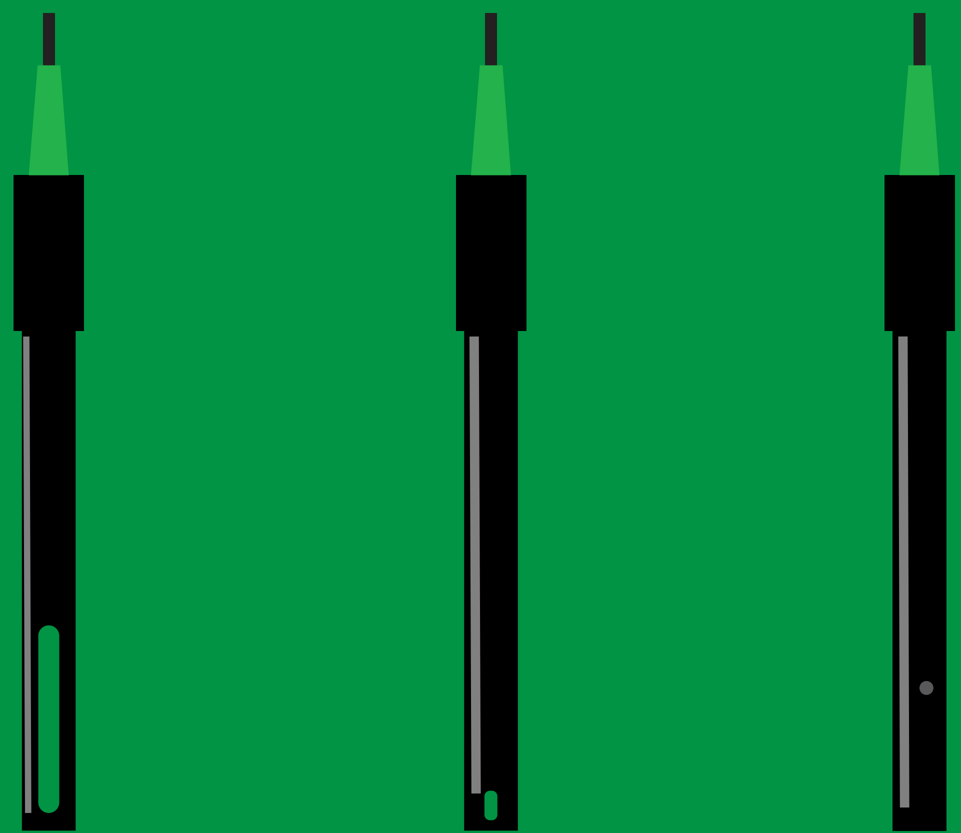
Command syntax

K 1.0 is the default value

K,n <cr> n = any value; floating point in ASCII

K,? <cr> probe K value?

Example	Response
K,10 <cr>	*OK <cr>
K,? <cr>	?K,10 <cr> *OK <cr>



K 0.1 K 1.0 K 10

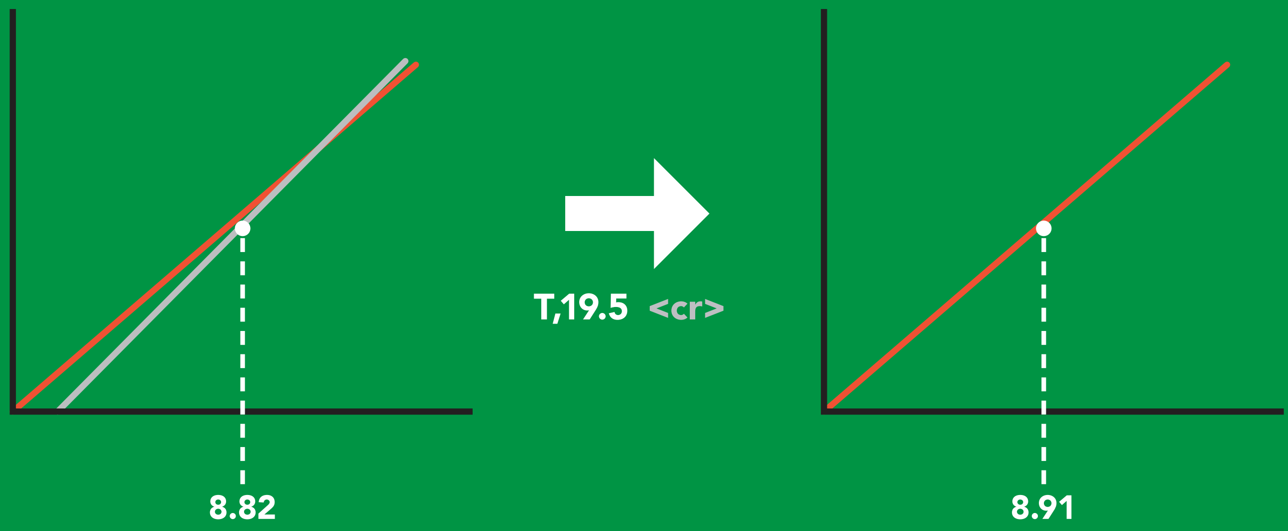
Temperature compensation

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

Command syntax

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> 8.91 <cr>
T,? <cr>	?T,19.5 <cr> *OK <cr>



Enable/disable parameters from output string

Command syntax

O, [parameter],[1,0]	<cr>	enable or disable output parameter
O,?	<cr>	enabled parameter?

Example	Response
O,EC,1 / O,EC,0 <cr>	*OK <cr> enable / disable conductivity
O,TDS,1 / O,TDS,0 <cr>	*OK <cr> enable / disable total dissolved solids
O,S,1 / O,S,0 <cr>	*OK <cr> enable / disable salinity
O,SG,1 / O,SG,0 <cr>	*OK <cr> enable / disable specific gravity
O,? <cr>	?,O,EC,TDS,S,SG <cr> if all are enabled

Parameters

EC	Conductivity = μS/cm
TDS	Total dissolved solids = ppm
S	Salinity = PSU (ppt) 0.00 – 42.00
SG	Specific gravity (sea water only) = 1.00 – 1.300

Followed by 1 or 0

1	enabled
0	disabled

*** If you disable all possible data types your readings will display "no output".**

Naming device

Command syntax

Do not use spaces in the name

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

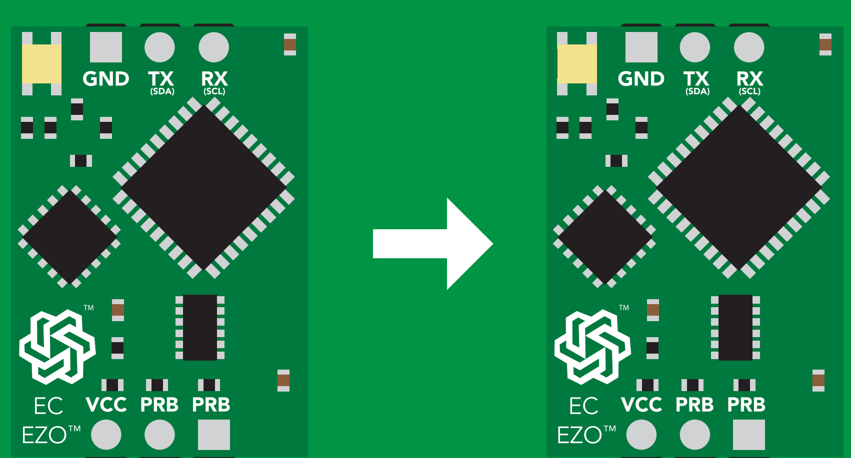
n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

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> *OK <cr>

Name,zzt

Name,?



*OK <cr>

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

AtlasScientific
Environmental Robotics

Device information

Command syntax

i <cr>	device information

Example	Response
i <cr>	?i,EC,2.16 <cr> *OK <cr>

Response breakdown

?i,	EC,	2.16
	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>	1,413 <cr> *OK <cr>
*OK,0 <cr>	no response, *OK disabled
R <cr>	1,413 <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

AtlasScientific
Environmental Robotics

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> *OK <cr>

Response breakdown

?Status,	P,	5.038
	Reason for restart	Voltage at Vcc

Restart codes

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

Sleep mode/low power

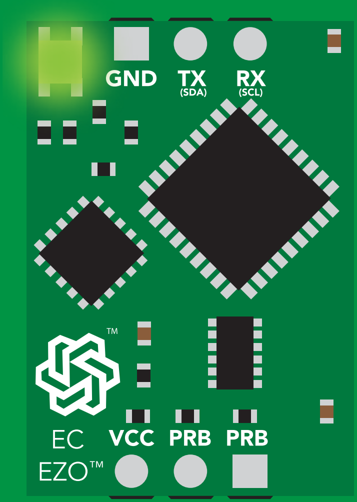
Command syntax

Send any character or command to awaken device.

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

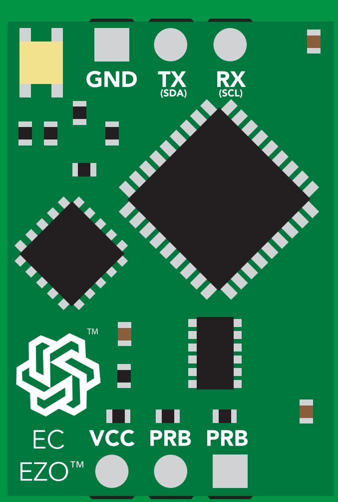
Example	Response
Sleep <cr>	*OK <cr> *SL <cr>
Any command	*WA <cr> wakes up device

	STANDBY	SLEEP
5V	**18.14 mA**	**0.7 mA**
3.3V	**16.85 mA**	**0.4 mA**



Standby
18.14 mA

Sleep <cr>



Sleep
0.7 mA

AtlasScientific™ Environmental Robotics

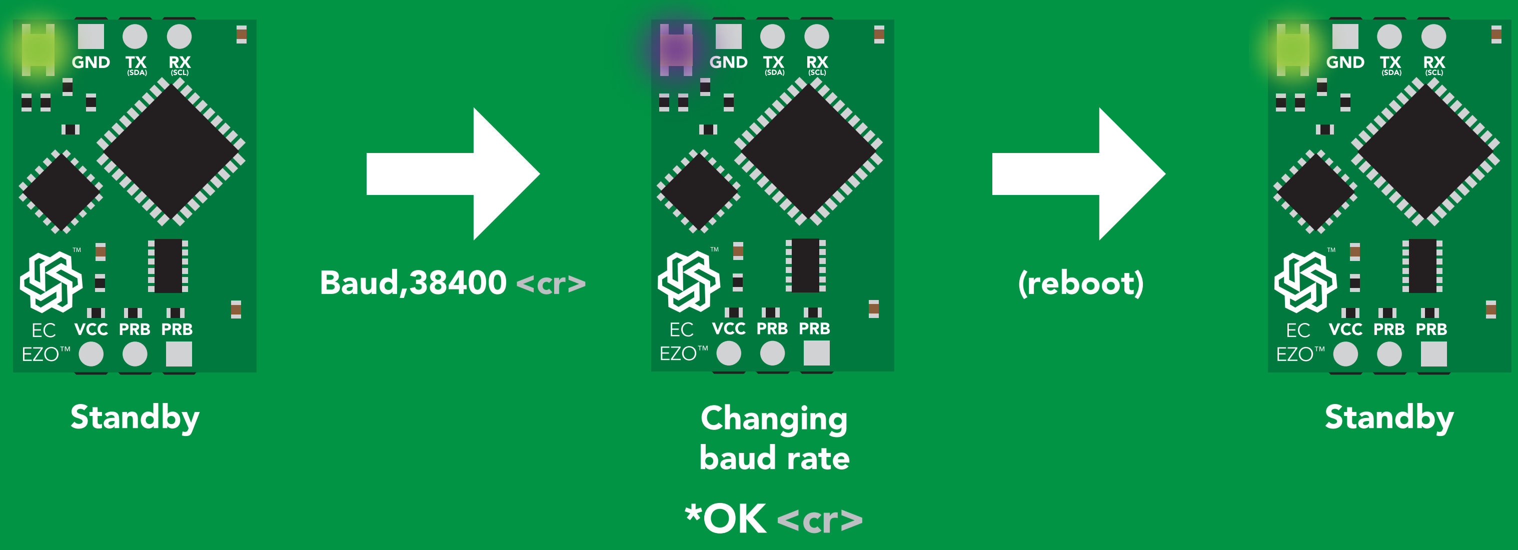
Change baud rate

Command syntax

Baud,n <cr> change baud rate

Example	Response
Baud,38400 <cr>	*OK <cr>
Baud,? <cr>	?Baud,38400 <cr> *OK <cr>

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



Protocol lock

Command syntax

Locks device to UART mode.

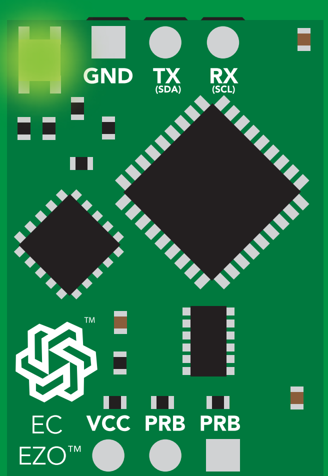
Plock,1 <cr> enable Plock

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

Plock,? <cr> Plock on/off?

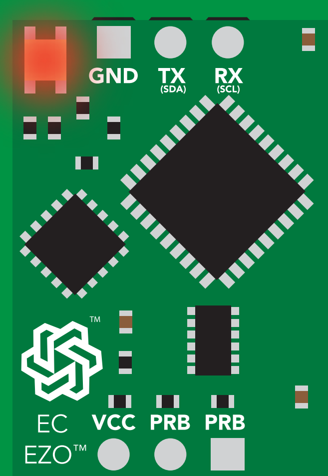
Example	Response
Plock,1 <cr>	*OK <cr>
Plock,0 <cr>	*OK <cr>
Plock,? <cr>	?Plock,1 <cr> or ?Plock,0 <cr>

Plock,1



*OK <cr>

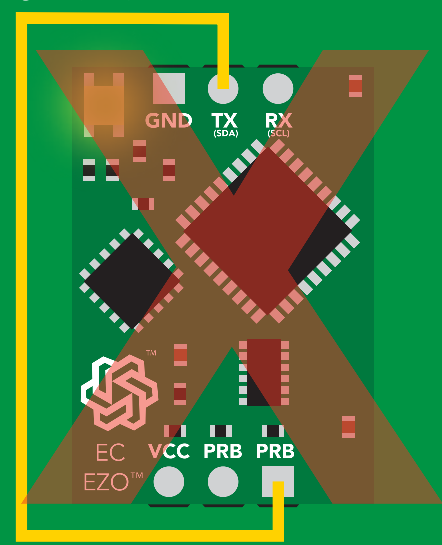
I2C,100



cannot change to I²C

*ER <cr>

Short



cannot change to I²C

AtlasScientific
Environmental Robotics

Factory reset

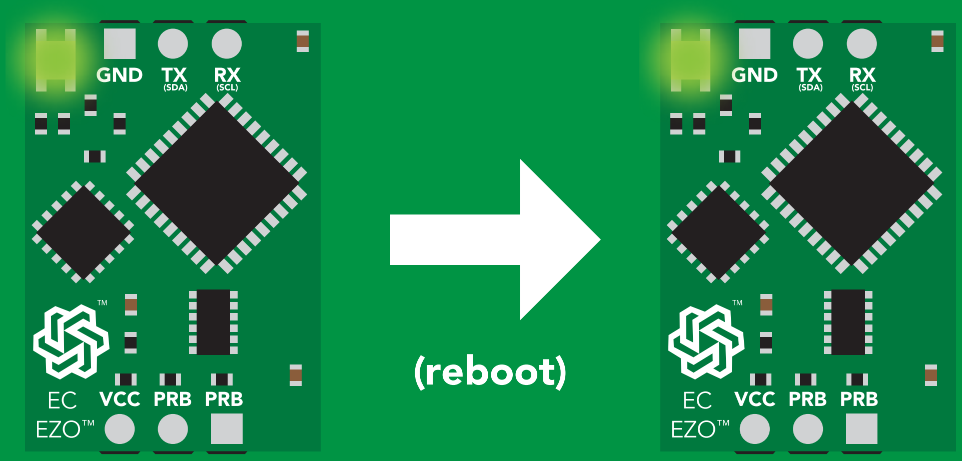
Clears calibration
LED on
"*OK" enabled

Command syntax

Factory <cr> enable factory reset

Example	Response
Factory <cr>	*OK <cr>

Factory <cr>



*OK <cr>

*RS <cr>
*RE <cr>

Baud rate will not change

Change to I^{2}C mode

Command syntax

Default I^{2}C address 100 (0x64)

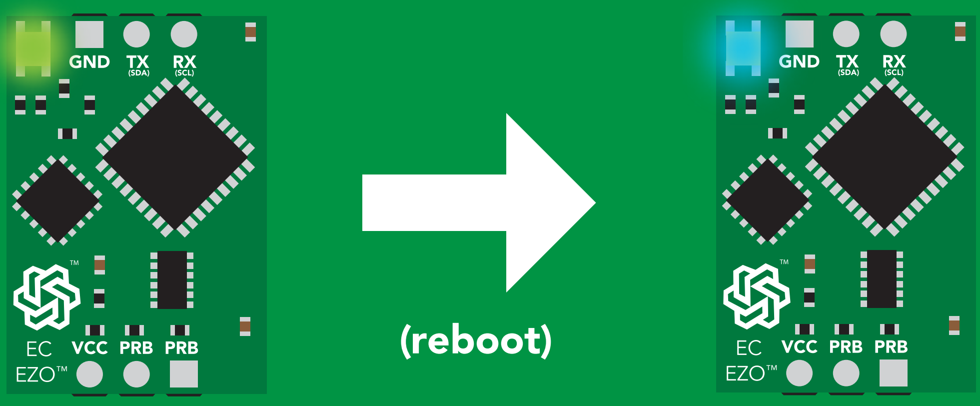
I2C,n <cr> sets I^{2}C address and reboots into I^{2}C mode

n = any number 1 – 127

Example	Response
I2C,100 <cr>	*OK (reboot in I^{2}C mode)

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

I2C,100



Green
*OK <cr>

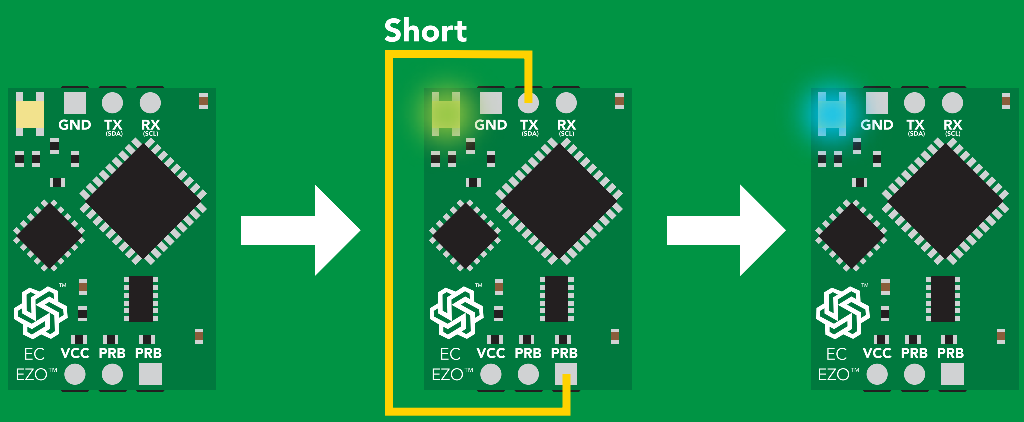
Blue
now in I^{2}C mode

Manual switching to I^2C

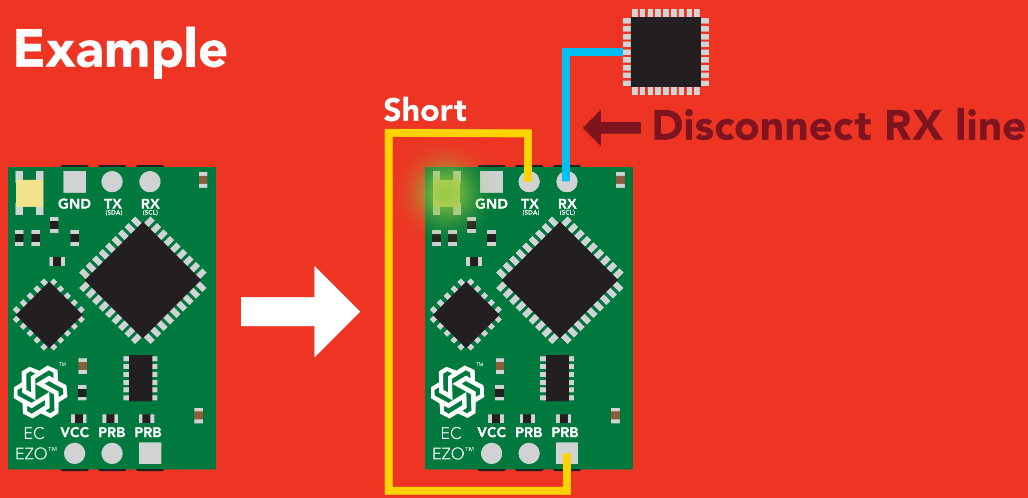
- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to the right PRB
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from Green to Blue
- Disconnect ground (power off)
- Reconnect all data and power

Manually switching to I^2C will set the I^2C address to 100 (0x64)

Example



Wrong Example



I^2C mode

The I^2C 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™ device into I^2C mode click here

Settings that are retained if power is cut

Calibration
Change I^2C address
Enable/disable parameters
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
Temperature compensation

I²C mode

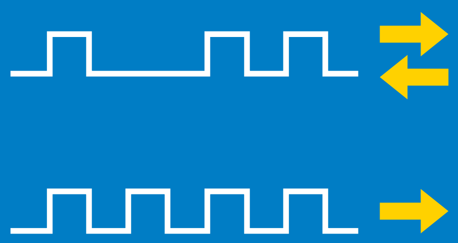
I²C address (0x01 – 0x7F)
100 (0x64) default

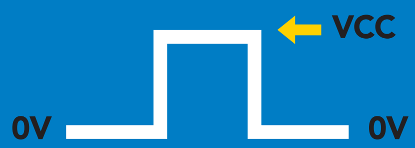
Vcc 3.3V – 5.5V

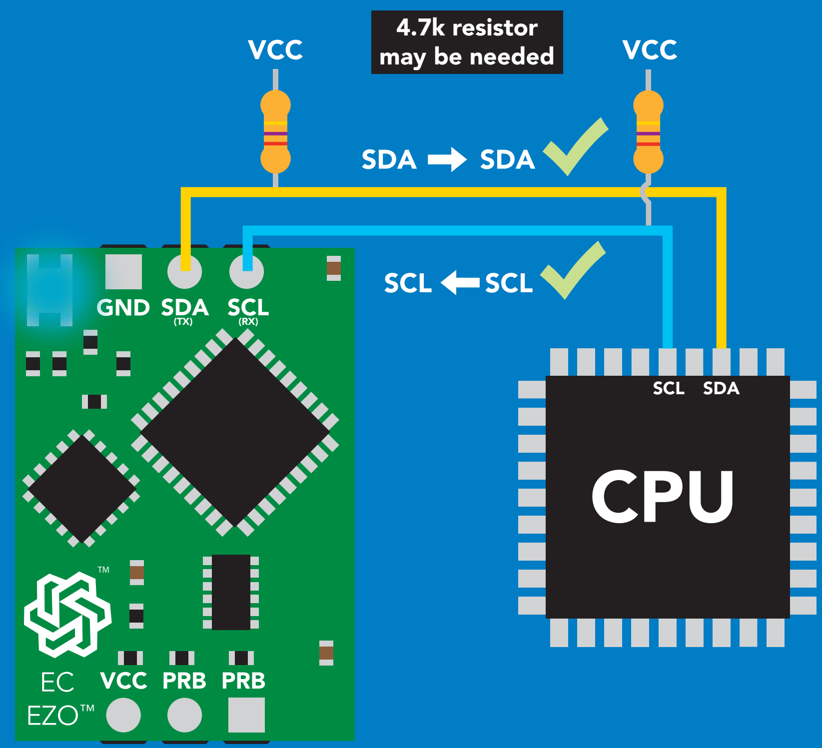
Clock speed 100 – 400 kHz

SDA

SCL







Data format

Reading

Conductivity = Deafult

Total dissolved solids
Salinity
Specific gravity
} = Must be enabled

Order EC,TDS,SAL,SG

Encoding ASCII

Format	string
Data type	floating point
Decimal places	3
Smallest string	3 characters
Largest string	40 characters

Sending commands to device

5 parts

Start | **I²C address** | **Write** | **Command (not case sensitive)** | **Stop**

I²C address: 100 (0x64)

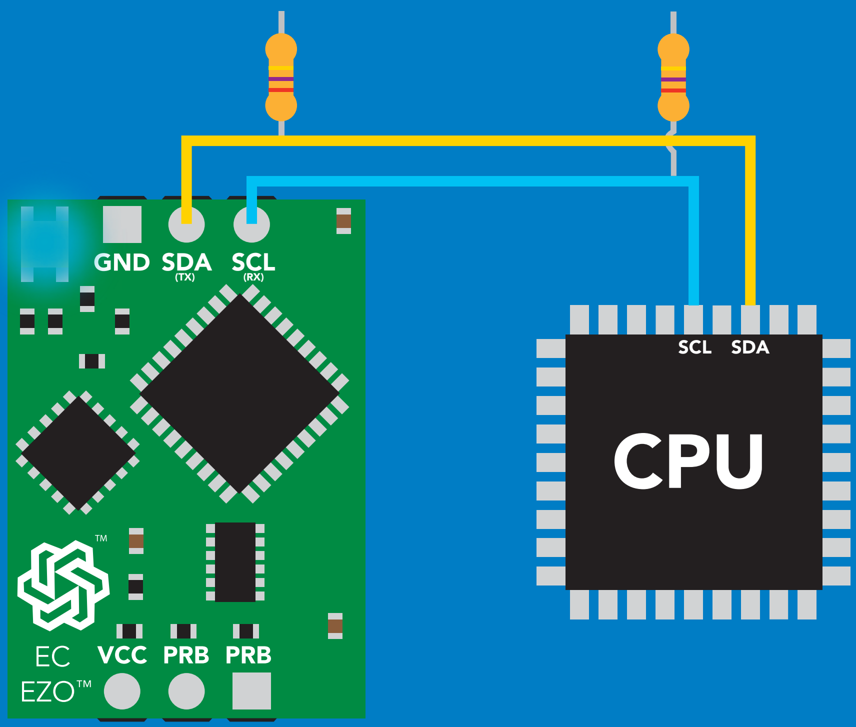
Command: ASCII command string

Example

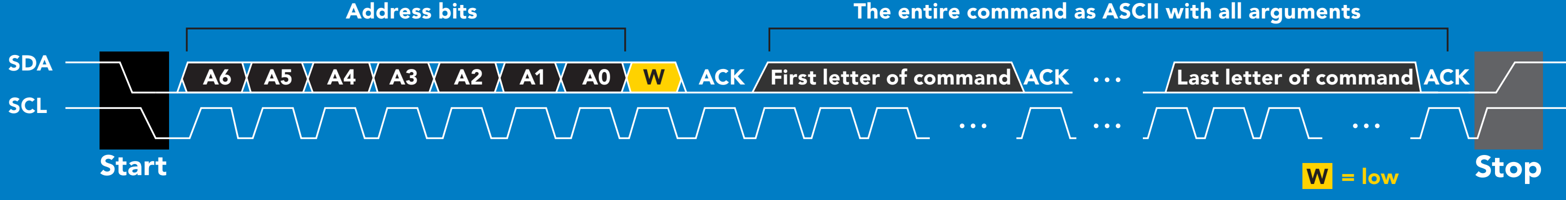
Start | **100 (0x64)** | **Write** | **Sleep** | **Stop**

100 (0x64): I²C address

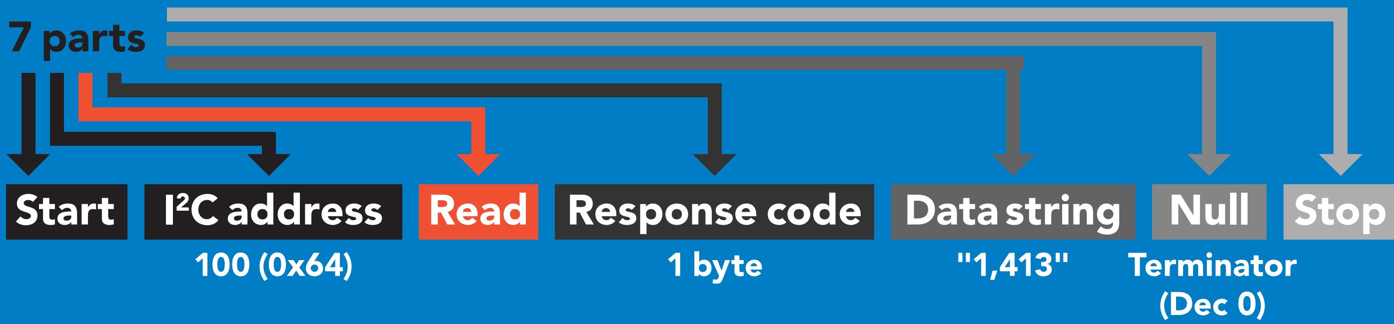
Sleep: Command

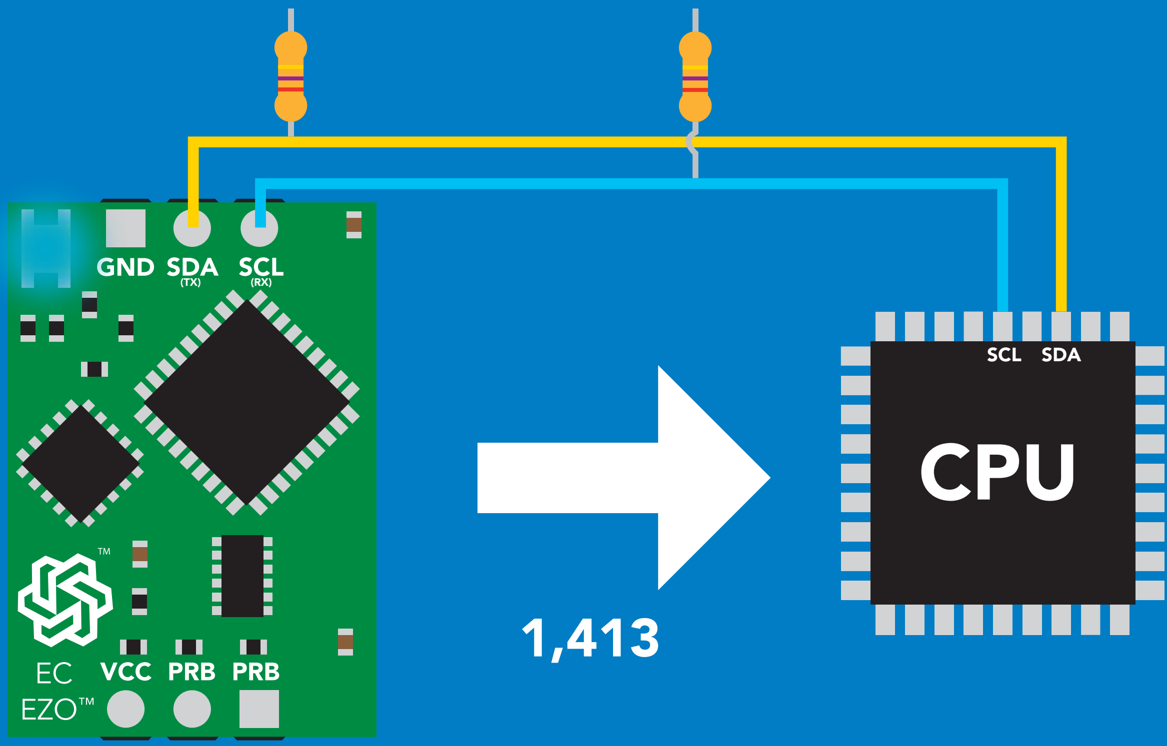


Advanced

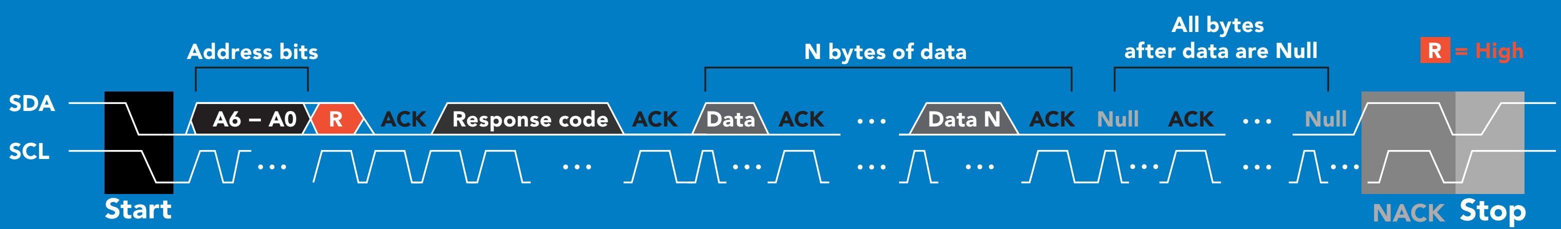


Requesting data from device





Advanced

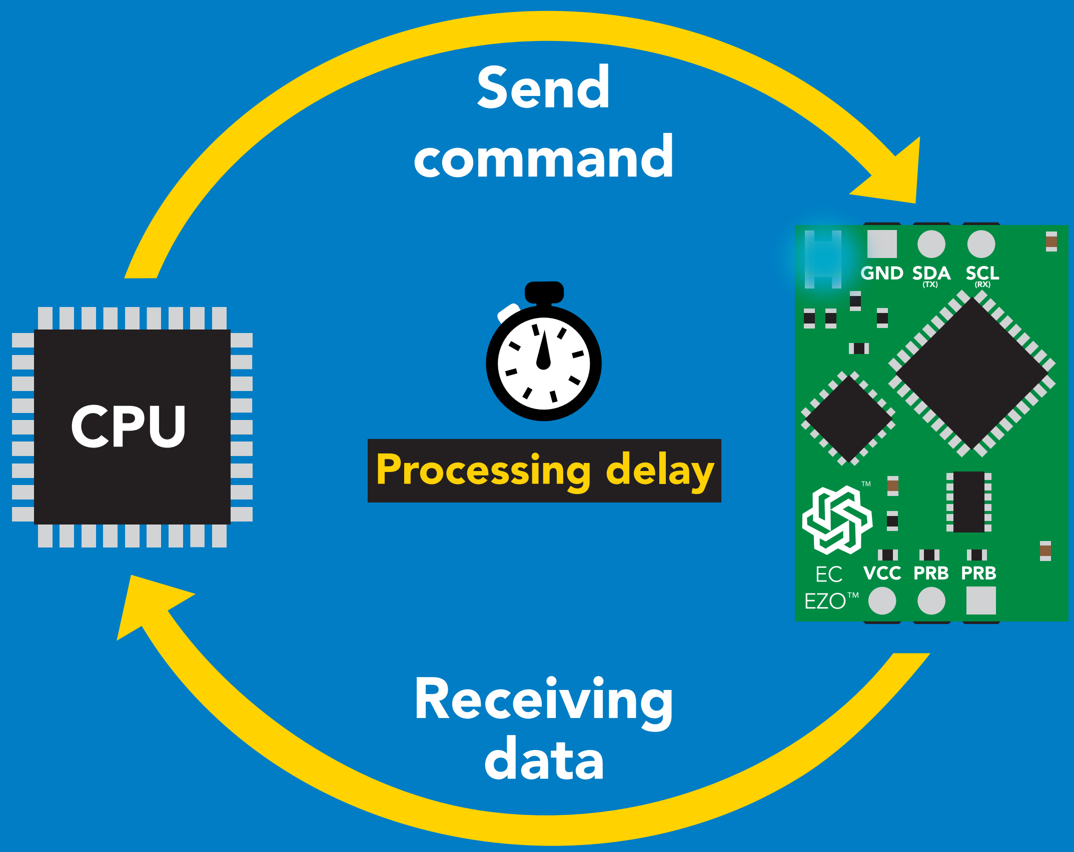


AtlasScientific™
Environmental Robotics

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);
```


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

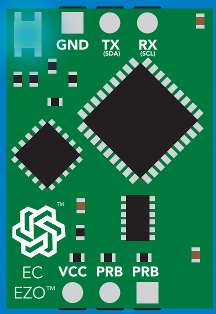
The response code will always be 254, if you do not wait for the processing delay.

Response codes

Single byte, not string

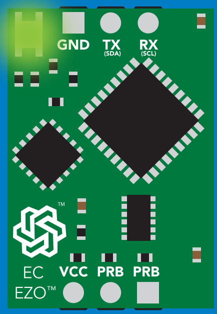
255	no data to send
254	still processing, not ready
2	syntax error
1	successful request

LED color definition



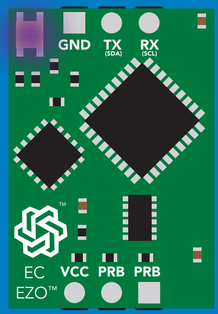
Blue

I²C standby



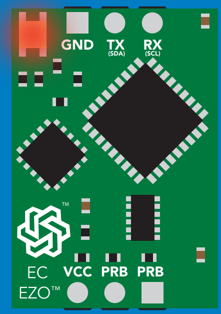
Green

Taking reading



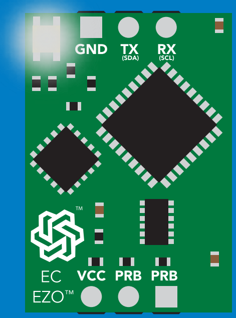
Purple

Changing I²C address



Red

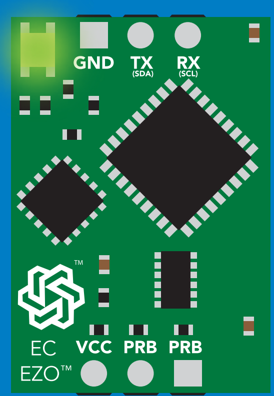
Command not understood



White

Find

	LED ON
5V	**+2.5 mA**
3.3V	**+1 mA**



Solid Green LED

in UART mode
Not I²C ready

I²C mode
command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function	
Baud	switch back to UART mode	pg. 63
Cal	performs calibration	pg. 49
Export	export calibration	pg. 51
Factory	enable factory reset	pg. 62
Find	finds device with blinking white LED	pg. 47
i	device information	pg. 57
I2C	change I²C address	pg. 61
Import	import calibration	pg. 52
K	set probe type	pg. 53
L	enable/disable LED	pg. 46
Name	set/show name of device	pg. 56
O	enable/disable parameters	pg. 55
Plock	enable/disable protocol lock	pg. 60
R	returns a single reading	pg. 48
Sleep	enter sleep mode/low power	pg. 59
Status	retrieve status information	pg. 58
T	temperature compensation	pg. 54
TDS	change the TDS conversion factor	pg. 50

AtlasScientific™
Environmental Robotics

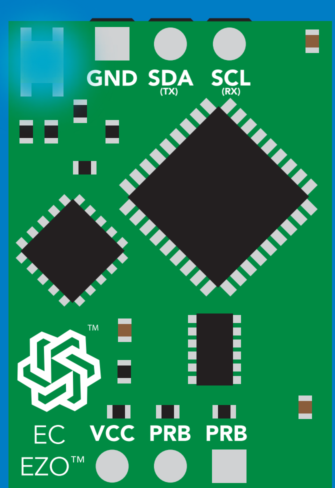
LED control

Command syntax

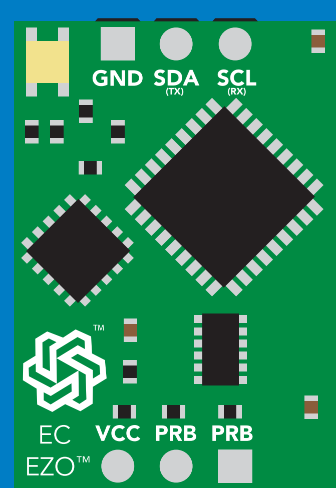
300ms processing delay

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 1 (Dec) ?L,0 (ASCII) 0 (Null)



L,1



L,0

AtlasScientific™
Environmental Robotics

Find

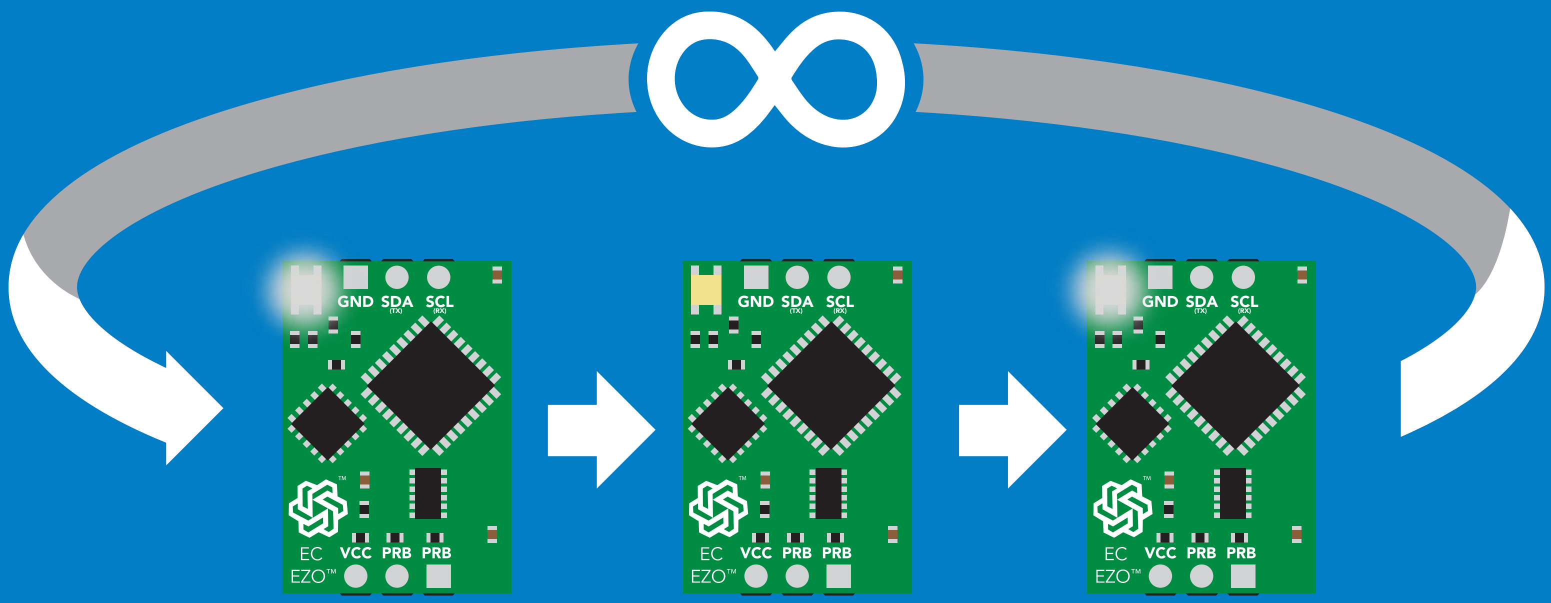
300ms processing delay

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



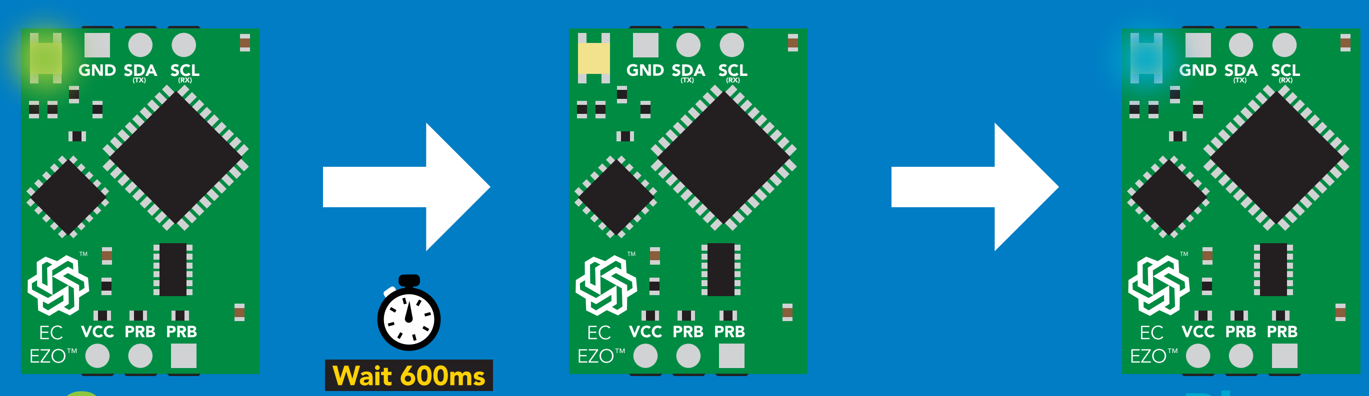
Taking reading

Command syntax

600ms processing delay

R return 1 reading

Example	Response
R	Wait 600ms · 1 (Dec) · 1,413 (ASCII) · 0 (Null)



Green Taking reading — Transmitting — **Blue** Standby

Calibration

600ms processing delay

Command syntax

Dry calibration must always be done first!

Cal,dry	dry calibration
Cal,n	single point calibration, where n = any value
Cal,low,n	low end calibration, where n = any value
Cal,high,n	high end calibration, where n = any value
Cal,clear	delete calibration data
Cal,?	device calibrated?

Example	Response
Cal,dry	Wait 600ms; 1 Dec; 0 Null
Cal,84	Wait 600ms; 1 Dec; 0 Null
Cal,low,12880	Wait 600ms; 1 Dec; 0 Null
Cal,high,80000	Wait 600ms; 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 (two point) 0 Null or 1 Dec ?CAL,2 ASCII (three point) 0 Null

Two point calibration:
Step 1. "cal,dry"
Step 2. "cal,n"
Calibration complete!

Three point calibration:
Step 1 "cal,dry"
Step 2 "cal,low,n"
Step 3 "cal,high,n"
Calibration complete!

Changing the TDS (ppm) conversion factor

300ms processing delay

There are several different conversion factors used to read TDS(ppm). For some applications, it may be necessary to use a conversion factor other than the default value of 0.54

Command syntax

TDS,n	set custom conversion factor, n = any value between 0.01 – 1.00
TDS,?	conversion factor being used

Example	Response
TDS,?	Wait 300ms; 1 (Dec) ?TDS,0.54 (ASCII) 0 (Null)
R	Wait 300ms; 1 (Dec) 100,54 (ASCII: EC, TDS) 0 (Null)
TDS,0.46	Wait 300ms; 1 (Dec) 0 (Null)
R	Wait 300ms; 1 (Dec) 100,46 (ASCII: EC, TDS) 0 (Null)

Common conversion factors

NaCl	0.47 – 0.50
KCL	0.50 - 0.57
"442"	0.65 – 0.85

Formula

EC x conversion factor = TDS

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

Example	Response
Export,?	Wait 300ms — 1 (Dec) 10,120 (ASCII) 0 (Null)
Export	Wait 300ms — 1 (Dec) 59 6F 75 20 61 72 (ASCII) 0 (Null) (1 of 10)
Export	Wait 300ms — 1 (Dec) 65 20 61 20 63 6F (ASCII) 0 (Null) (2 of 10)
(7 more)	⋮
Export	Wait 300ms — 1 (Dec) 6F 6C 20 67 75 79 (ASCII) 0 (Null) (10 of 10)
Export	Wait 300ms — 1 (Dec) *DONE (ASCII) 0 (Null)

Response breakdown

10, 120

10 = # of strings to export
120 = # of bytes to export

Export strings can be up to 12 characters long

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

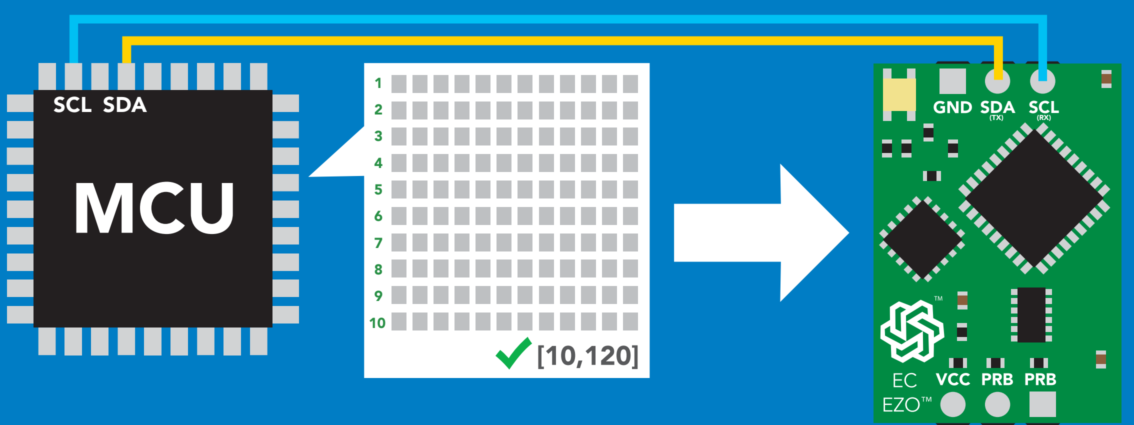
Wait 300ms 1 Dec 0 Null

Wait 300ms 1 Dec 0 Null

⋮

Wait 300ms 1 Dec 0 Null

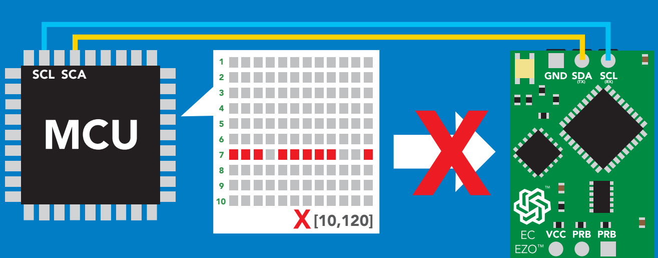
Import,n



1 *Pending 0

Dec ASCII Null

system will reboot



reboot

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

AtlasScientific Environmental Robotics

Setting the probe type

Command syntax

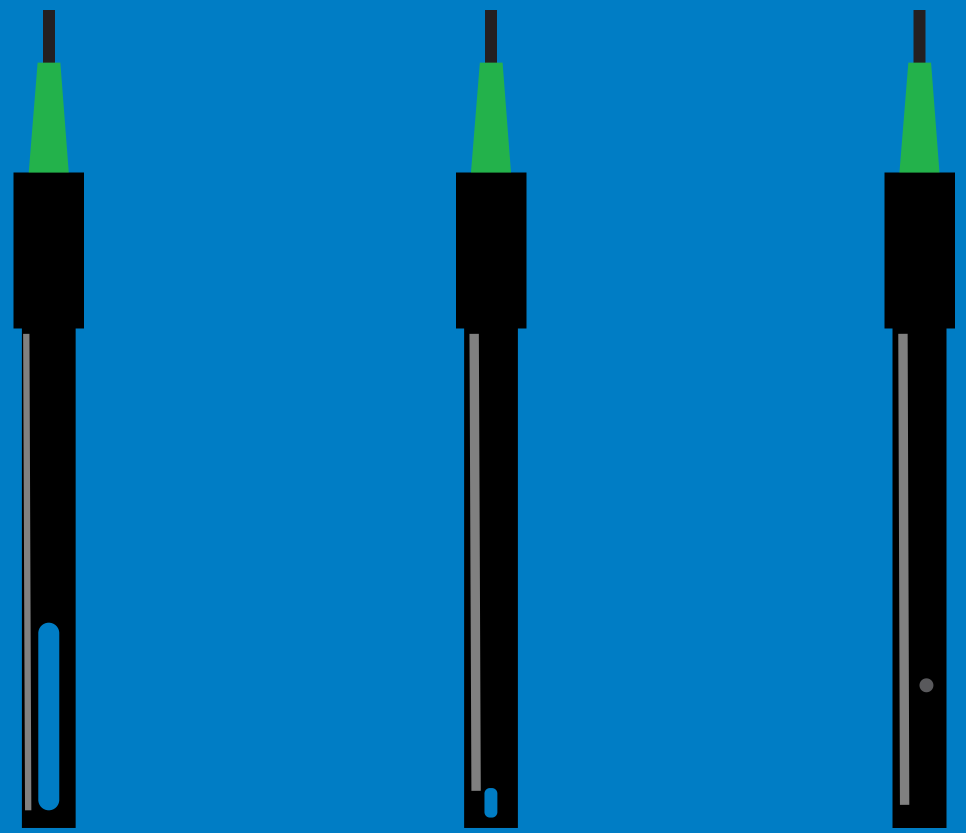
300ms processing delay

K,n n = any value; floating point in ASCII

K,? probe K value?

K 1.0 is the default value

Example	Response
K,10	Wait 300ms — 1 (Dec) 0 (Null)
K,?	Wait 600ms — 1 (Dec) K,10 (ASCII) 0 (Null)



K 0.1 K 1.0 K 10

AtlasScientific™ Environmental Robotics

Temperature compensation

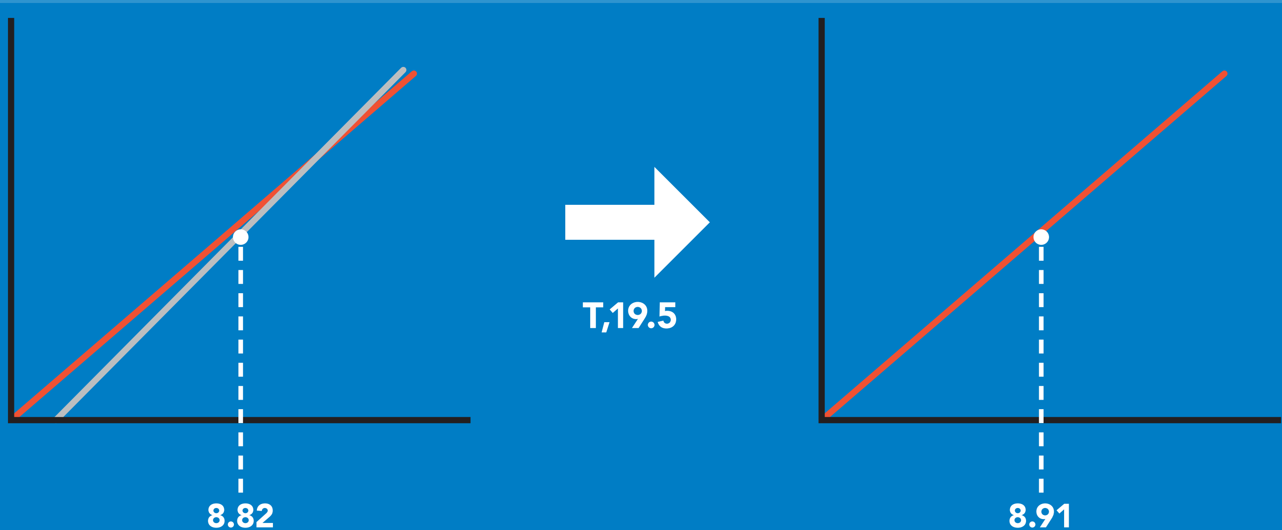
Command syntax

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
T,?	compensated temperature value?
RT,n	set temperature compensation and take a reading

300ms processing delay

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)



AtlasScientific Environmental Robotics

Enable/disable parameters from output string

Command syntax

300ms processing delay

O, [parameter],[1,0]	enable or disable output parameter
O,?	enabled parameter?

Example	Response	
O,EC,1 / O,EC,0	Wait 300ms · 1 Dec · 0 Null	enable / disable conductivity
O,TDS,1 / O,TDS,0	Wait 300ms · 1 Dec · 0 Null	enable / disable total dissolved solids
O,S,1 / O,S,0	Wait 300ms · 1 Dec · 0 Null	enable / disable salinity
O,SG,1 / O,SG,0	Wait 300ms · 1 Dec · 0 Null	enable / disable specific gravity
O,?	Wait 300ms · 1 Dec · ?,O,EC,TDS,S,SG ASCII · 0 Null	if all are enabled

Parameters

EC	Conductivity = μS/cm
TDS	Total dissolved solids = ppm
S	Salinity = PSU (ppt) 0.00 – 42.00
SG	Specific gravity (sea water only) = 1.00 – 1.300

* If you disable all possible data types your readings will display "no output".

Followed by 1 or 0

1	enabled
0	disabled

AtlasScientific™ Environmental Robotics

Naming device

300ms processing delay

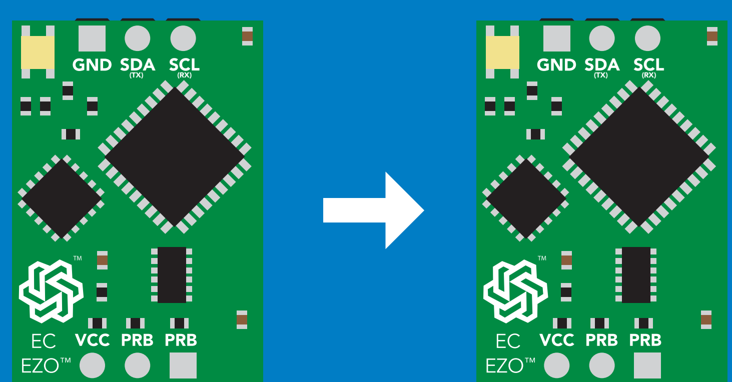
Command syntax

Do not use spaces in the name

Name,n	set name	n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ (1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16)
Name,	clears name	Up to 16 ASCII characters
Name,?	show name	

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 → Name,?



1 0 → 1 ?Name,zzt 0

Atlas Scientific™ Environmental Robotics

Device information

Command syntax

300ms processing delay

i	device information

Example

i

Response

Wait 300ms

1	?i,EC, 2.16	0
Dec	ASCII	Null

Response breakdown

?i,	EC,	2.16
	Device	Firmware

Reading device status

Command syntax

300ms processing delay

Status	voltage at Vcc pin and reason for last restart

Example	Response
Status	Wait 300ms · 1 (Dec) · ?Status,P,5.038 (ASCII) · 0 (Null)

Response breakdown

?Status,	P,	5.038
	Reason for restart	Voltage at Vcc

Restart codes

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

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	18.14 mA	0.7 mA
3.3V	16.85 mA	0.4 mA

Standby

Sleep

Protocol lock

Command syntax

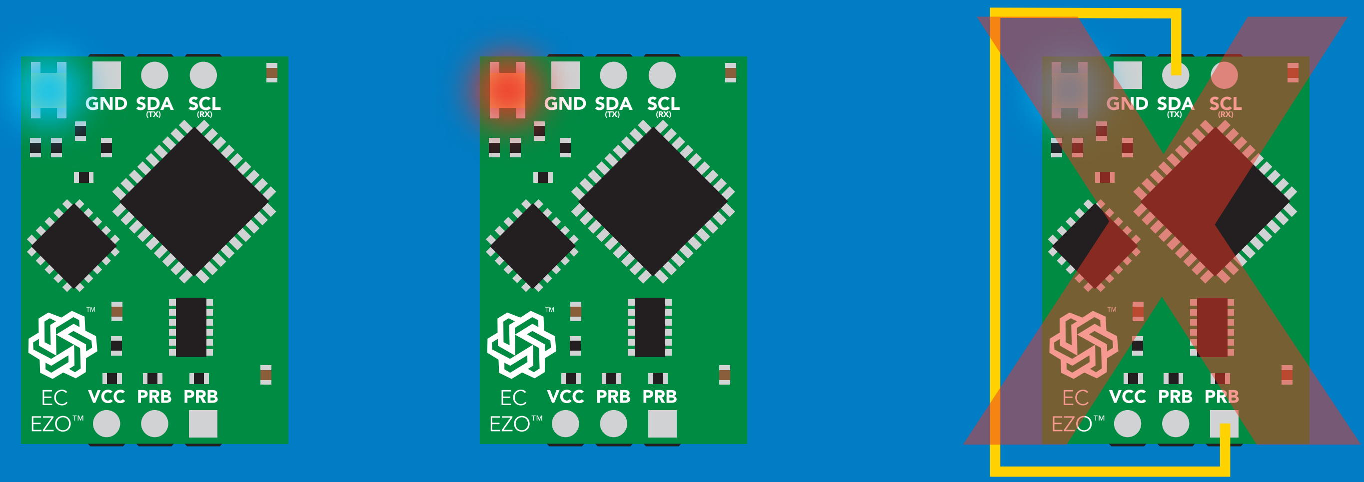
300ms processing delay

Plock,1	enable Plock	
Plock,0	disable Plock	default
Plock,?	Plock on/off?	

Locks device to I²C mode.

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**

AtlasScientific Environmental Robotics

I²C address change

Command syntax

300ms processing delay

I2C,n sets I²C address and reboots into I²C mode

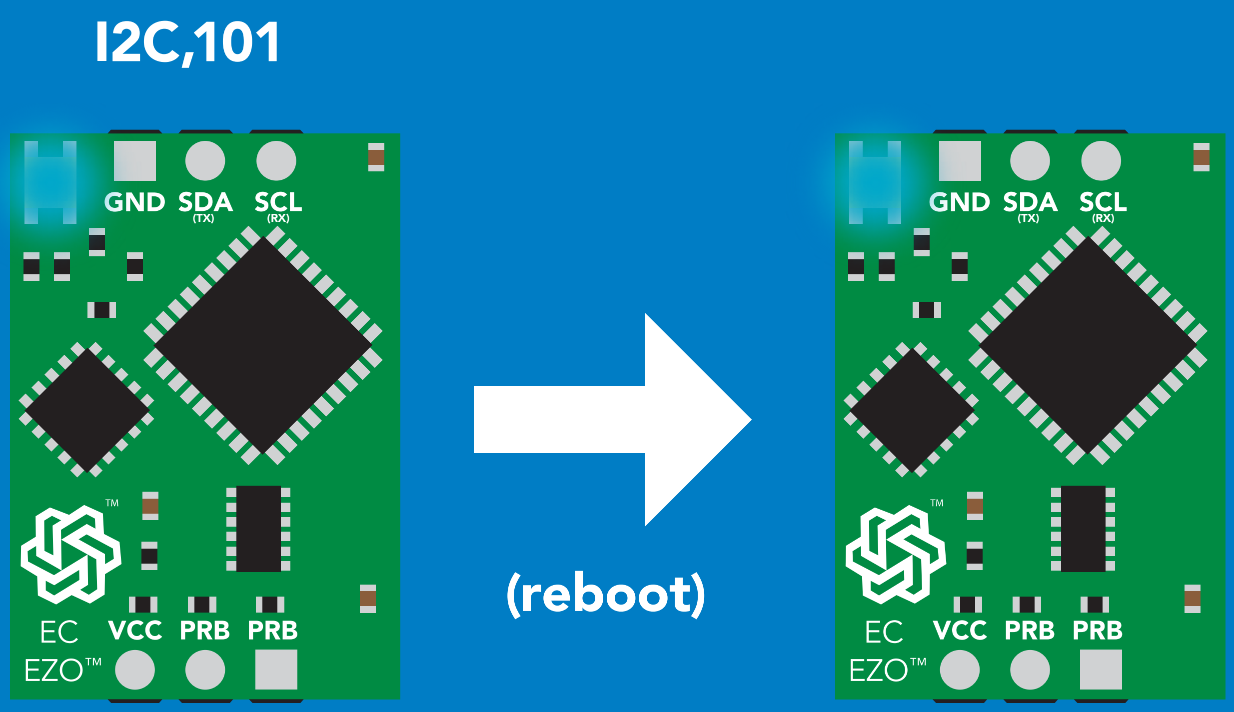
Example	Response
I2C,101	device reboot (no response given)

Warning!

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

Default I²C address is 100 (0x64).

n = any number 1 – 127



AtlasScientific Environmental Robotics

Factory reset

Command syntax

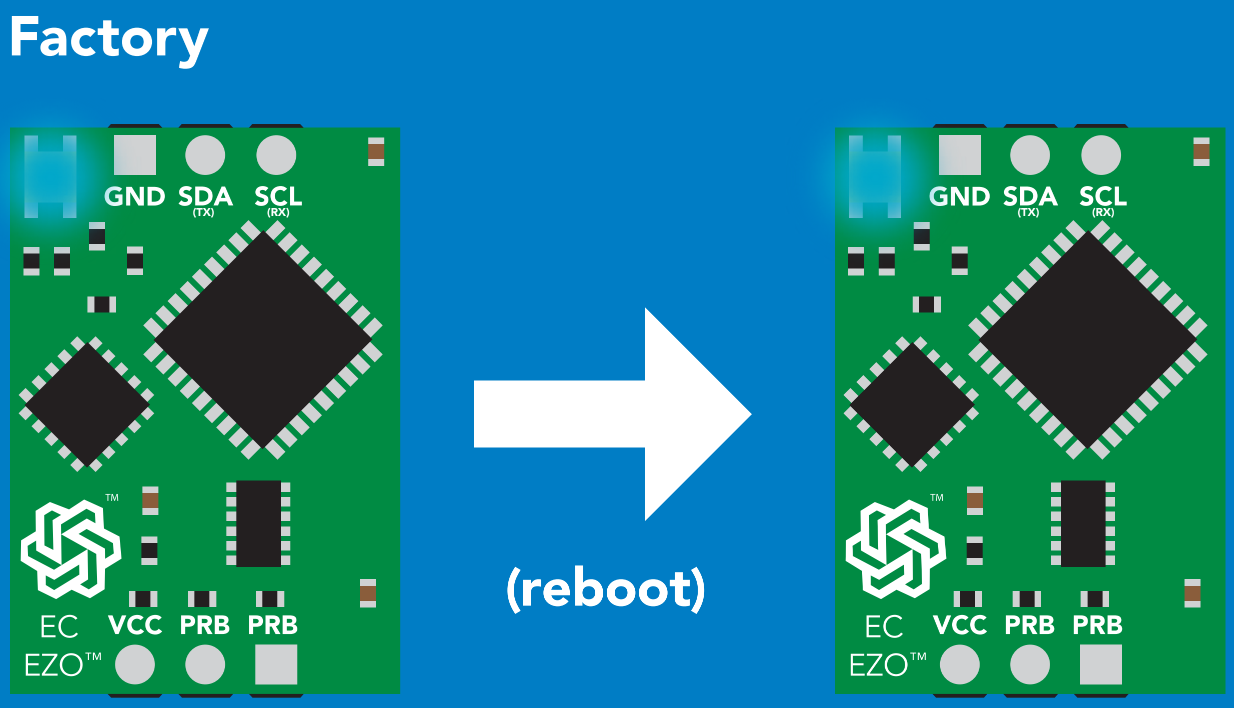
Factory reset will not take the device out of I²C mode.

Factory	**enable factory reset**

I²C address will not change

Example	Response
Factory	device reboot (no response given)

Clears calibration
LED on
Response codes enabled



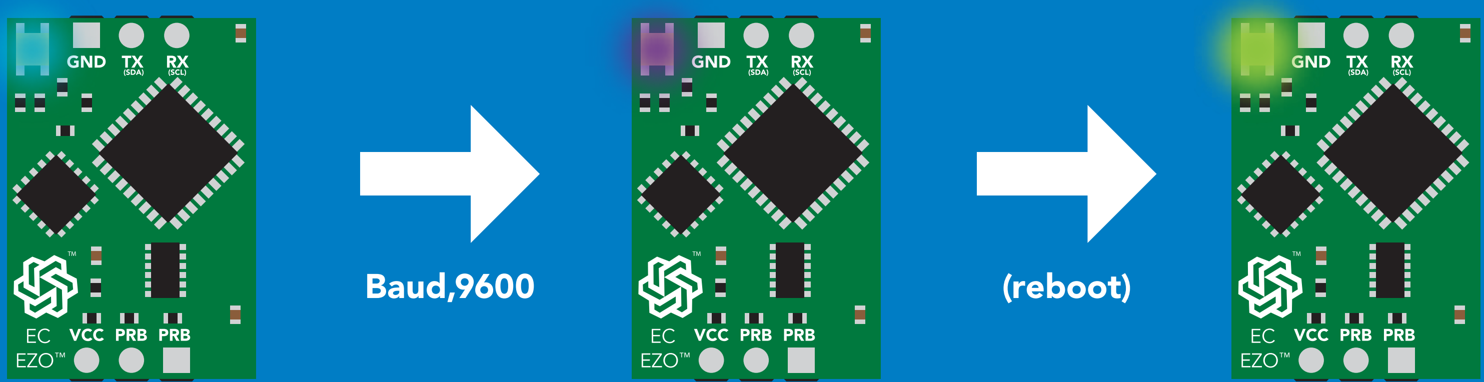
Change to UART mode

Command syntax

Baud,n switch from I^{2}C to UART

Example	Response
Baud,9600	reboot in UART mode (no response given)

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



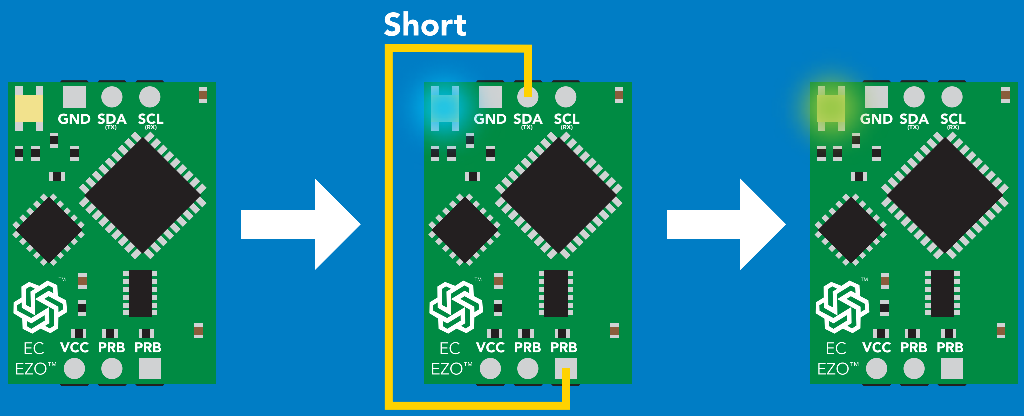
Changing to UART mode

Atlas Scientific™
Environmental Robotics

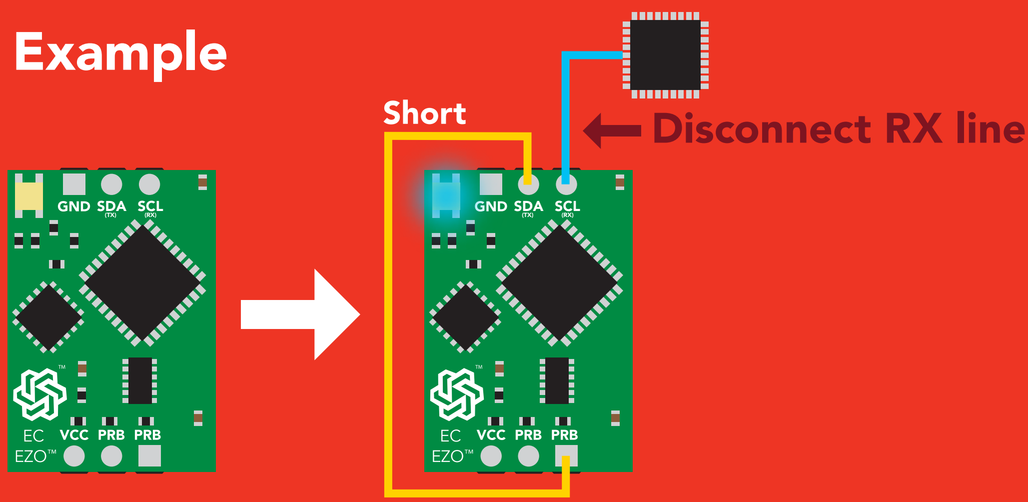
Manual switching to UART

- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to the right PRB
- 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

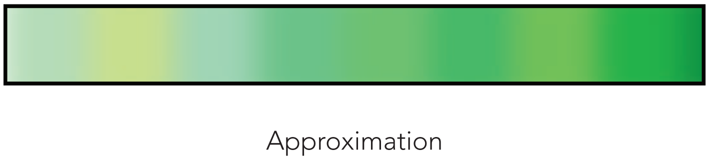
The accuracy of your readings is directly related to the quality of your calibration. (Calibration is not difficult, and a little bit of care goes a long way)

A properly calibrated conductivity probe will never need recalibration. Once calibrated, you can use the probe continuously year after year without concern. This is because a conductivity probe does not contain any parts that wear out over time.

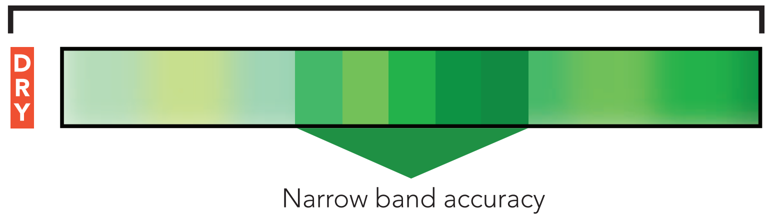
However, changing the cable length of the probe or moving the EZO-EC circuit from one machine to another may require recalibration. This is because such actions will change the electrical properties of the probe or EC circuit.

Two point or Three point calibration

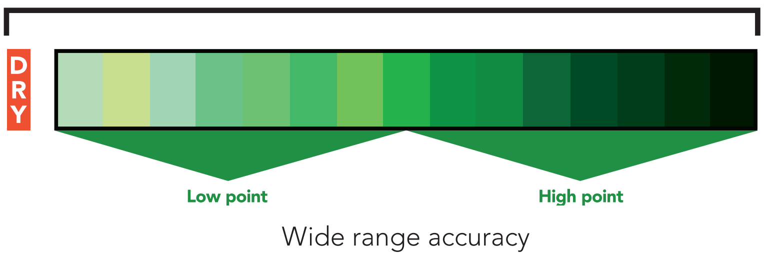
No calibration



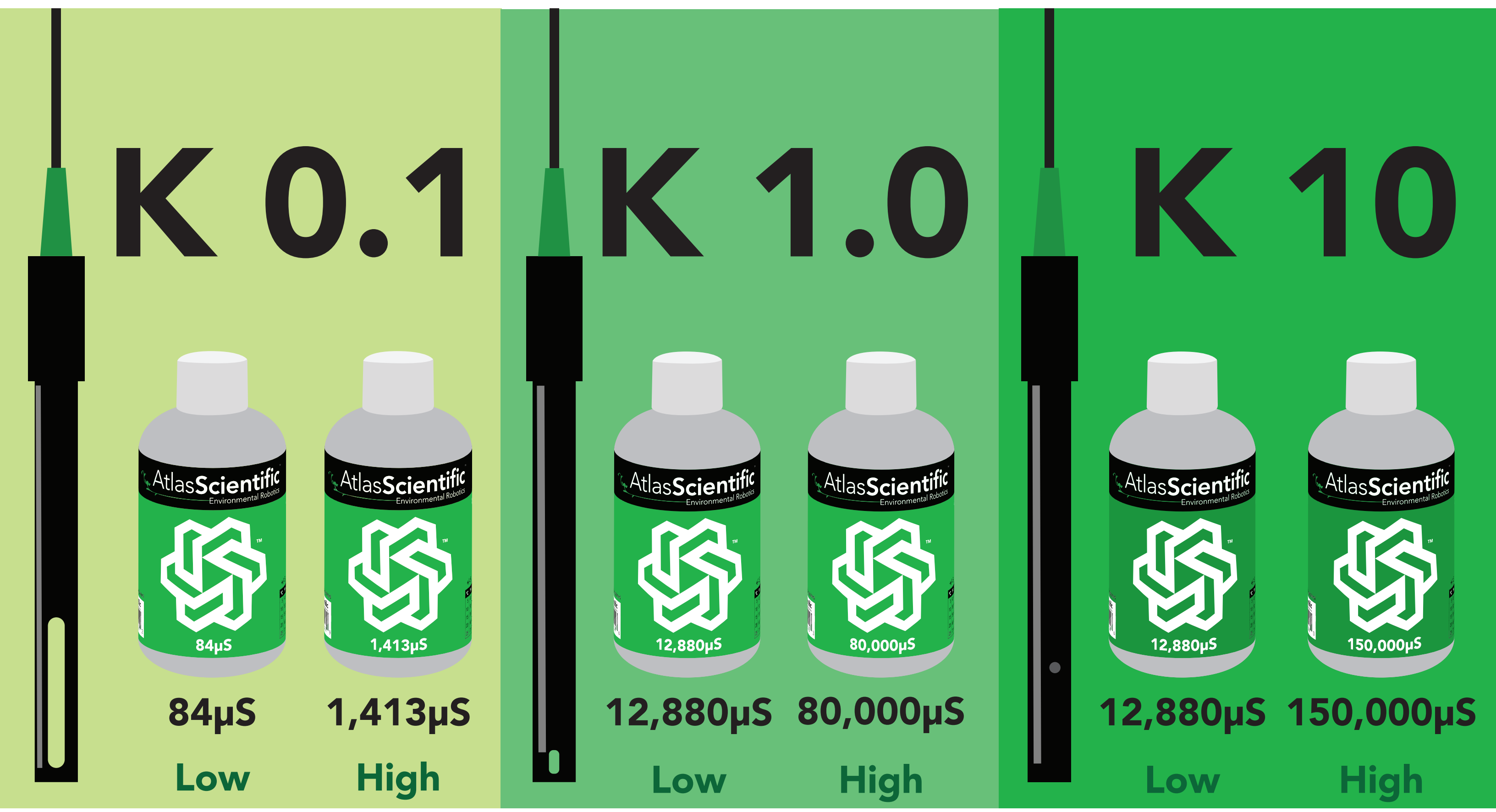
Two point calibration



Three point calibration



Recommended calibration points



When calibrating, Atlas Scientific recommends using the above µS values. However, you can use any µS values you want.

Temperature compensation during calibration

Temperature has a significant effect on conductivity readings. The EZO™ Conductivity circuit has its temperature compensation set to 25° C as the default. ***At no point should you change the default temperature compensation during calibration.***

If the solution is +/- 5° C *(or more)*, refer to the chart on the bottle, and calibrate to that value.

+/- 5 25°C / 77°F

°C	°F	µS/cm	°C	°F	µS/cm
5	41	8,220	30	86	14,120
10	50	9,330	35	95	15,550
15	59	10,480	40	104	16,880
20	68	11,670	45	113	18,210
25	77	12,880	50	122	19,550

Temperature compensation example

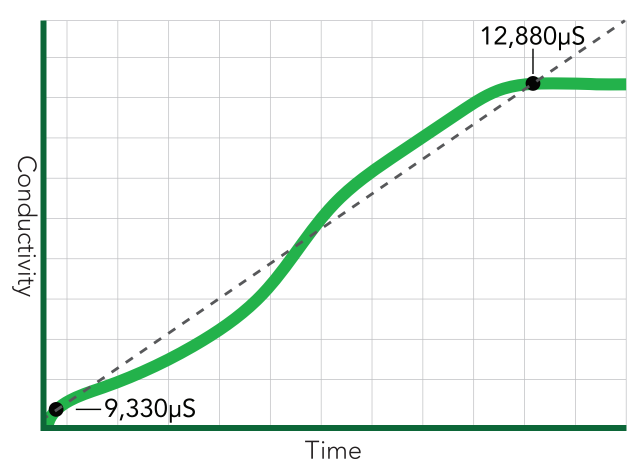
For this example, we brought the temperature of the solution down to 10° C. Referring to chart on the bottle, you can see the value you should calibrate to is **9,330µS**.

+/- 5 25°C / 77°F

°C	°F	µS/cm	°C	°F	µS/cm
5	41	8,220	30	86	14,120
10	50	9,330	35	95	15,550
15	59	10,480	40	104	16,880
20	68	11,670	45	113	18,210
25	77	12,880	50	122	19,550

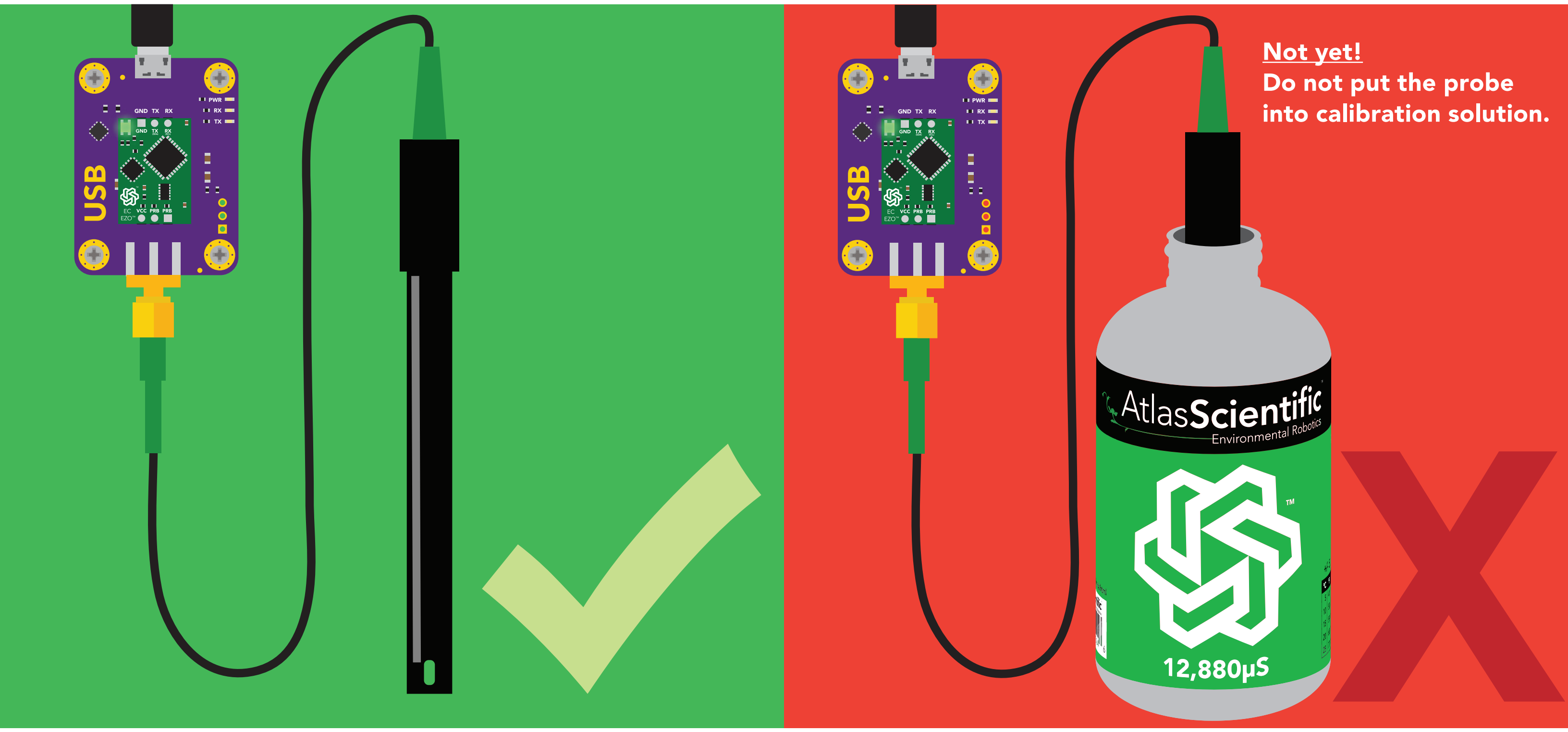
Over time, the readings will normalize as the solution warms to 25° C.

See pages **27** or **54** for more information.



1. Pre-calibration setup

Connect the dry conductivity probe and take continous readings.



2. Set probe type

If your probe ≠ K 1.0 *(default)*, then set the probe type by using the **"K,n"** command. *(where n = K value of your probe)* for more information, see page **26** or **53**.

3. Dry calibration

Perform a dry calibration using the command **"Cal,dry"** Even though you may see readings of 0.00 before issuing the **"Cal,dry"** command, it is still a necessary part of calibration.

00.00 → **"Cal,dry"** → 0.00 ✔ **Correct**

17.00 → **"Cal,dry"** → 0.00 ✔ **Also correct**

AtlasScientific™
Environmental Robotics

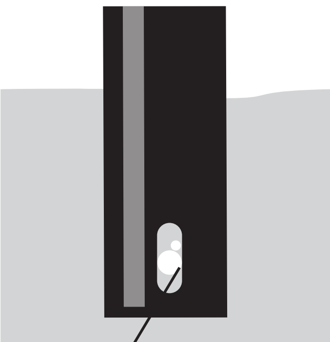
4. Calibration

Atlas Scientific recommends performing a three point calibration (*dry, low point & high point)* to obtain the greatest sensing range possible. However, depending on your situation a two point calibration may suffice.

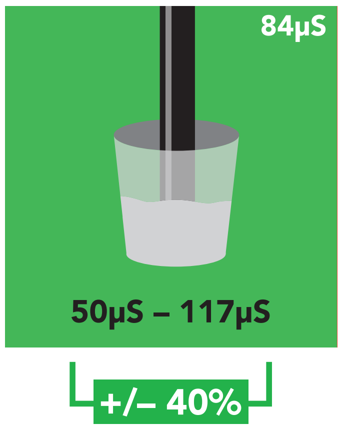
To perform a two or three point calibration, follow the instructions below.

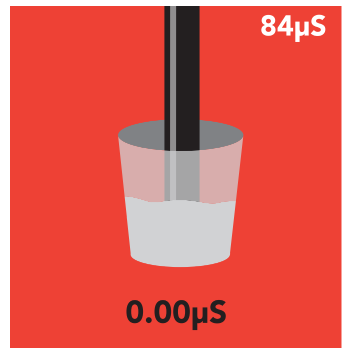
Two point calibration

After completing the dry calibration; Pour a small amount of calibration solution into a cup *(µS value of your choice)*. Shake the probe to make sure you do not have trapped air in the probe. You should see readings that are off by +/- 40% from the stated value of the calibration solution. Wait for readings to stabilize *(small movement from one reading to the next is normal)*.

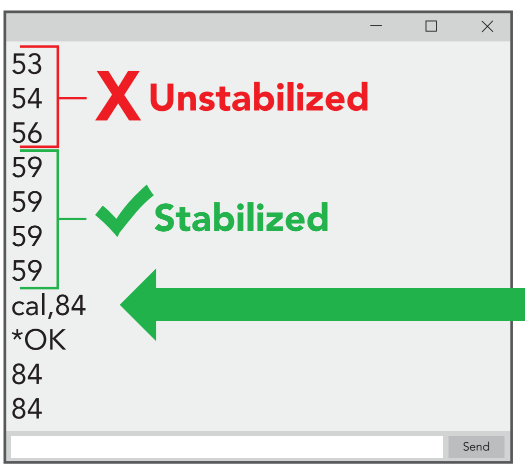


Trapped air in sensing area (shake to remove)





check probe connection, you cannot calibrate to 0.

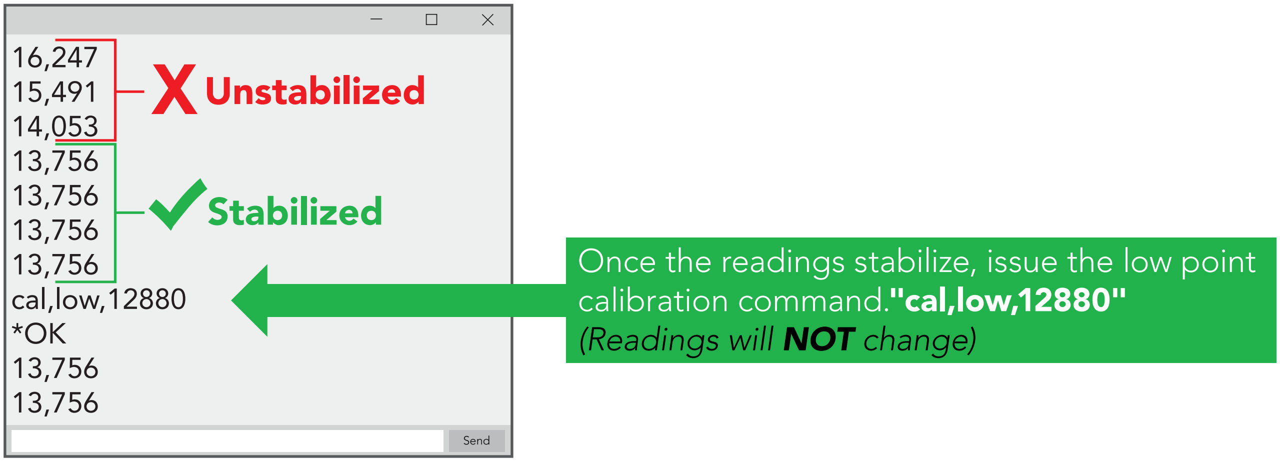


Once the readings stabilize, issue the single point calibration command.**"cal,n"** where n = any value. *(Readings **will** change, calibration complete).a*

Calibration complete!

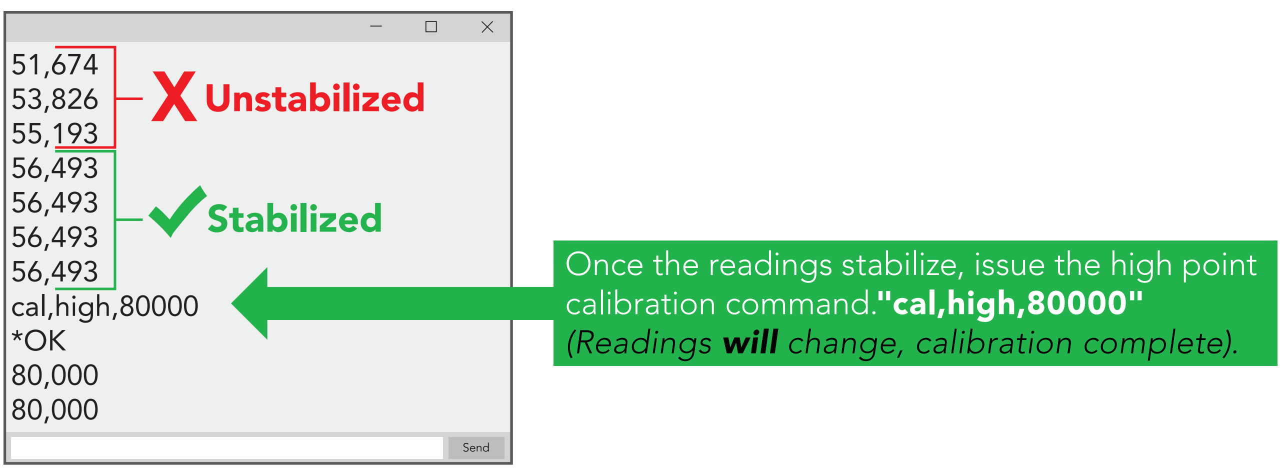
Three point calibration - low point

- Complete the dry calibration process first.
- Pour a small amount of the low point calibration solution into a cup.
- Shake the probe to remove trapped air.
- Readings may be off by +/- 40%
- Wait for readings to stabilize.



Three point calibration - high point

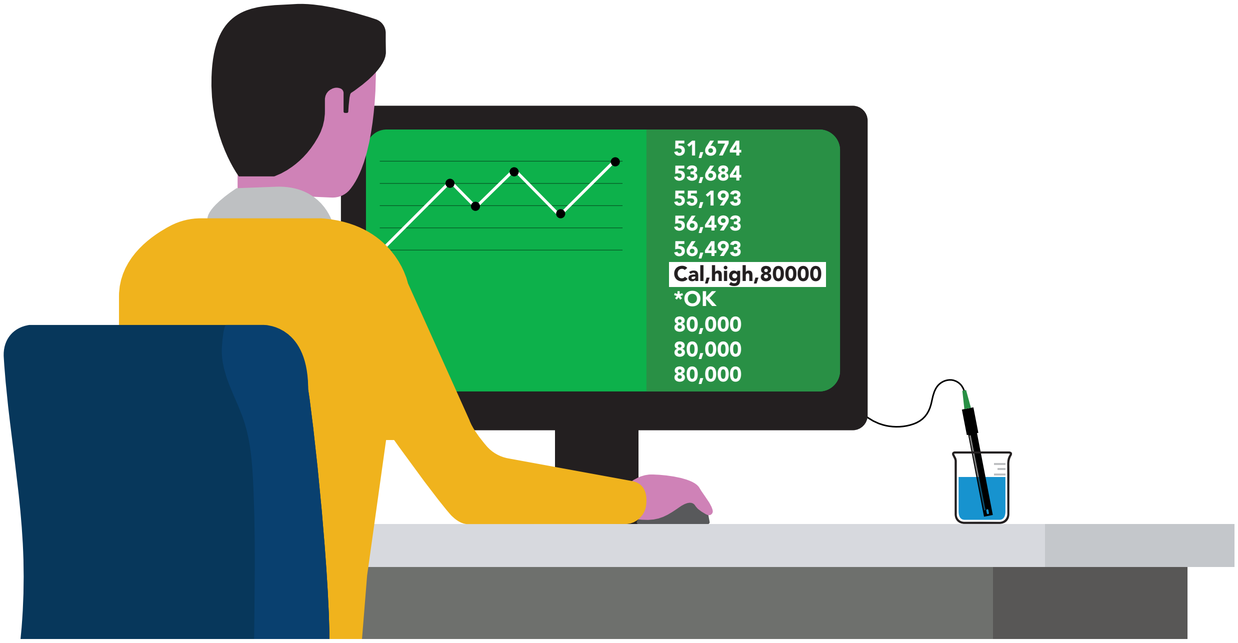
- Rinse off the probe before calibrating to the high point.
- Pour a small amount of the high point calibration solution into a cup.
- Shake the probe to remove trapped air.
- Readings may be off by +/- 40%
- Wait for readings to stabilize.



Calibration complete!

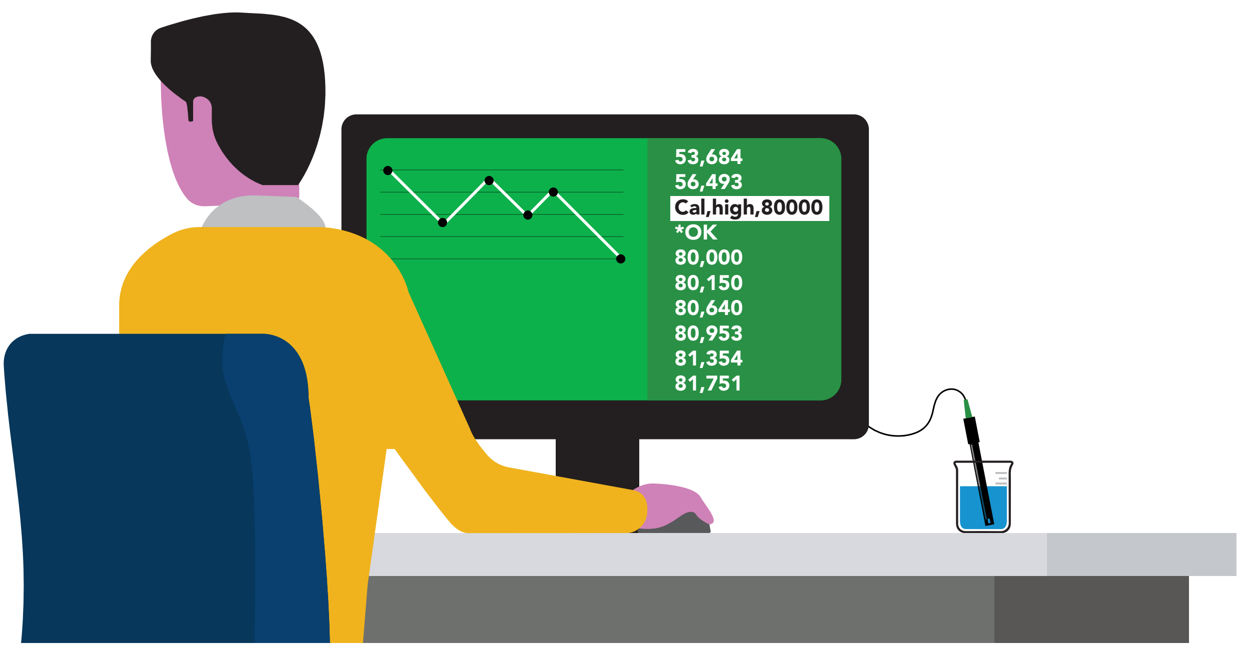
Best practices for calibration

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



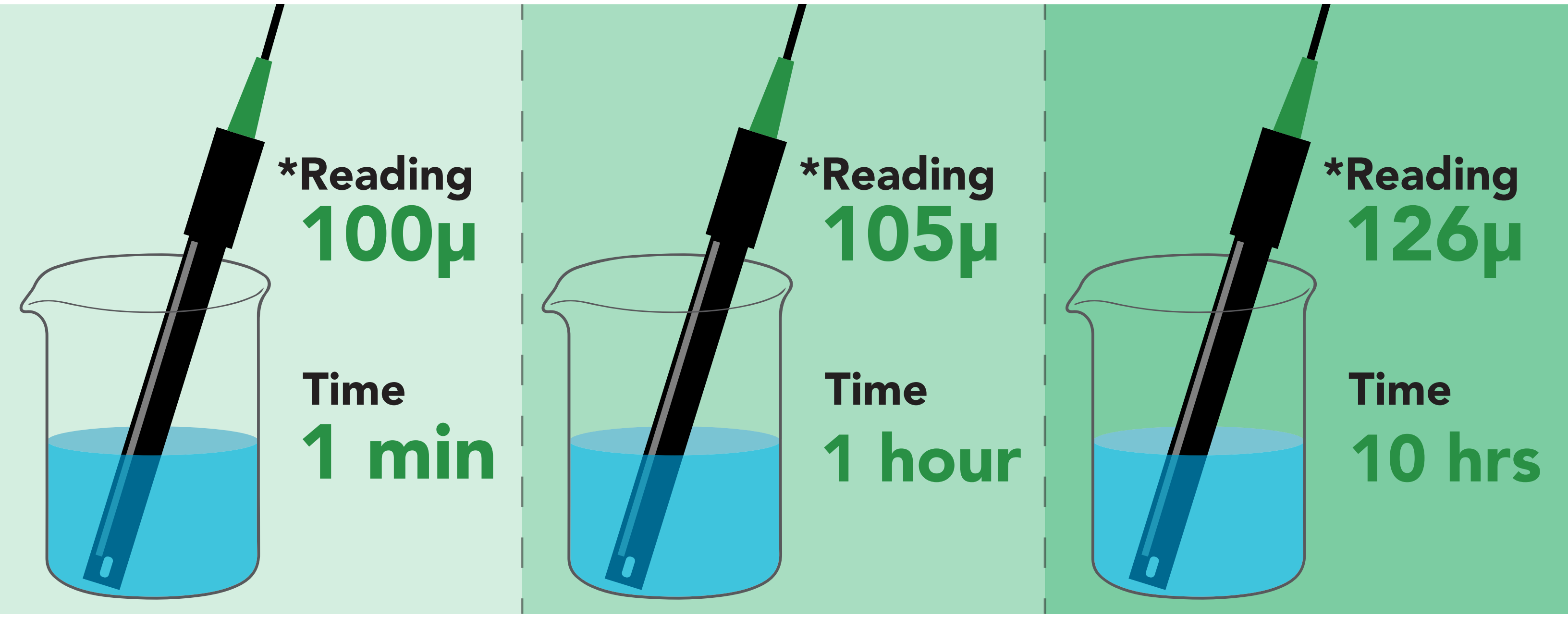
⚠ Never do a blind calibration! ⚠

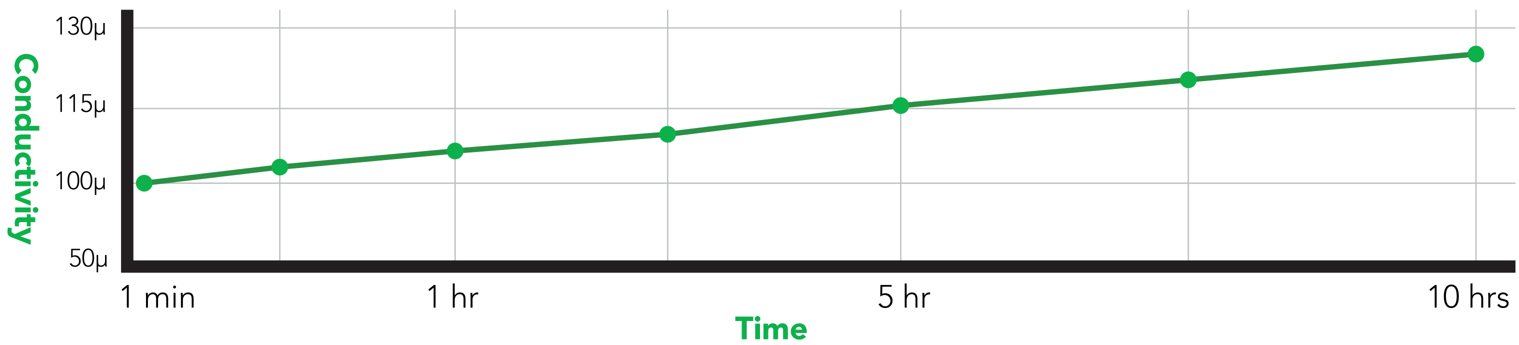
Issuing a calibration command before the readings stabilize will result in drifting readings.



Long-term conductivity measurements in stagnant water

Taking continuous conductivity readings in stagnant water:





A small amount of energy must be put into the water to measure conductivity. This small amount of energy will start to affect the readings in stagnant water. Over time, the energy passing through the stagnant water will start to align the dissolved salts along a path of least resistance. Lowering the resistance of the water will increase the water's conductivity.

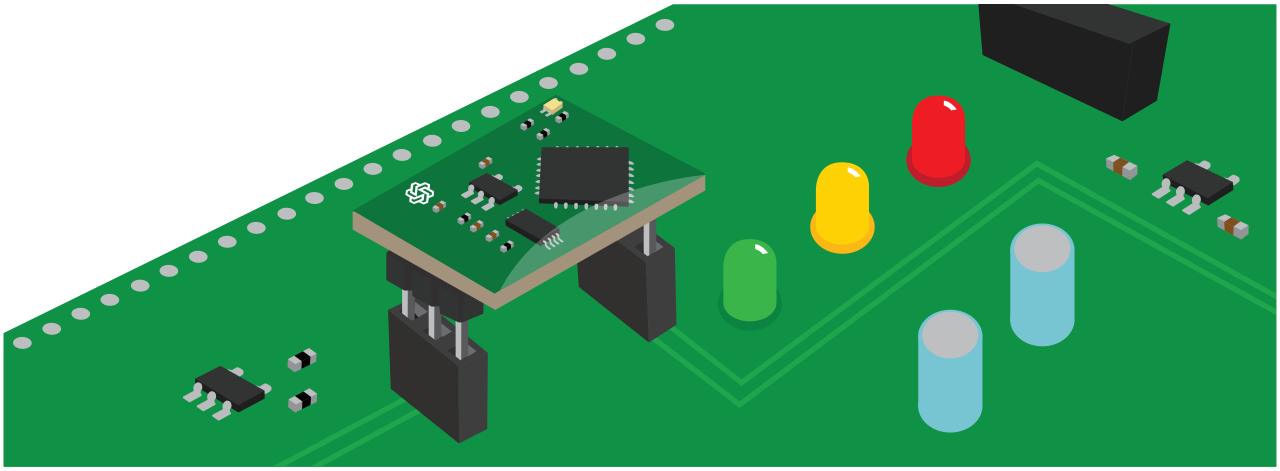
Moving the probe or the water will disrupt this alignment and cause the readings to suddenly return to normal.

These are example readings; there is no way to predict how the readings will change over time.

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.

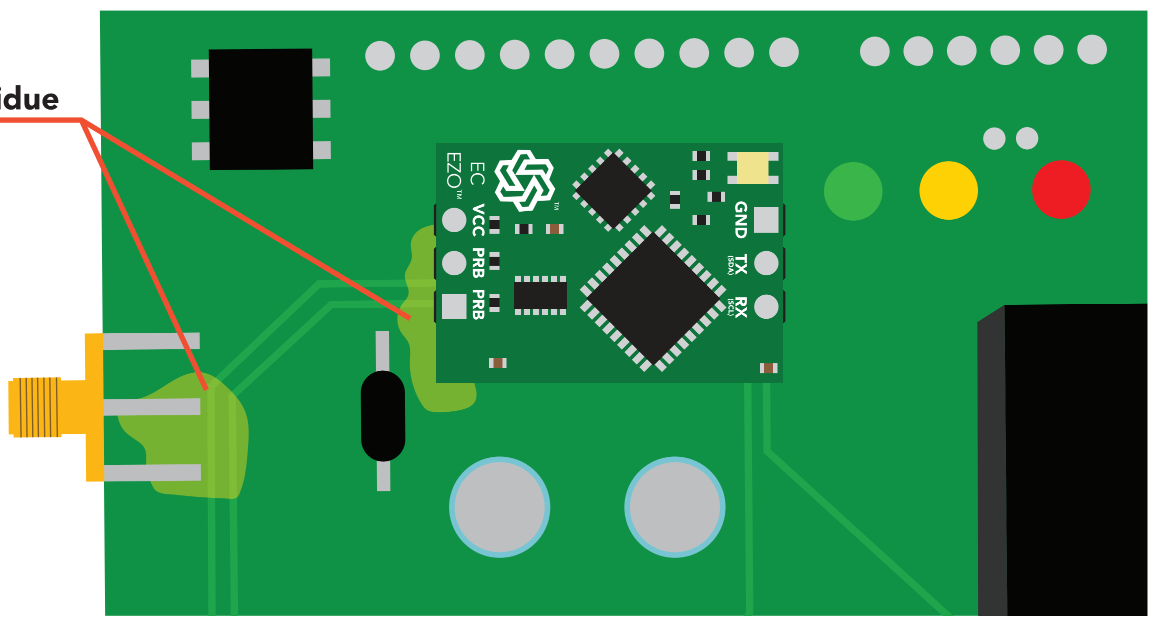
Instead, solder female header pins to your PCB and place the EZO device in the female headers.



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".

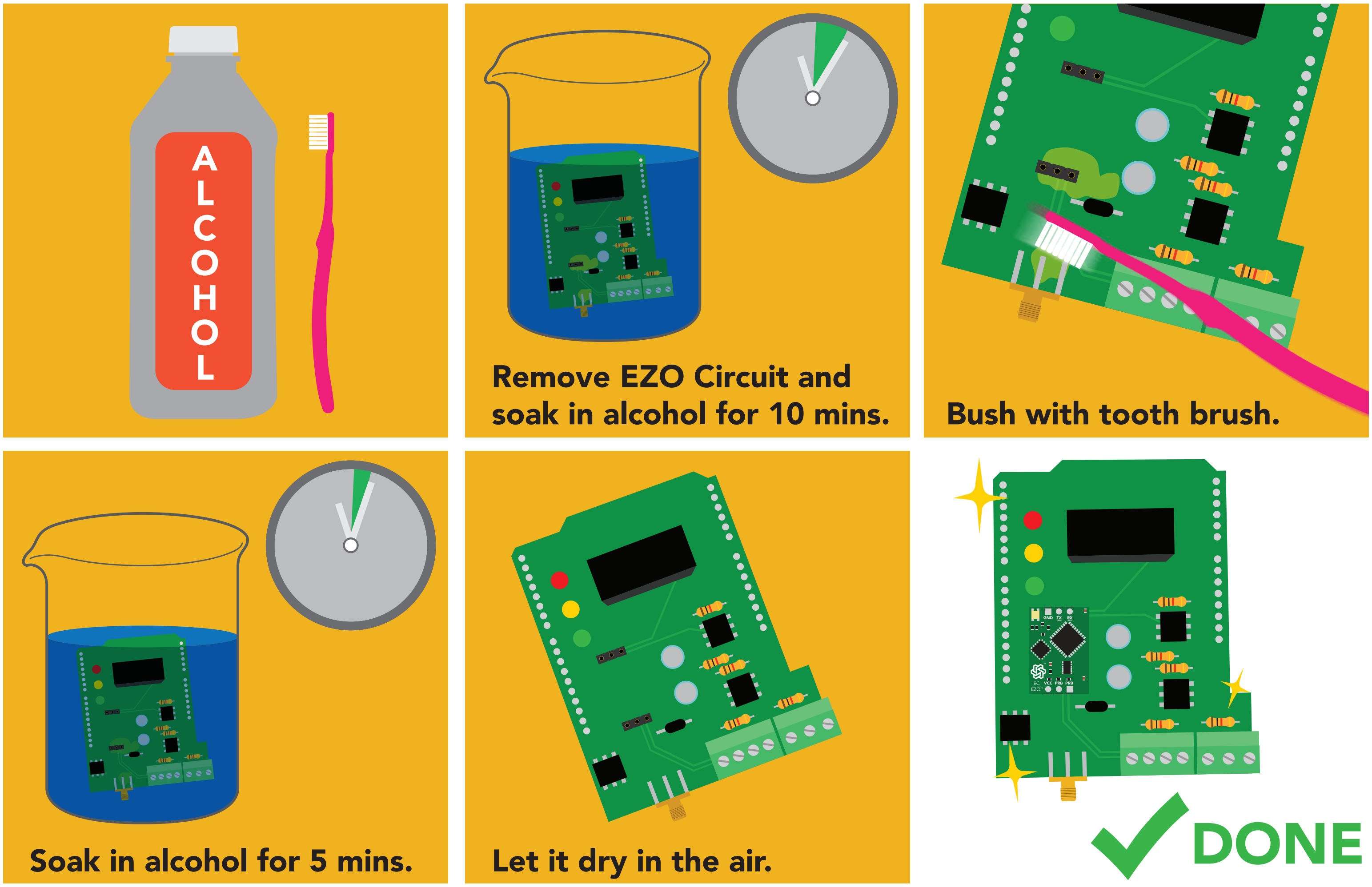
Flux residue



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

Soldering

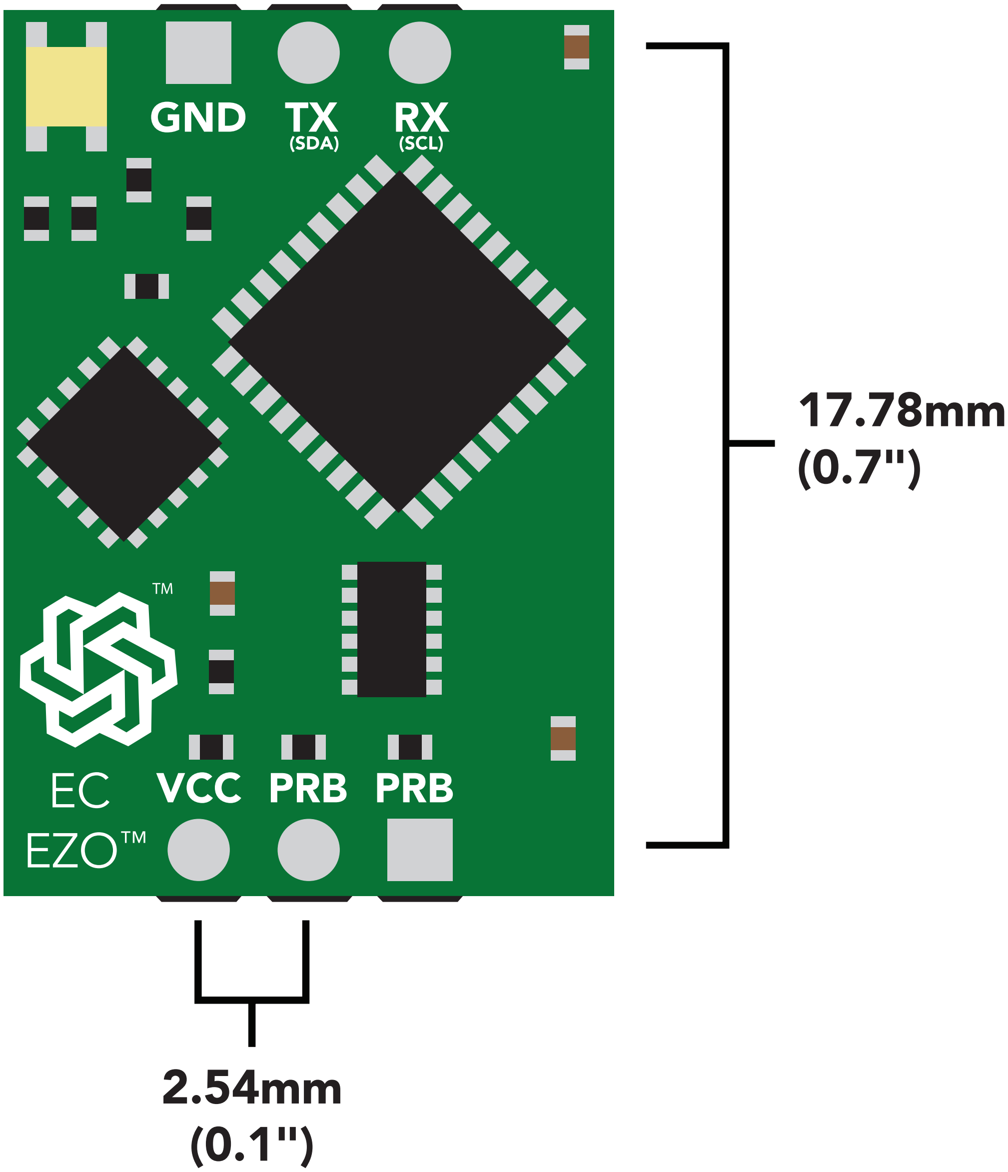
Removing flux residue can be done with commercially available products such as flux off or you can use alcohol and a tooth brush.



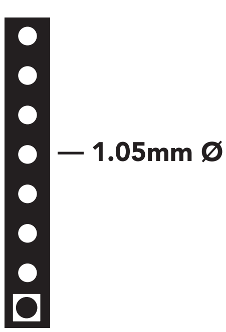
What does a flux short look like?

Readings move slowly and take serval minutes to reach the correct value.

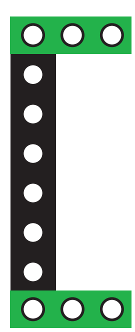
EZO™ circuit footprint



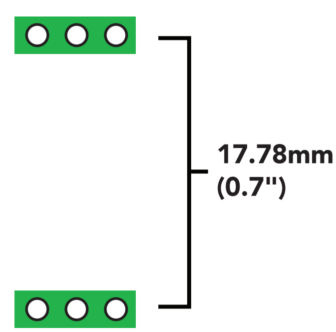
1 In your CAD software, place a 8 position header.



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 6.5

Revised calibration theroy on pages 65 - 70.

Datasheet V 6.4

Revised entire document.

Datasheet V 6.3

Revised naming device info on pages 36 & 63.

Datasheet V 6.2

Added new command:
"TDS,n" Changing the TDS (ppm) conversion factor on pages 30 (UART) & 57 (I^2C).

Datasheet V 6.1

Corrected typos within the datasheet.

Datasheet V 6.0

Changed the K value range from 0.1 to 0.01 on pg 5.

Datasheet V 5.9

Moved Default state to pg 17.

Datasheet V 5.8

Revised conductivity probe range information on pg 5.

Datasheet V 5.7

Revised response for the sleep command in UART mode on pg 39.

Datasheet V 5.6

Added more information on the Export calibration and Import calibration commands.

Datasheet V 5.5

Revised calibration theory pages, added information on temperature compensation on pg. 15, moved data isolation to pg 9, and correct wiring to pg 11.

Datasheet V 5.4

Revised isolation schematic on pg. 13

Datasheet V 5.3

Added new command:
"RT,n" for Temperature compensation located on pages 30 (UART) & 55 (I^2C).
Added firmware information to Firmware update list.

Datasheet V 5.2

Revised calibration information on pages 27 & 52.

Datasheet V 5.1

Added more information about temperature compensation on pages 30 & 55.

Datasheet V 5.0

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

Datasheet V 4.9

Removed note from certain commands about firmware version.
Added steps to calibration command pages 27 (UART) and 52 (I^2C).

Datasheet V 4.8

Revised definition of response codes on pg 46.

Datasheet V 4.7

Revised cover page art.

Datasheet V 4.6

Updated calibration processing delay time on pg.52.

Datasheet V 4.5

Revised Enable/disable parameters information on pages 31 & 56.

Datasheet V 4.4

Updated High point calibration info on page 11.

Datasheet V 4.3

Updated calibration info on pages 27 (UART) and 52 (I^2C).

Datasheet V 4.2

Revised Plock pages to show default value.

Datasheet V 4.1

Corrected I^2C calibration delay on pg. 52.

Datasheet V 4.0

Revised entire datasheet.

Firmware updates

V1.0 – Initial release (April 17, 2014)

V1.1 – (June 2, 2014)
- Change specific gravity equation to return 1.0 when the uS reading is < 1000 (previously returned 0.0)
- Change accuracy of specific gravity from 2 decimal places to 3 decimal places
- Don't save temperature changes to EEPROM

V1.2 – (Aug 1, 2014)
- Baud rate change is now a long, purple blink

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

V1.6 – I2C bug (Dec 1, 2014)
- Fixed I^2C bug where the circuit may inappropriately respond when other I2C devices are connected

V1.8 – 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
  This would cause the EZO circuit to revert back to UART mode if set to I2C

V2.10 – (April 12, 2017)
- Added "Find" command.
- Added "Export/import" command.
- Modified continuous mode to be able to send readings every "n" seconds.
- Default output changed from CSV string of 4 values to just conductivity; Other values must be enabled

V2.11 – (April 28, 2017)
- Fixed "Sleep"bug, where it would draw excessive current.

V2.12 – (May 9, 2017)
- Fixed bug in sleep mode, where circuit would wake up to a different I^2C address.

V2.13 – (July 16, 2018)
- Added "RT" command to Temperature compensation

V2.14 – (Nov 26, 2019)
- The K value range has been extended to 0.01

V2.15 – (June 29, 2020)
- Fixed bug where output doesnt always round to 0

Firmware updates

V2.16 – (Dec 14, 2021)

• Internal update for new part compatibility.

Warranty

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

The debugging phase

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

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

Atlas Scientific™
Environmental Robotics

V 5.6

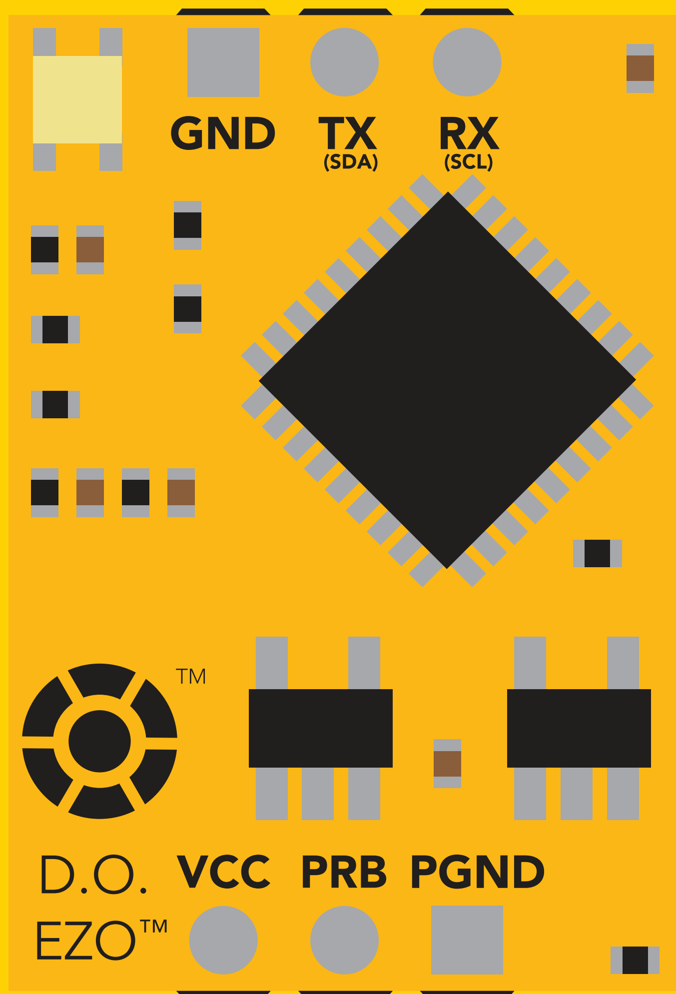
Revised 4/23

EZO-DO™

Embedded Dissolved Oxygen Circuit

ISO 5814 Compliant
(determination of dissolved oxygen)

Reads	**Dissolved Oxygen**
Range	**0.00 – 100 mg/L** **0 – 350% saturation**
Accuracy	**+/– 0.05 mg/L**
D.O. reading time	**600ms**
Supported probes	**Any galvanic probe**
Calibration	**1 or 2 point**
Temperature, salinity and pressure compensation	**Yes**
Data protocol	**UART & I²C**
Default I²C address	**97 (0x61)**
Operating voltage	**3.3V – 5V**
Data format	**ASCII**

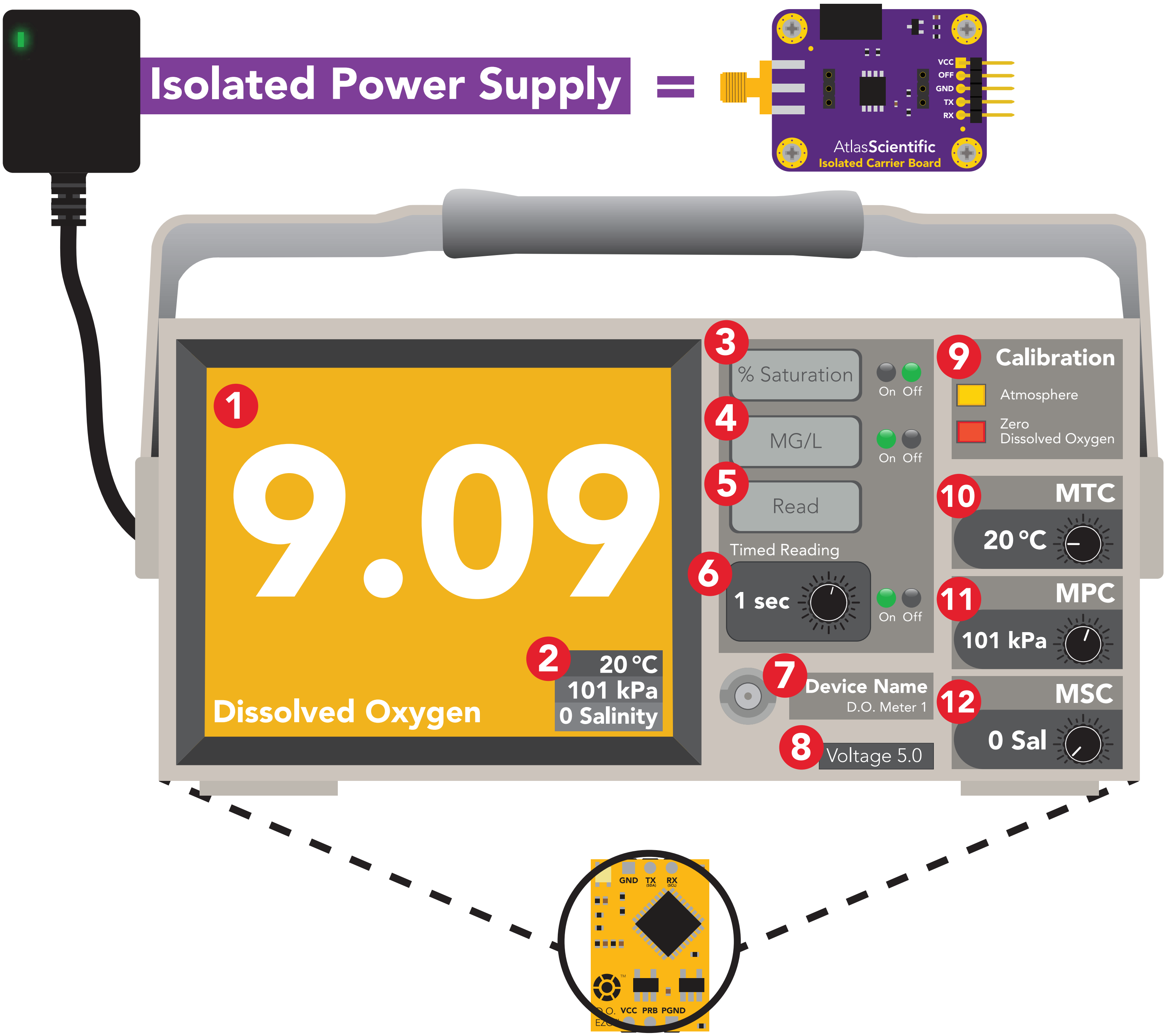


PATENT PROTECTED

Written by Jordan Press
Designed by Noah Press

This is an evolving document, check back for updates.

The EZO™ D.O. Circuit has all the features of this bench top meter.



1 Two decimal D.O. reading

2 Temperature, pressure, and salinity compensation value

3 Percent saturation

4 Milligrams per liter

5 Immediate reading

6 Timed readings

7 Set device name

8 Voltage usage

9 Multi point calibration

10 Manual temperature compensation

11 Manual pressure compensation

12 Manual salinity compensation

The EZO™ D.O. Circuit is compatible with any brand of galvanic D.O. probe.

Available data protocols

UART

Default

I^{2}C

Unavailable data protocols

SPI

Analog

RS-485

Mod Bus

4–20mA

STOP

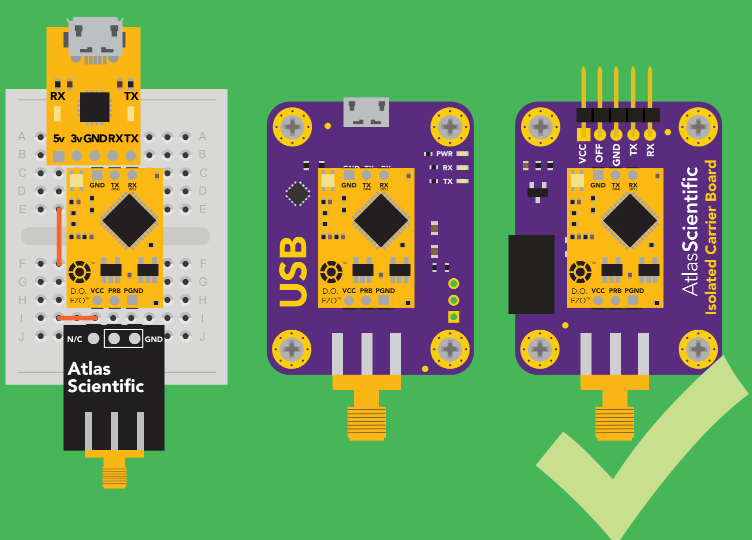
SOLDERING THIS DEVICE VOIDS YOUR WARRANTY.

Are there specific soldering instructions? **Yes, see page 71.**

Can you make a warranty claim after soldering? **No.**

If you have not used this product before; Observe how a properly working sensor behaves *BEFORE* embedding it into your PCB.

Get this device working using one of these methods first.



Do not embed before you have experience with this sensor.

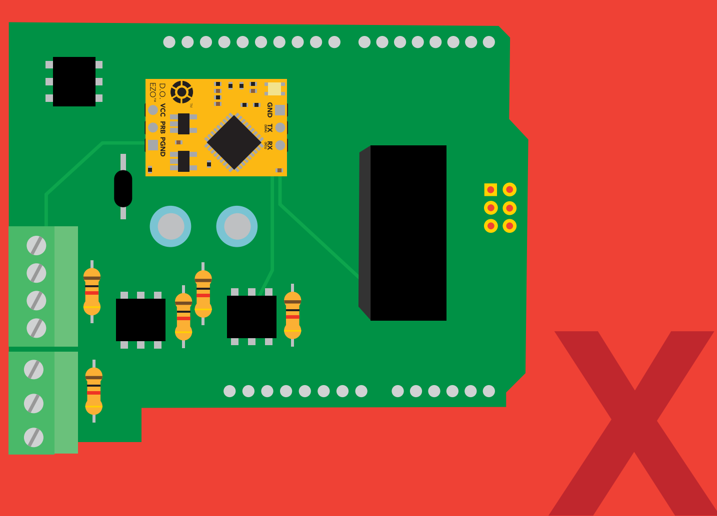
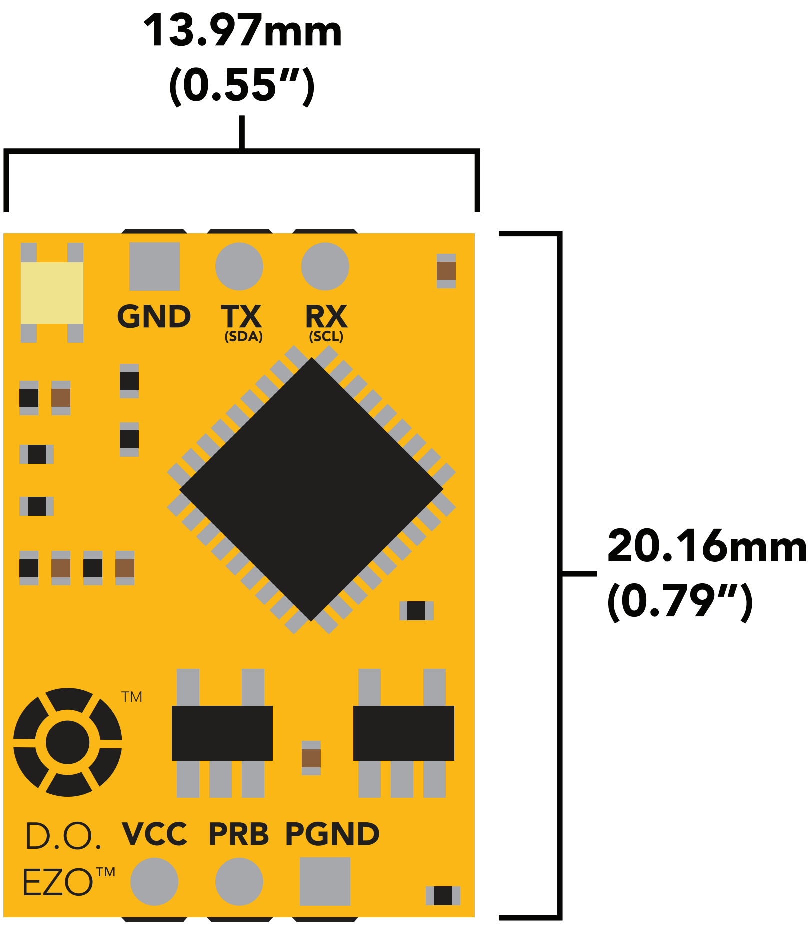


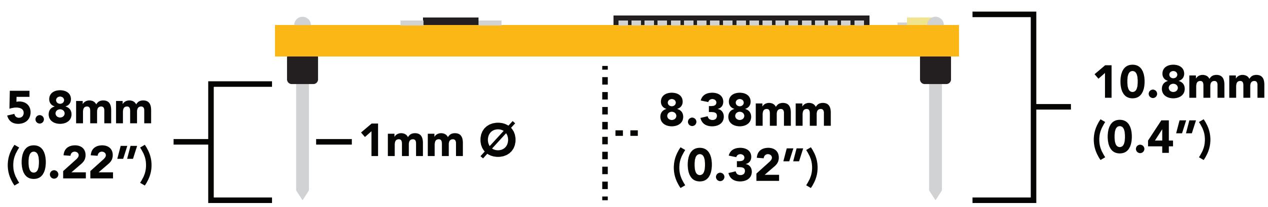
Table of contents

UART

I²C

EZO™ circuit dimensions





Power consumption

	LED	MAX	STANDBY	SLEEP
5V	ON	13.5 mA	13.1 mA	0.66 mA
	OFF	12.7 mA	12.7 mA	
3.3V	ON	12.1 mA	12 mA	0.3 mA
	OFF	11.9 mA	11.9 mA	

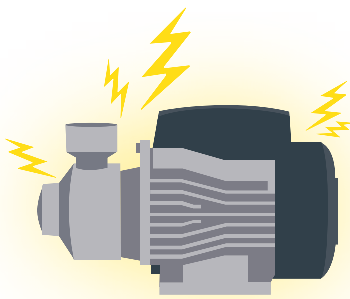
Absolute max ratings

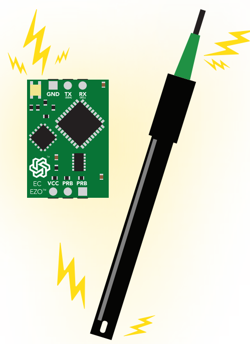
Parameter	**MIN**	**TYP**	**MAX**
Storage temperature (EZO™ D.O.)	-65 °C		125 °C
Operational temperature (EZO™ D.O.)	-40 °C	25 °C	85 °C
VCC	3.3V	5V	5.5V

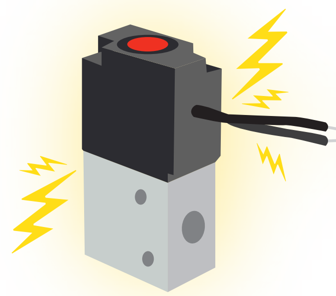
Electrical isolation

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

When electrical noise is interfering with the Dissolved Oxygen readings it is common to see rapidly fluctuating readings or readings that are consistently off. To verify that electrical noise is causing inaccurate readings, place the Dissolved Oxygen probe in a cup of water by itself. The readings should stabilize quickly, confirming that electrical noise was the issue.







Advice:

When reading D.O. along with other sensors, electrical isolation is strongly recommended.
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

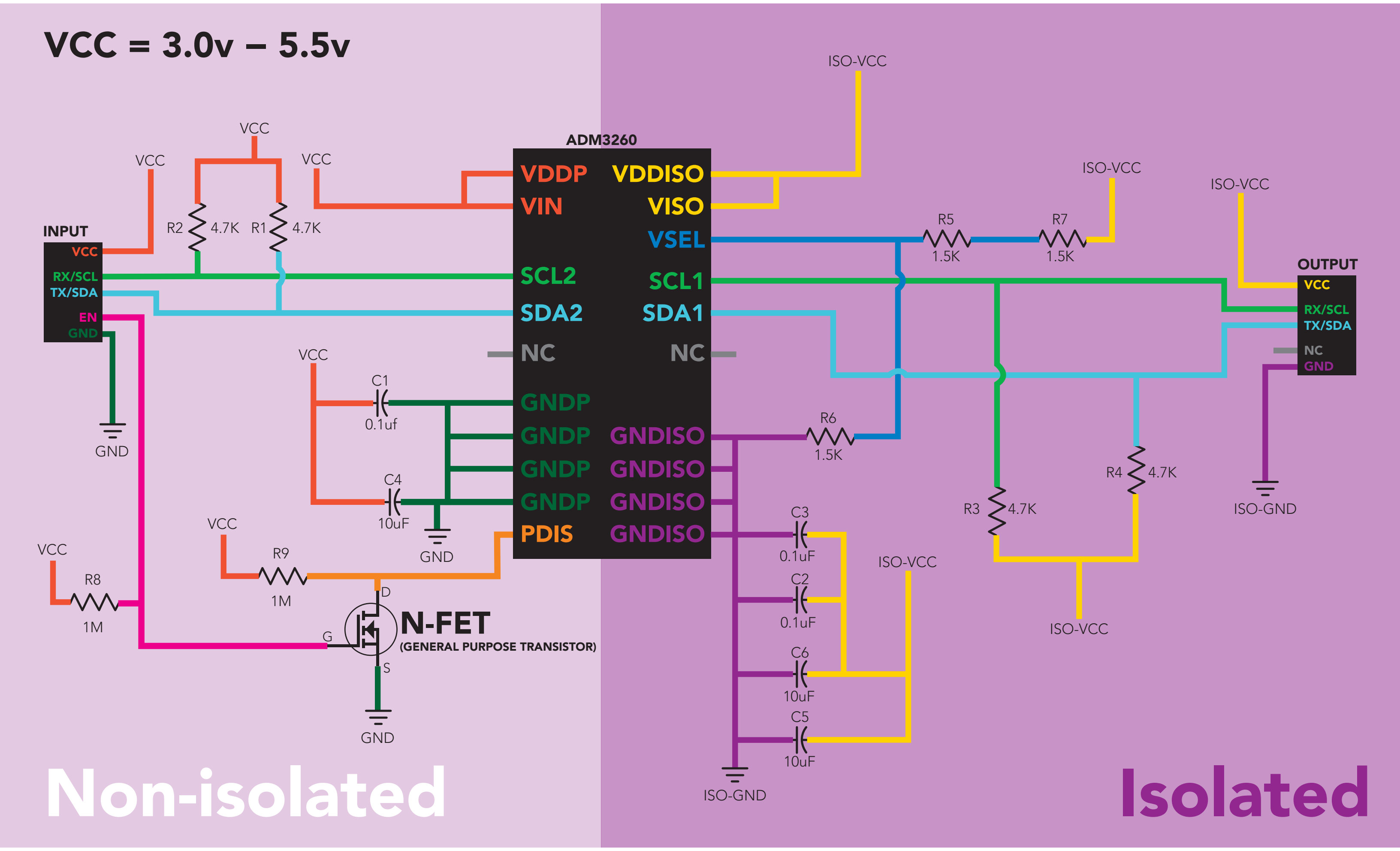
Electrically Isolated EZO™ Carrier Board (old style)

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.

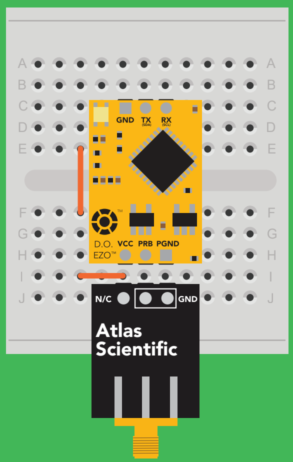
This isolation schematic is based on the ADM3260, which can output up to 150 mW of isolated power. PCB layout requires special attention for EMI/EMC and RF Control. Having good ground planes and keeping the capacitors as close to the chip as possible are crucial for proper performance.

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.

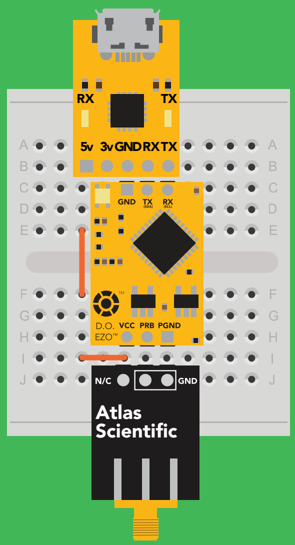
Isolated ground is different from non-isolated ground, these two lines should not be connected together.



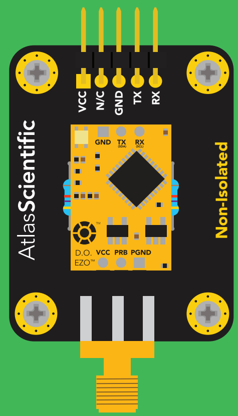
Correct wiring



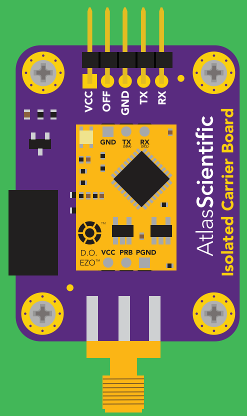
Bread board



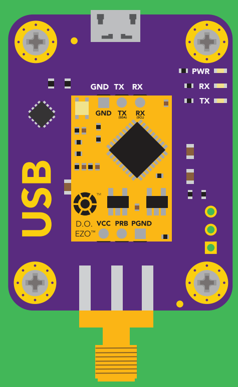
Bread board via USB



Non-Isolated
EZO™ Carrier Board



Electrically Isolated
EZO™ Carrier Board

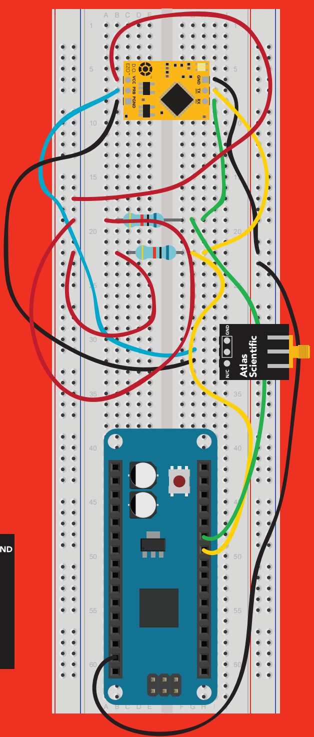


USB
carrier board

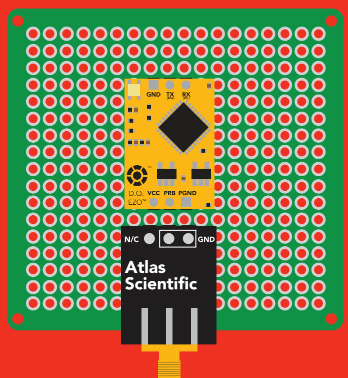
Incorrect wiring

Extended leads

Sloppy setup



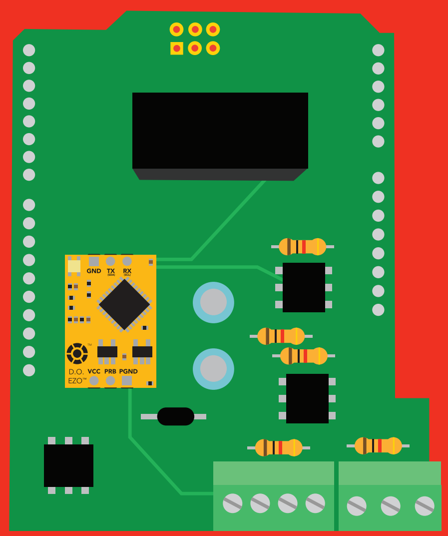
Perfboards or Protoboards



NEVER
use Perfboards or Protoboards

Flux residue and shorting wires make it very hard to get accurate readings.

*Embedded into your device

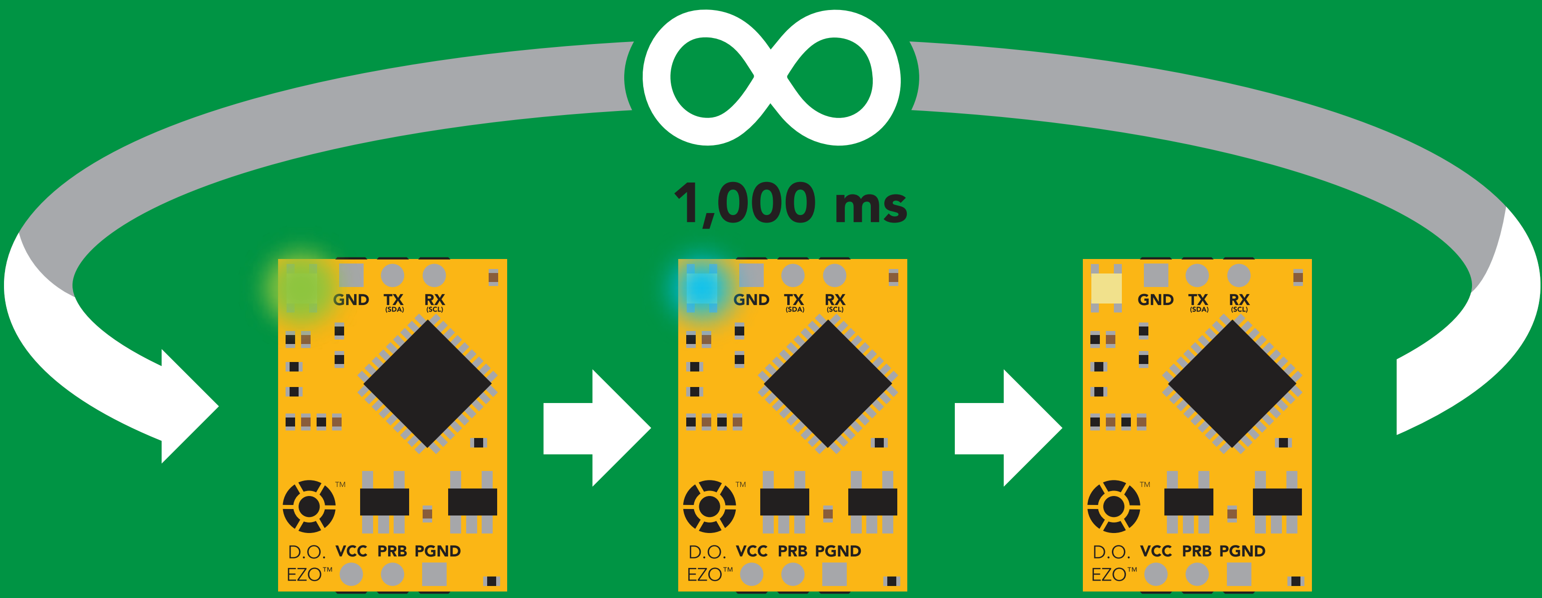


***Only after you are familar with EZO™ circuits operation**

AtlasScientific™
Environmental Robotics

Default state
UART mode

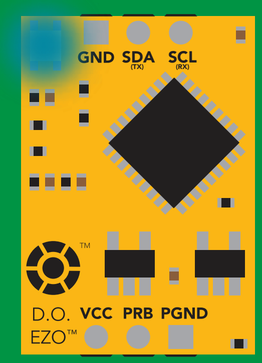
Baud	9,600
Readings	continuous
Units	mg/L
Speed	1 reading per second
LED	on



Green
Standby

Cyan
Taking reading

Transmitting

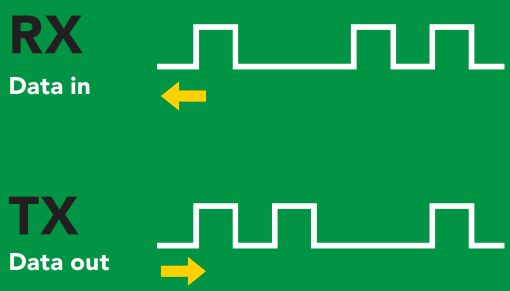


Solid Blue LED
in I²C mode
Not UART ready

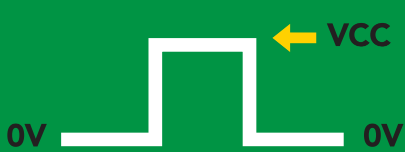
UART mode

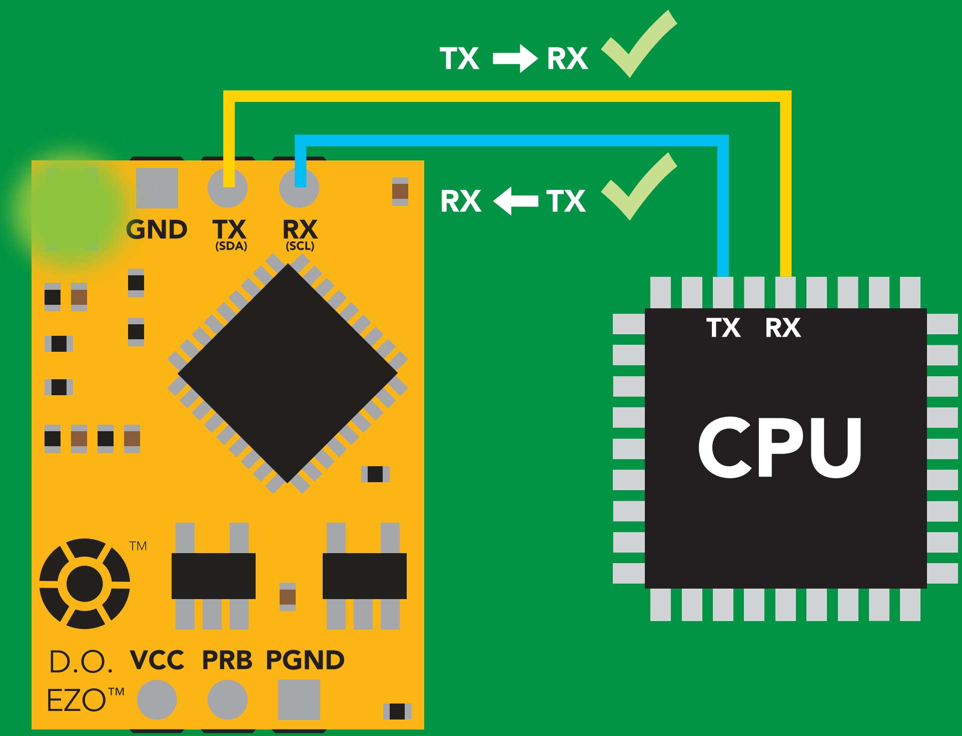
8 data bits	no parity
1 stop bit	no flow control

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



Vcc 3.3V – 5.5V



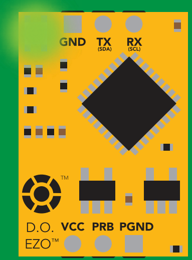


Data format

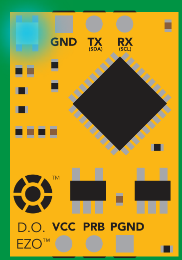
Reading	D.O.
Order	mg/L & (% sat) when enabled
Encoding	ASCII
Format	string (CSV string when % sat is enabled)
Terminator	carriage return

Data type	floating point
Decimal places	mg/L = 2 % sat = 1
Smallest string	4 characters
Largest string	40 characters

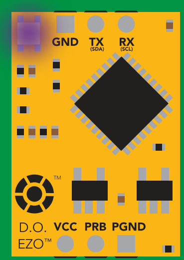
LED color definition



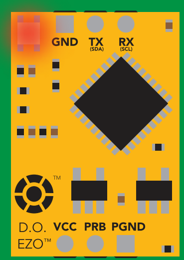
Green
UART standby



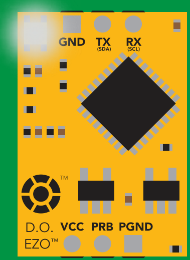
Cyan
Taking reading



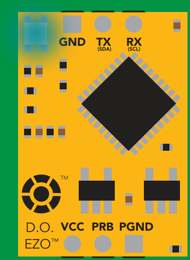
Purple
Changing baud rate



Red
Command not understood



White
Find



Blue
I2C standby

	LED ON
5V	**+0.4 mA**
3.3V	**+0.2 mA**

Settings that are retained if power is cut

Baud rate
Calibration
Continuous mode
Device name
Enable/disable parameters
Enable/disable response codes
Hardware switch to I²C mode
LED control
Protocol lock
Software switch to I²C mode

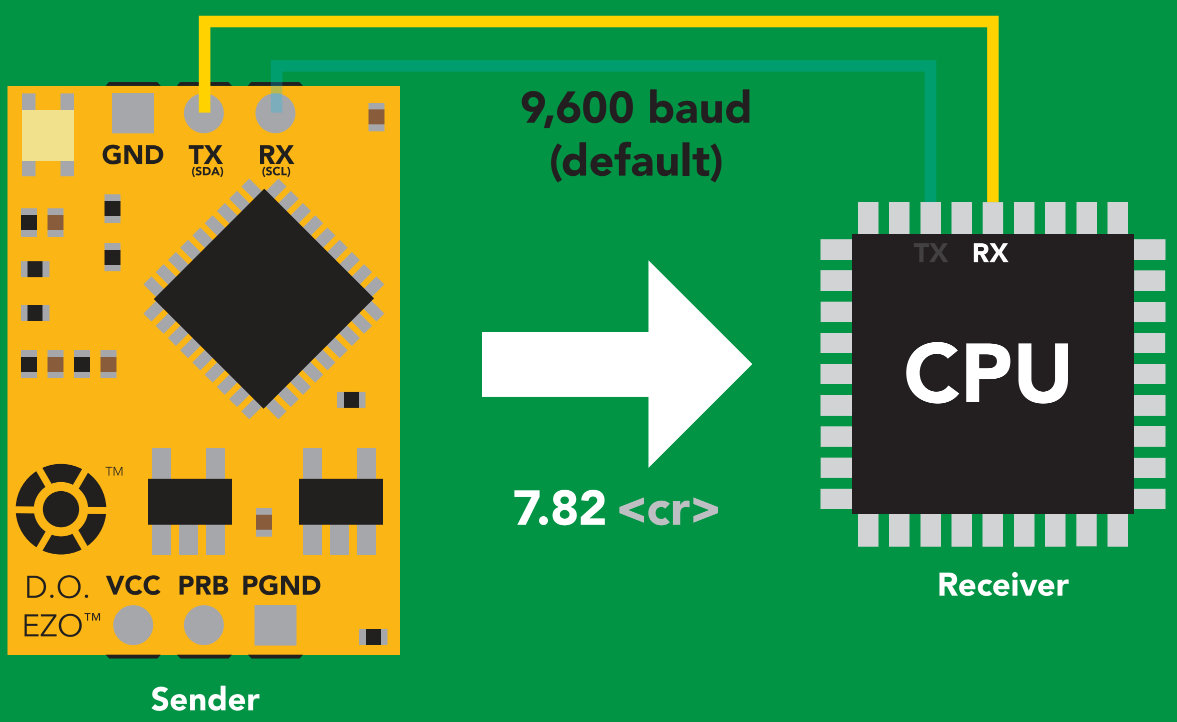
Settings that are *NOT* retained if power is cut

Find
Pressure compensation
Salinity compensation
Sleep mode
Temperature compensation

Receiving data from device

2 parts

ASCII data string	Carriage return <cr>
Command	Terminator



Advanced

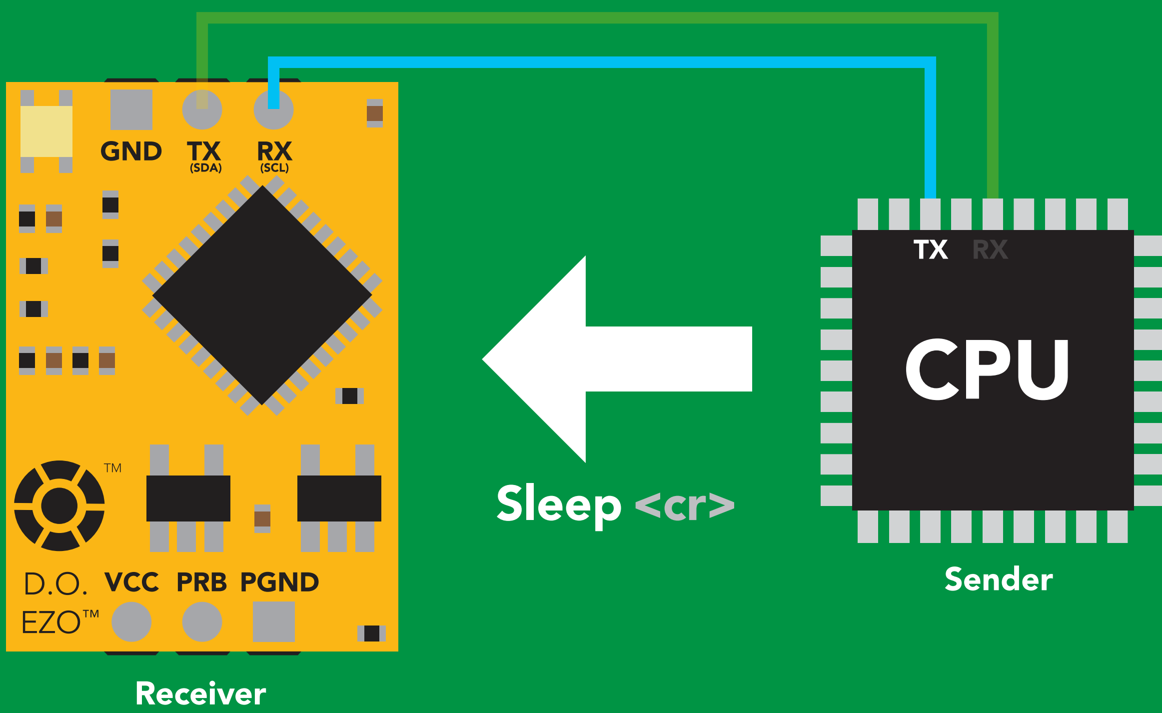
ASCII:	7	.	8	2	<cr>
Hex:	37	2E	38	32	0D
Dec:	55	46	56	50	13

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

UART mode

command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function		Default state
Baud	change baud rate	pg. 32	9,600
C	enable/disable continuous reading	pg. 18	enabled
Cal	performs calibration	pg. 20	n/a
Export	export calibration	pg. 21	n/a
Factory	enable factory reset	pg. 34	n/a
Find	finds device with blinking white LED	pg. 17	n/a
i	device information	pg. 28	n/a
I2C	change to I^2C mode	pg. 35	not set
Import	import calibration	pg. 22	n/a
L	enable/disable LED	pg. 16	enabled
Name	set/show name of device	pg. 27	not set
O	enable/disable parameters	pg. 26	mg/L
P	atmospheric pressure compensation	pg. 25	101.3 kPa
Plock	enable/disable protocol lock	pg. 33	disabled
R	returns a single reading	pg. 19	n/a
S	salinity compensation	pg. 24	n/a
Sleep	enter sleep mode/low power	pg. 31	n/a
Status	retrieve status information	pg. 30	n/a
T	temperature compensation	pg. 23	20°C
*OK	enable/disable response codes	pg. 29	enable

LED control

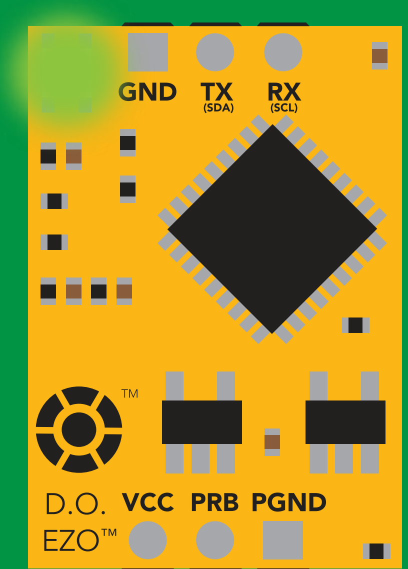
Command syntax

L,1 <cr> LED on **default**

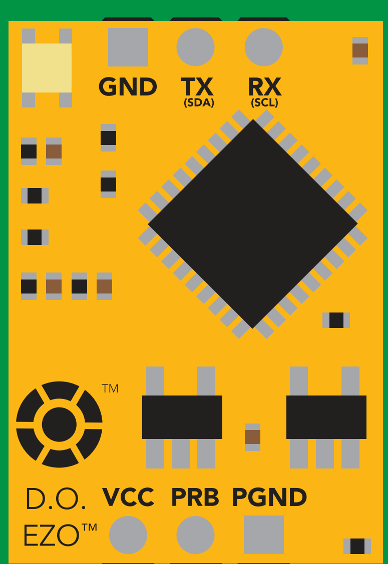
L,0 <cr> LED off

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> *OK <cr>



L,1



L,0

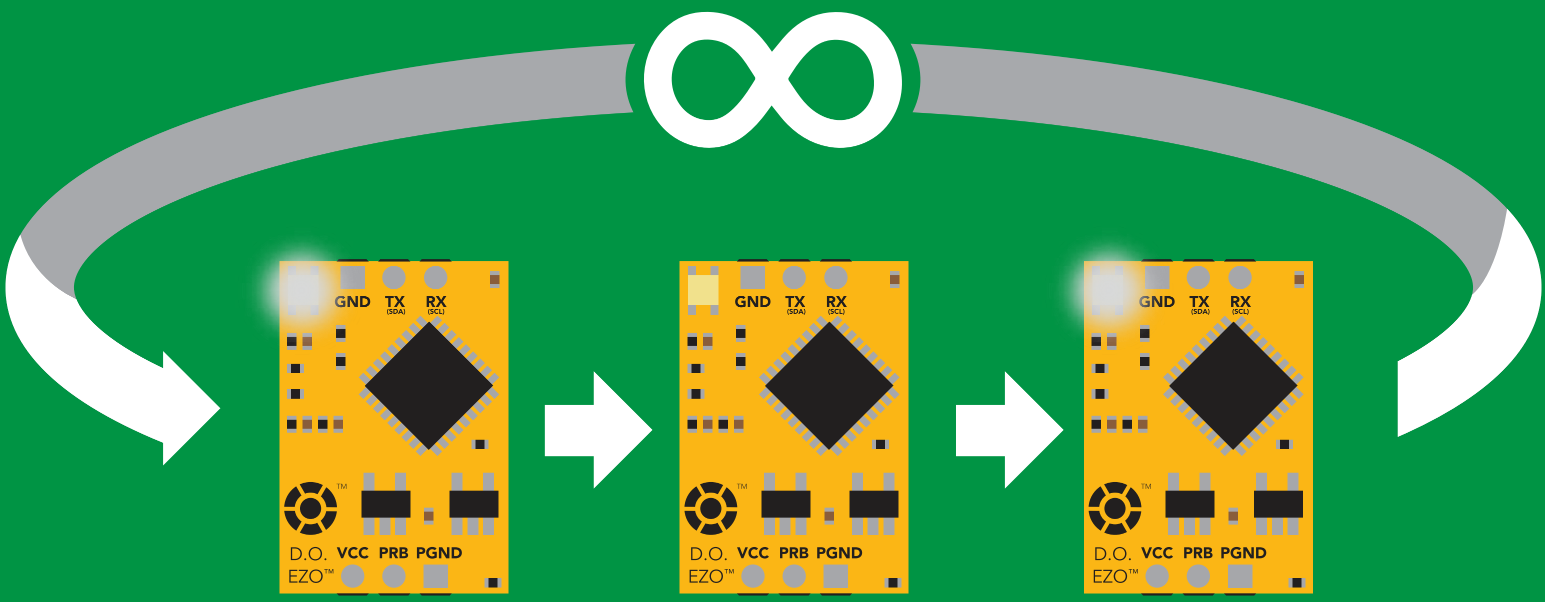
Find

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 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?

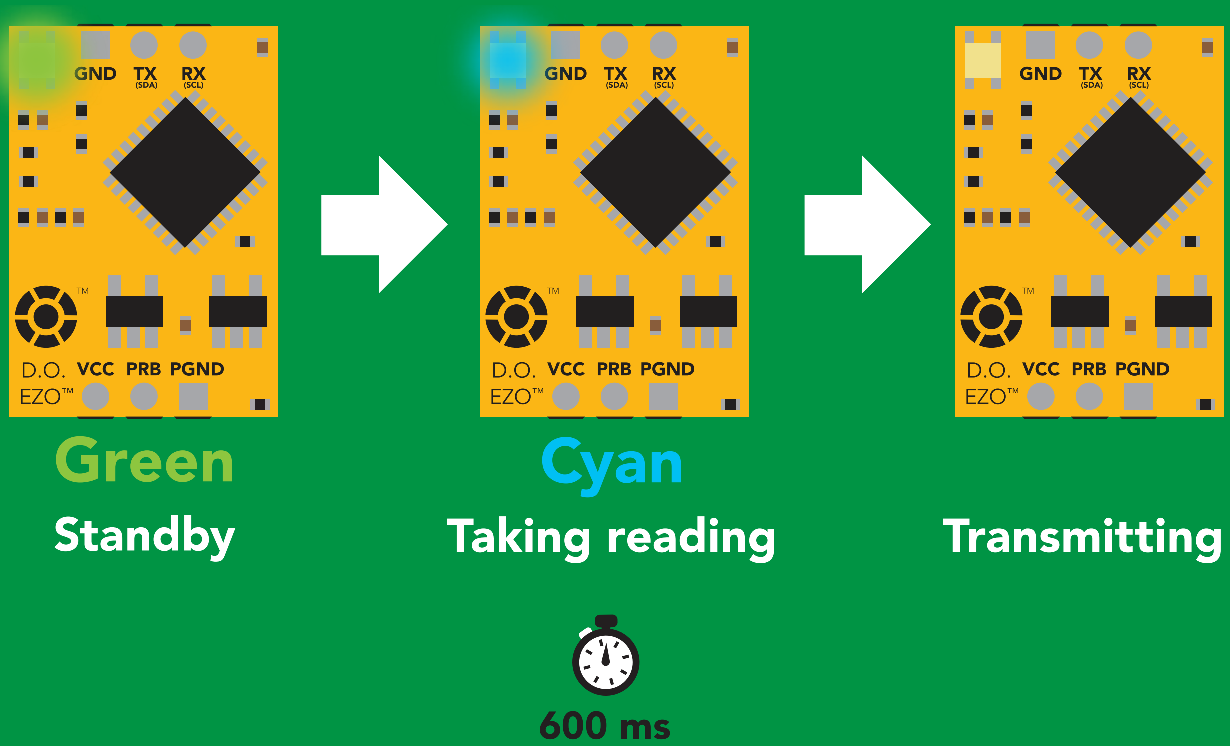
Example	Response
C,1 <cr>	*OK <cr> DO (1 sec) <cr> DO (2 sec) <cr> DO (3 sec) <cr>
C,30 <cr>	*OK <cr> DO (30 sec) <cr> DO (60 sec) <cr> DO (90 sec) <cr>
C,0 <cr>	*OK <cr>
C,? <cr>	?C,1 <cr> or ?C,0 <cr> or ?C,30 <cr> *OK <cr>

Single reading mode

Command syntax

R <cr> **takes single reading**

Example	Response
R <cr>	**7.82** <cr> ***OK** <cr>



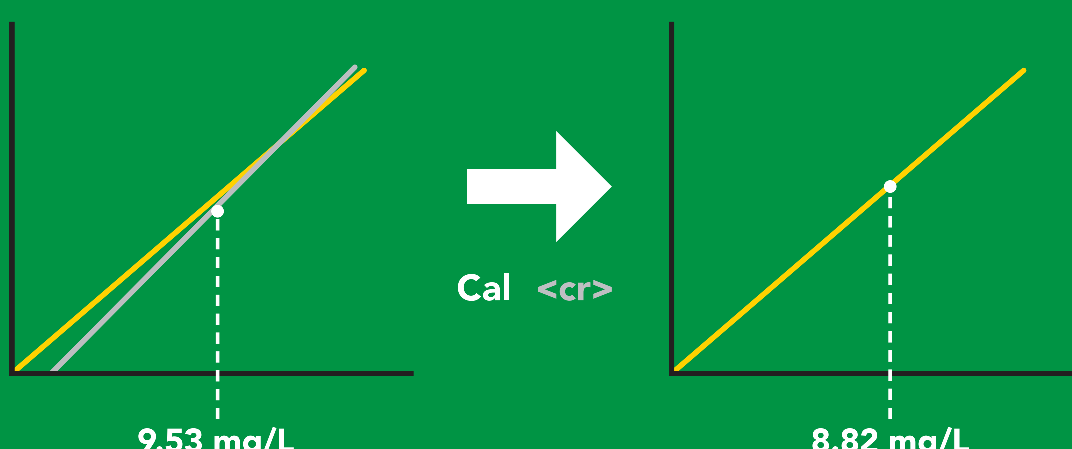
Calibration

Command syntax

The EZO™ Dissolved Oxygen circuit uses single and/or two point calibration

Cal	<cr>	calibrate to atmospheric oxygen levels
Cal,0	<cr>	calibrate device to 0 dissolved oxygen
Cal,clear	<cr>	delete calibration data
Cal,?	<cr>	device calibrated?

Example	Response
Cal <cr>	*OK <cr>
Cal,0 <cr>	*OK <cr>
Cal,clear <cr>	*OK <cr>
Cal,? <cr>	?Cal,0 <cr> or ?Cal,1 <cr> (single point) or ?Cal,2 <cr> (two point) *OK <cr>



Export calibration

Command syntax

Export: Use this command to download calibration settings

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

Example

Response

Export,? <cr>

10,120 <cr>

Response breakdown

10, 120

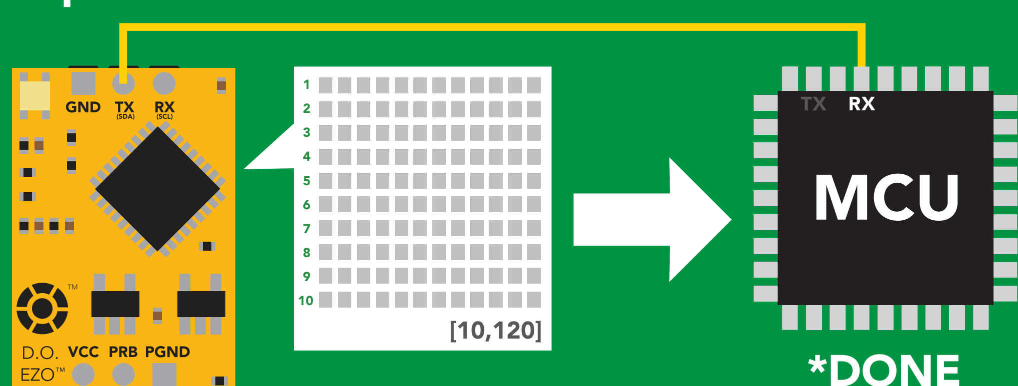
# of strings to export — # of bytes to export

Export strings can be up to 12 characters long, and is always followed by <cr>

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

Disabling *OK simplifies this process

Export <cr>



AtlasScientific™ Environmental Robotics

Import calibration

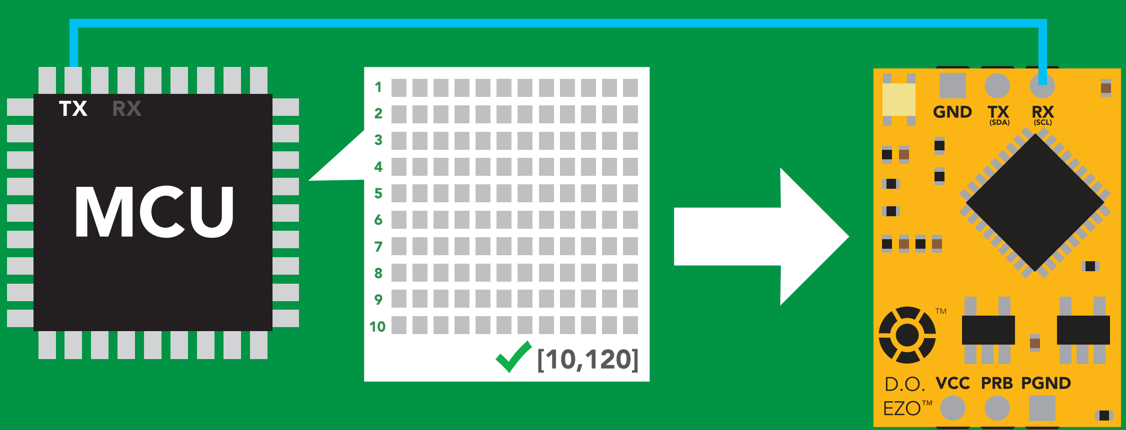
Command syntax

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

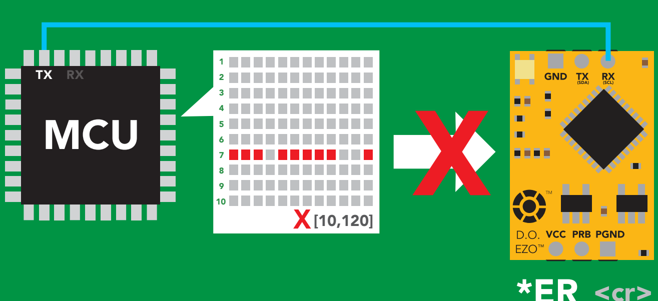
Import,n <cr> import calibration string to new device

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>

Import,n <cr>



*OK <cr>
system will reboot



*ER <cr>

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

Temperature compensation

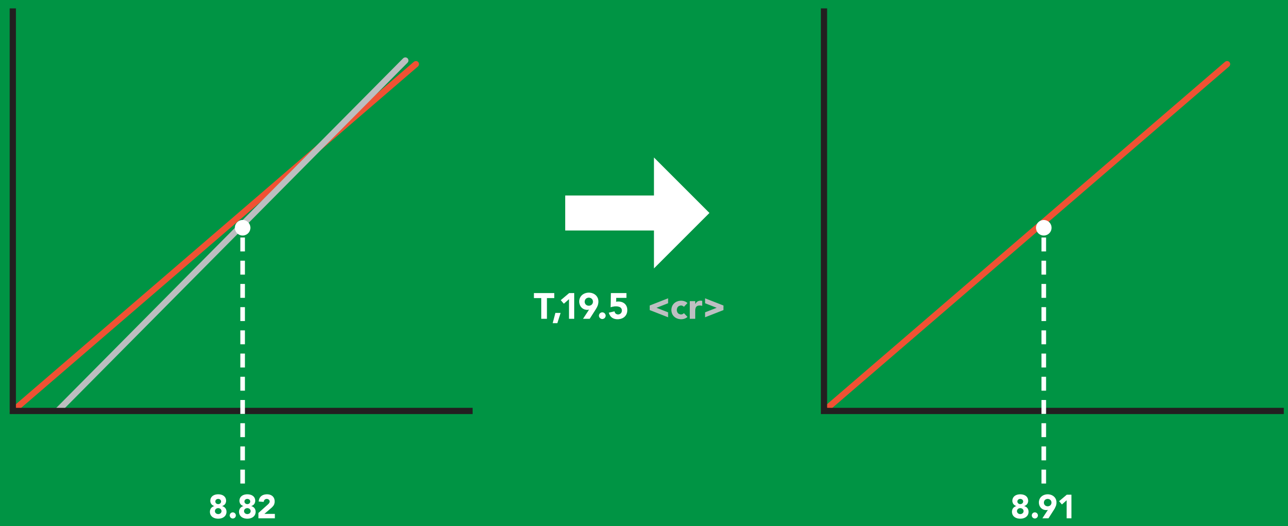
Command syntax

Default temperature = 20°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*

This is a new command for firmware V2.13

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



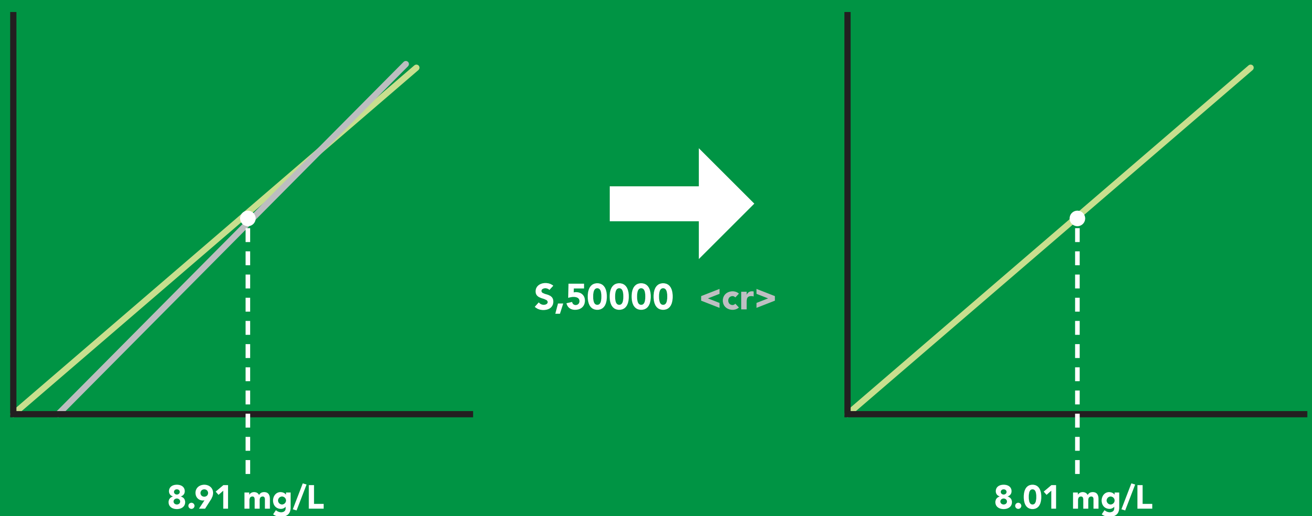
Salinity compensation

Command syntax

Default value = 0 µs
If the conductivity of your water is less than 2,500µS this command is irrelevant

S,n	<cr>	n = any value in microsiemens
S,n,ppt	<cr>	n = any value in ppt
S,?	<cr>	compensated salinity value?

Example	Response
S,50000 <cr>	*OK <cr>
S,37.5,ppt <cr>	*OK <cr>
S,? <cr>	?S,50000,µS <cr> or ?S,37.5,ppt <cr> *OK <cr>

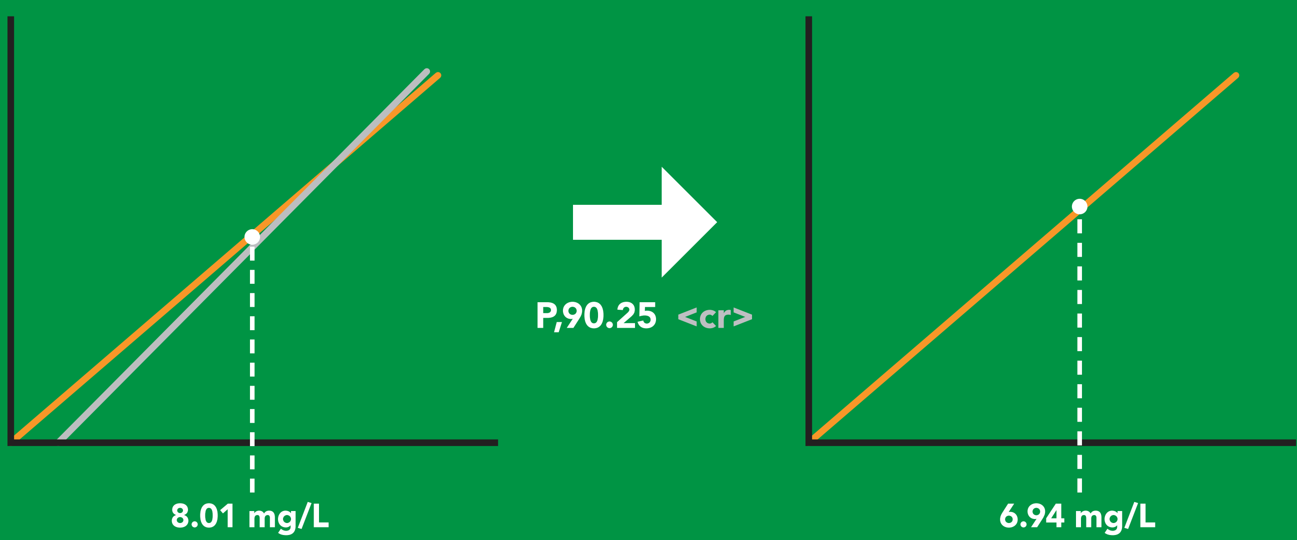


Atmospheric pressure compensation

Command syntax

P,n	<cr>	n = any value in kPa
P,?	<cr>	compensated pressure value?

Example	Response
P,90.25 <cr>	*OK <cr>
P,? <cr>	?,P,90.25 <cr> *OK <cr>



Enable/disable parameters from output string

Command syntax

O, [parameter],[1,0]	<cr>	enable or disable output parameter
O,?	<cr>	enabled parameter?

Example	Response
O,mg,1 / O,mg,0 <cr>	*OK <cr> enable / disable mg/L
O,%,1 / O,%,0 <cr>	*OK <cr> enable / disable percent saturation
O,? <cr>	?,O,%,mg <cr> if both are enabled

Parameters

mg	mg/L
%	percent saturation

Followed by 1 or 0

1	enabled
0	disabled

*** If you disable all possible data types your readings will display "no output".**

AtlasScientific 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

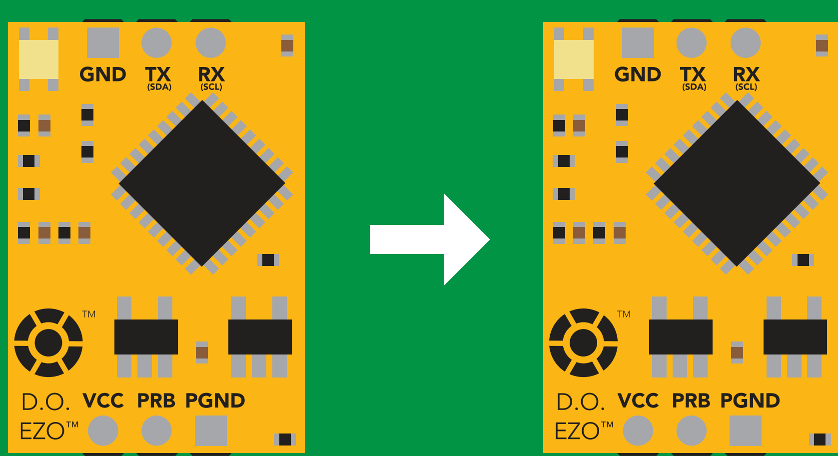
n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

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> *OK <cr>

Name,zzt

Name,?



*OK <cr>

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

Device information

Command syntax

i <cr> **device information**

Example	Response
i <cr>	?i,D.O.,1.98 <cr> *OK <cr>

Response breakdown

?i,	D.O.,	1.98
	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>	7.82 <cr> *OK <cr>
*OK,0 <cr>	no response, *OK disabled
R <cr>	7.82 <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> *OK <cr>

Response breakdown

?Status,	P,	5.038
	Reason for restart	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> *SL <cr>
Any command	*WA <cr> wakes up device

	STANDBY	SLEEP
5V	13.1 mA	0.66 mA
3.3V	12 mA	0.3 mA

Standby
13.1 mA

Sleep
0.66 mA

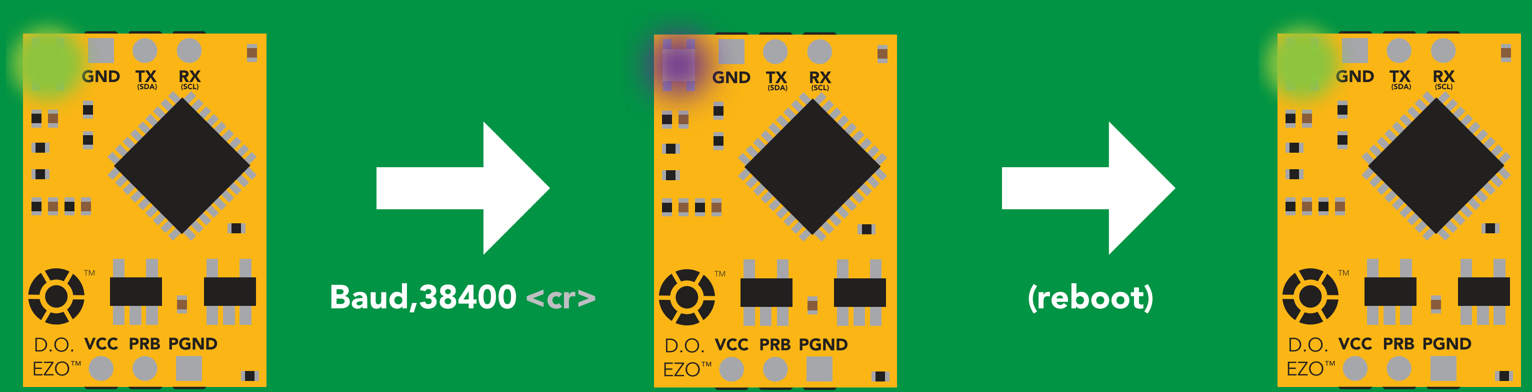
Change baud rate

Command syntax

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

Example	Response
Baud,38400 <cr>	*OK <cr>
Baud,? <cr>	?Baud,38400 <cr> *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.

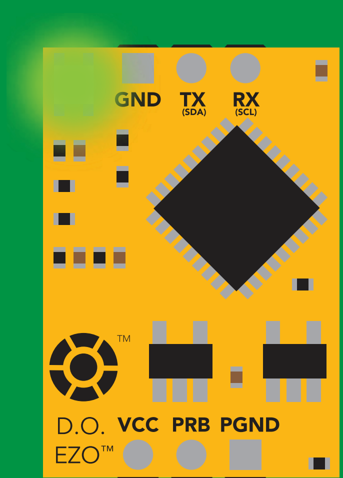
Plock,1 <cr> enable Plock

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

Plock,? <cr> Plock on/off?

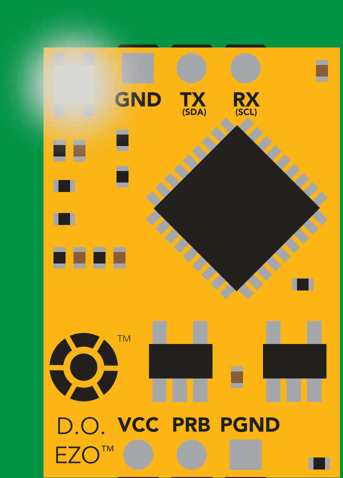
Example	Response
Plock,1 <cr>	*OK <cr>
Plock,0 <cr>	*OK <cr>
Plock,? <cr>	?Plock,1 <cr> or ?Plock,0 <cr>

Plock,1



*OK <cr>

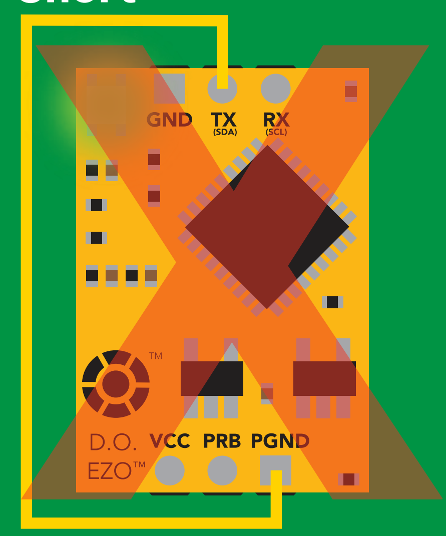
I2C,100



cannot change to I²C

*ER <cr>

Short



cannot change to I²C

Factory reset

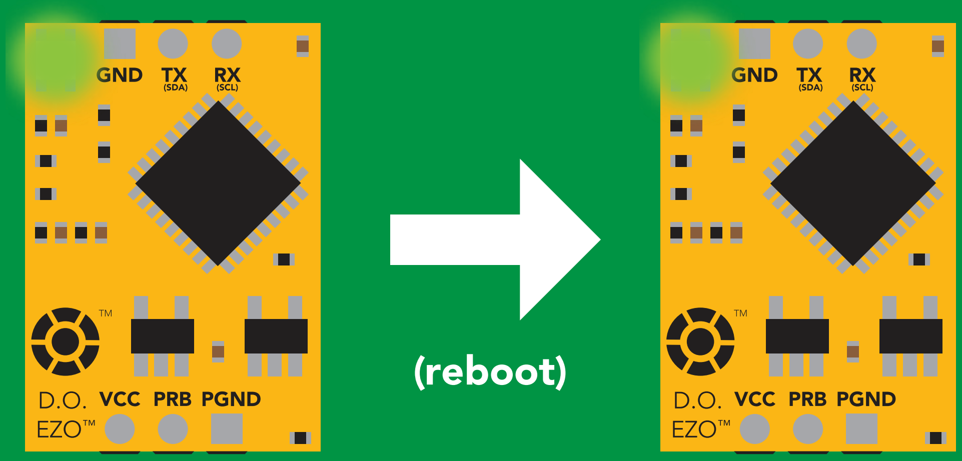
Command syntax

Clears calibration
LED on
"*OK" enabled

Factory <cr> enable factory reset

Example	Response
Factory <cr>	*OK <cr>

Factory <cr>



*OK <cr>

*RS <cr>
*RE <cr>

Baud rate will not change

Change to I²C mode

Command syntax

Default I²C address 97 (0x61)

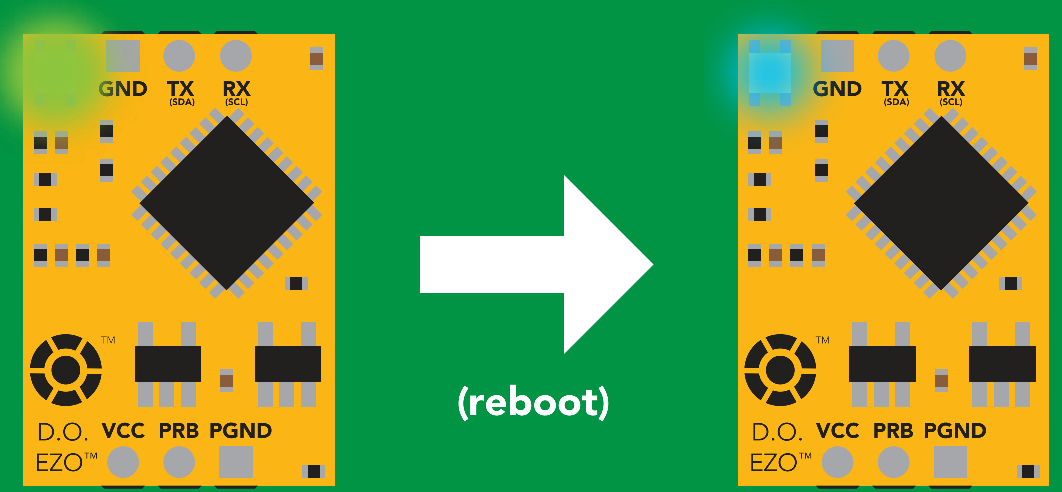
I2C,n <cr> sets I²C address and reboots into I²C mode

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



Green
*OK <cr>

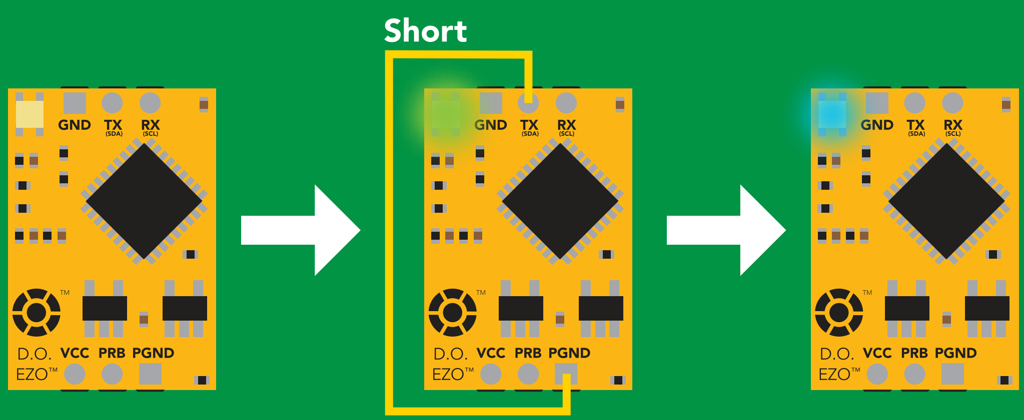
Blue
now in I²C mode

Manual switching to I^2C

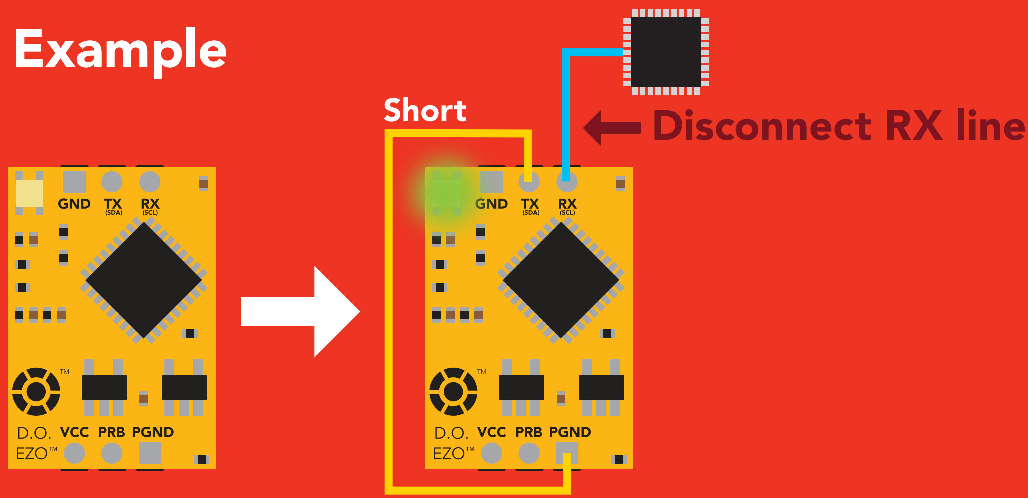
- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to PGND
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from Green to Blue
- Disconnect ground (power off)
- Reconnect all data and power

Manually switching to I^2C will set the I^2C address to 97 (0x61)

Example



Wrong Example



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™ device into I²C mode click here

Settings that are retained if power is cut

Calibration
Change I²C address
Enable/disable parameters
Hardware switch to UART mode
LED control
Protocol lock
Software switch to UART mode

Settings that are *NOT* retained if power is cut

Find
Pressure compensation
Salinity compensation
Sleep mode
Temperature compensation

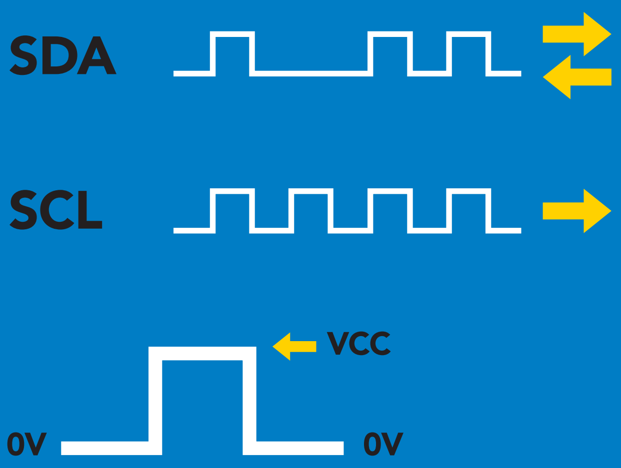
AtlasScientific™
Environmental Robotics

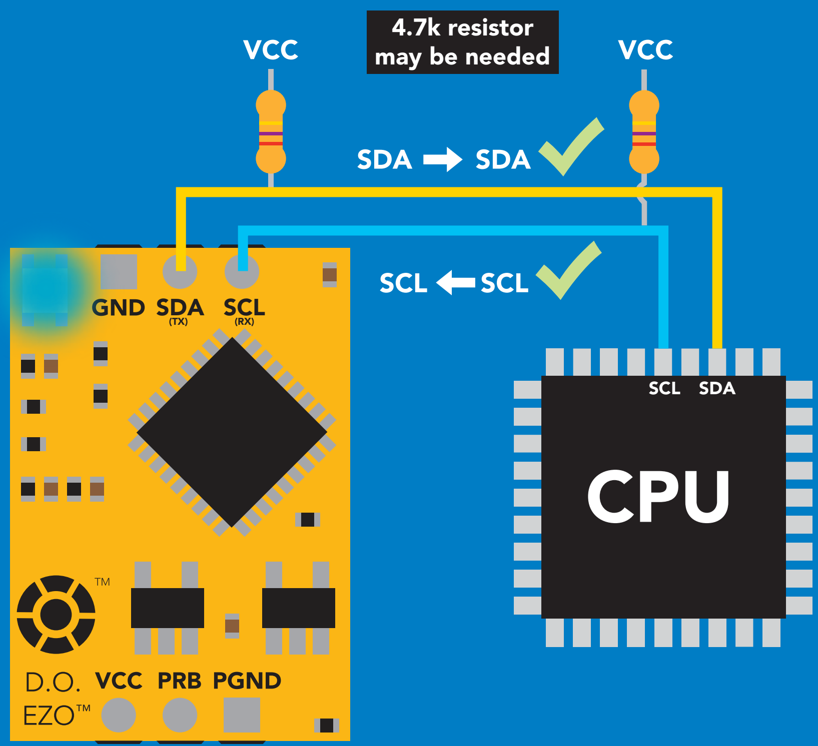
I²C mode

I²C address (0x01 – 0x7F)
97 (0x61) default

Vcc 3.3V – 5.5V

Clock speed 100 – 400 kHz





Data format

Reading	Dissolved Oxygen	**Data type**	floating point
Order	mg/L & (% sat) when enabled	**Decimal places**	mg/L = 2 % sat = 1
Encoding	ASCII	**Smallest string**	4 characters
Format	string (CSV string when % sat is enabled)	**Largest string**	16 characters

AtlasScientific™ Environmental Robotics

Sending commands to device

5 parts

Start | I²C address | Write | Command (not case sensitive) | Stop

I²C address: 97 (0x61)

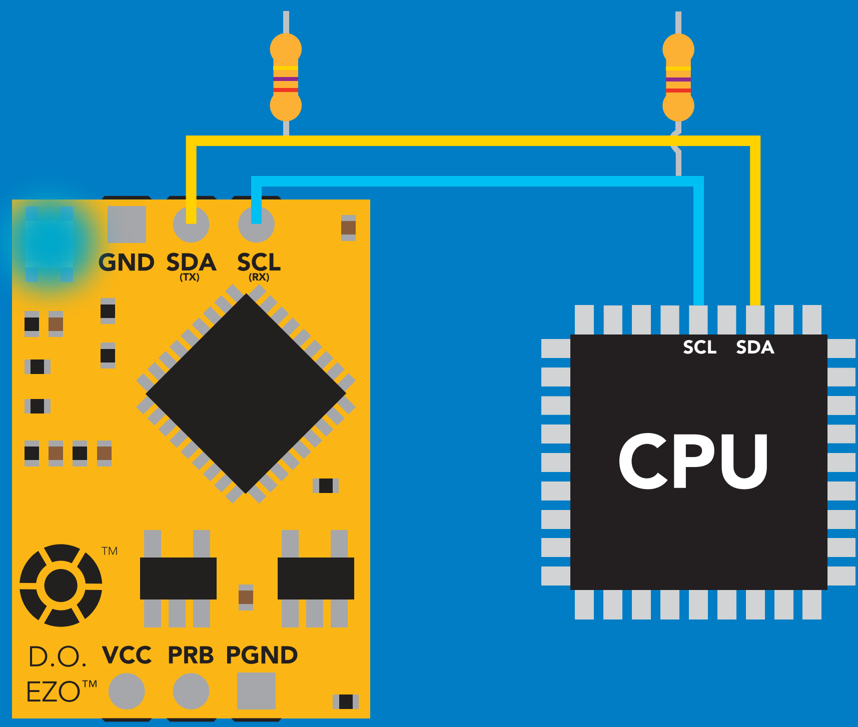
Command: ASCII command string

Example

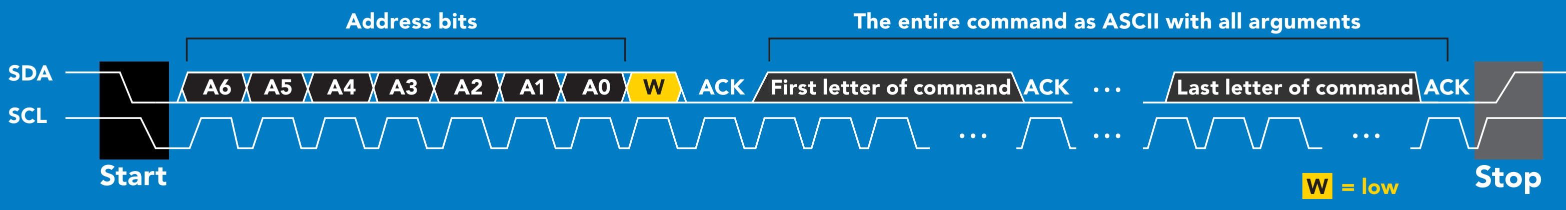
Start | 97 (0x61) | Write | Sleep | Stop

97 (0x61): I²C address

Sleep: Command



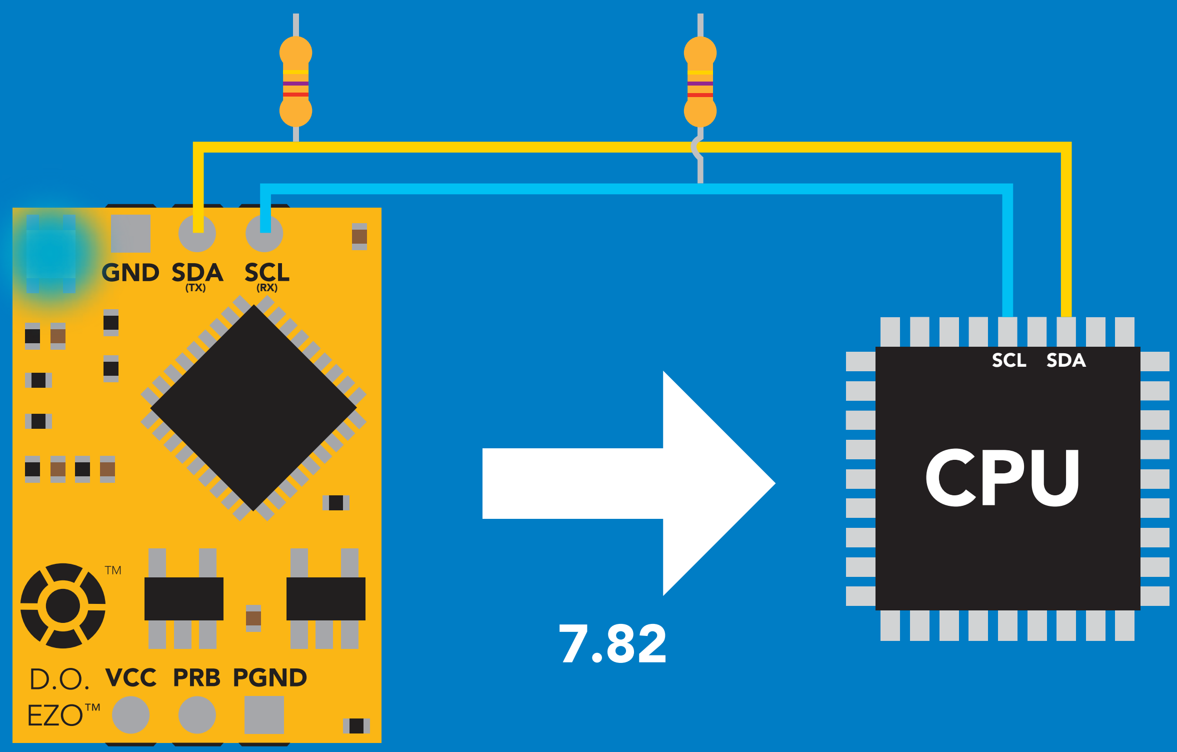
Advanced



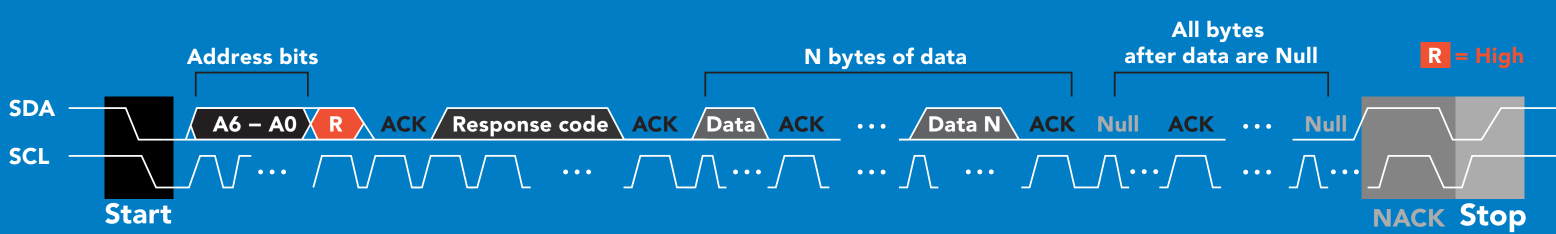
Requesting data from device

7 parts

Start	I²C address	Read	Response code	Data string	Null	Stop
	97 (0x61)		1 byte	"7.82"	Terminator (Dec 0)	



Advanced



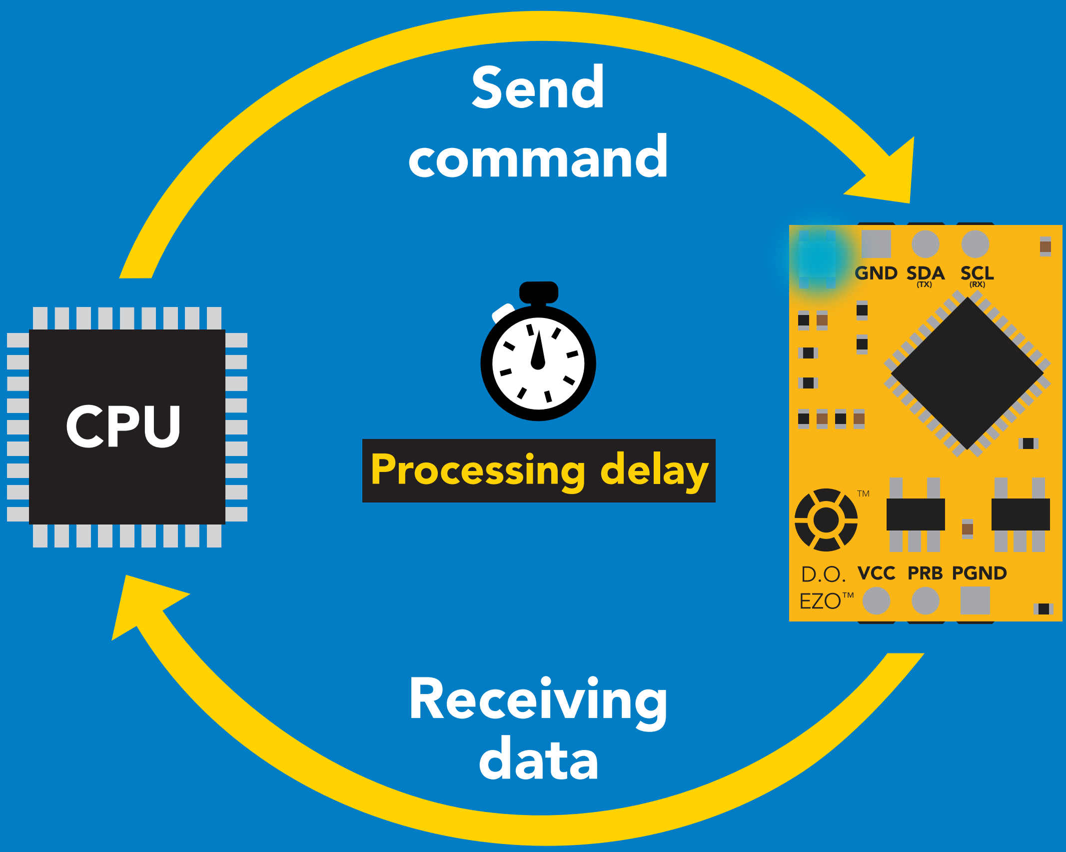
1	55	46	56	50	0	= 7.82
Dec	ASCII				Dec	

AtlasScientific™
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;
```

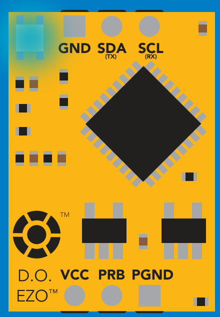
The response code will always be 254, if you do not wait for the processing delay.

Response codes

Single byte, not string

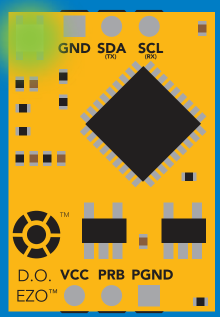
255	no data to send
254	still processing, not ready
2	syntax error
1	successful request

LED color definition



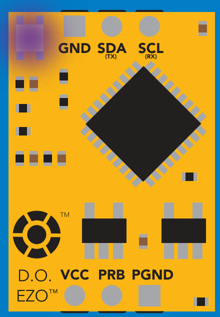
Blue

I²C standby



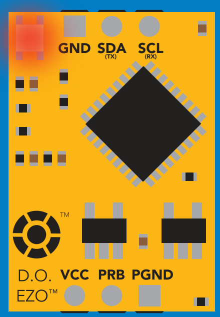
Green

Taking reading



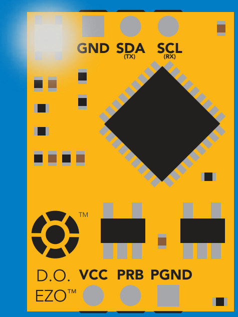
Purple

Changing
I²C address



Red

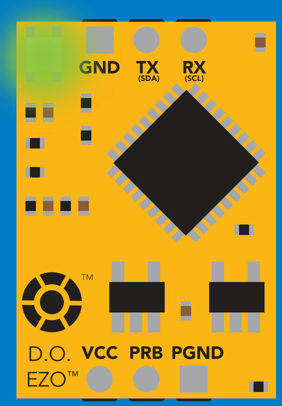
Command
not understood



White

Find

	LED ON
5V	+0.4 mA
3.3V	+0.2 mA



Solid Green LED

in UART mode
Not I²C ready

AtlasScientific™
Environmental Robotics

I^2C mode
command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function	
Baud	change back to UART mode	pg. 61
Cal	performs calibration	pg. 47
Export	export calibration	pg. 48
Factory	enable factory reset	pg. 60
Find	finds device with blinking white LED	pg. 45
i	device information	pg. 55
I2C	change I^2C address	pg. 59
Import	import calibration	pg. 49
L	enable/disable LED	pg. 44
Name	set/show name of device	pg. 54
O	removing parameters	pg. 53
P	atmospheric pressure compensation	pg. 52
Plock	enable/disable protocol lock	pg. 58
R	returns a single reading	pg. 46
S	salinity compensation	pg. 51
Sleep	enter sleep mode/low power	pg. 57
Status	retrieve status information	pg. 56
T	temperature compensation	pg. 50

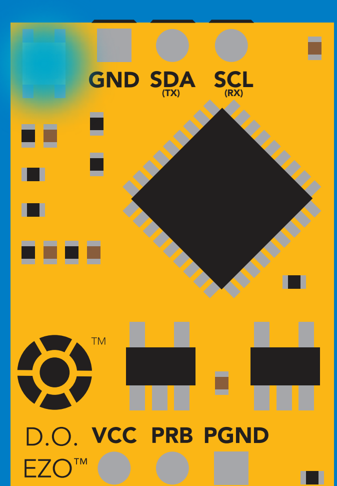
LED control

Command syntax

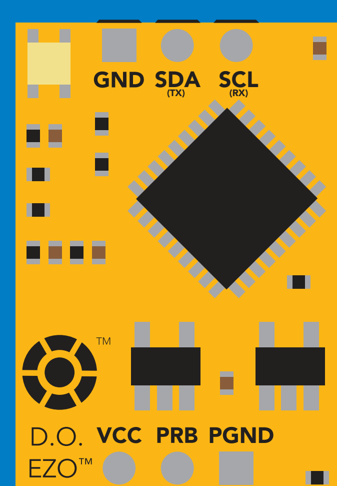
300ms processing delay

- **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** 1 (Dec) ?L,0 (ASCII) 0 (Null)



L,1



L,0

Find

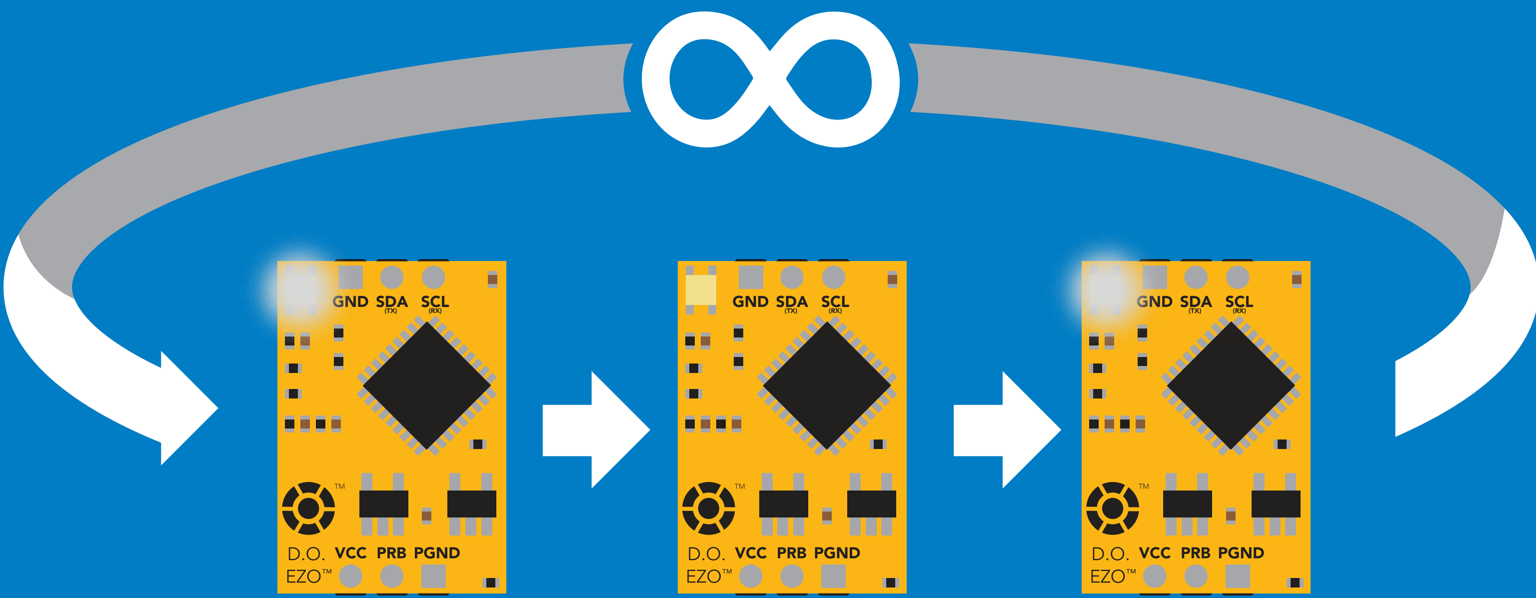
300ms processing delay

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 1 Dec 0 Null



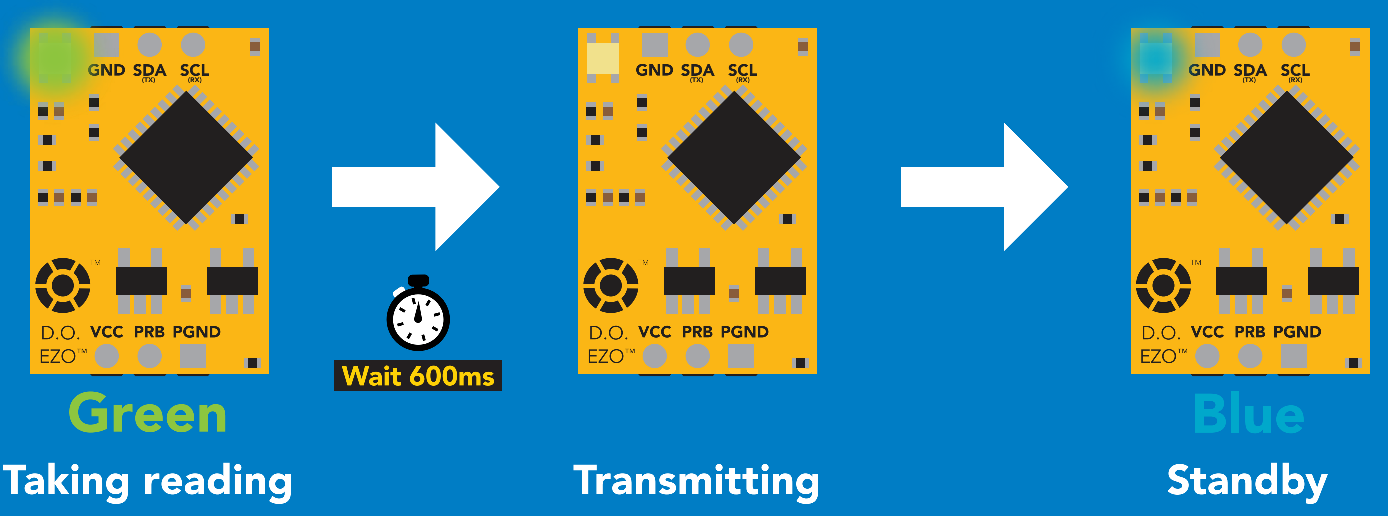
Taking reading

Command syntax

600ms processing delay

R return 1 reading

Example	Response
R	Wait 600ms · 1 Dec · 7.82 ASCII · 0 Null



Calibration

Command syntax

1300ms processing delay

Cal	calibrate to atmospheric oxygen levels
Cal,0	calibrate device to 0 dissolved oxygen
Cal,clear	delete calibration data
Cal,?	device calibrated?

The EZO™ Dissolved Oxygen circuit uses single and/or two point calibration

Example	Response
Cal	Wait 1300ms — 1 (Dec) 0 (Null)
Cal,0	Wait 1300ms — 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) 0 (Null) or 1 (Dec) ?Cal,2 (ASCII) 0 (Null)

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,?	Wait 300ms — 1 (Dec) 10,120 (ASCII) 0 (Null)
Export	Wait 300ms — 1 (Dec) 59 6F 75 20 61 72 (ASCII) 0 (Null) (1 of 10)
Export	Wait 300ms — 1 (Dec) 65 20 61 20 63 6F (ASCII) 0 (Null) (2 of 10)
(7 more)	⋮
Export	Wait 300ms — 1 (Dec) 6F 6C 20 67 75 79 (ASCII) 0 (Null) (10 of 10)
Export	Wait 300ms — 1 (Dec) *DONE (ASCII) 0 (Null)

Response breakdown

10, 120

10: # of strings to export
120: # of bytes to export

Export strings can be up to 12 characters long

AtlasScientific Environmental Robotics

Import calibration

300ms processing delay

Command syntax

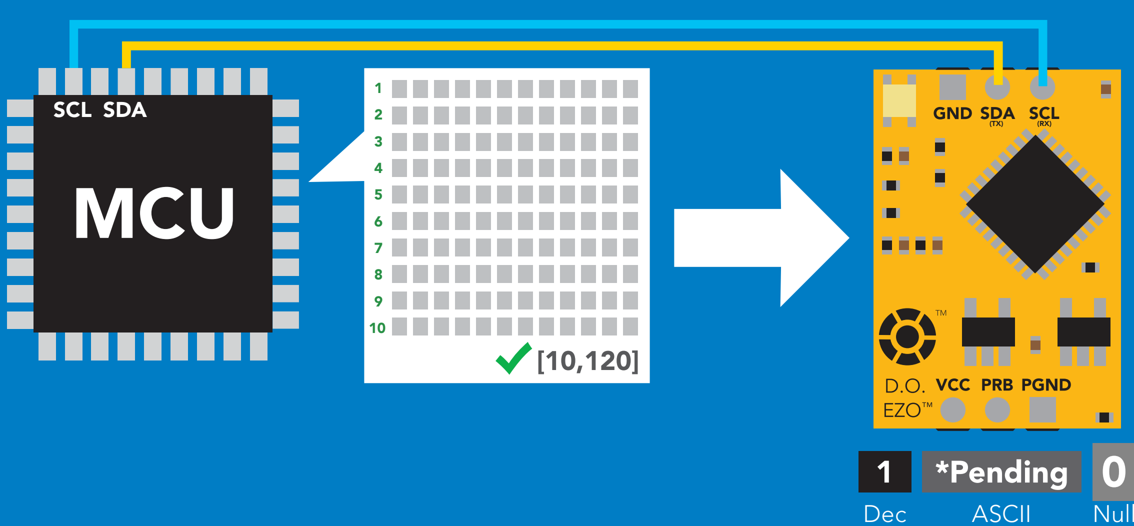
Import: Use this command to upload calibration settings to one or more devices.

Import,n	import calibration string to new device

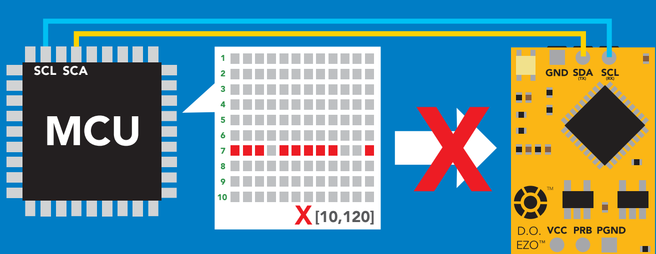
Example

Example		Response
Import, 59 6F 75 20 61 72	(1 of 10)	Wait 300ms · 1 Dec · 0 Null
Import, 65 20 61 20 63 6F	(2 of 10)	Wait 300ms · 1 Dec · 0 Null
⋮		⋮
Import, 6F 6C 20 67 75 79	(10 of 10)	Wait 300ms · 1 Dec · 0 Null

Import,n



system will reboot



reboot

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

AtlasScientific™ Environmental Robotics

Temperature compensation

Command syntax

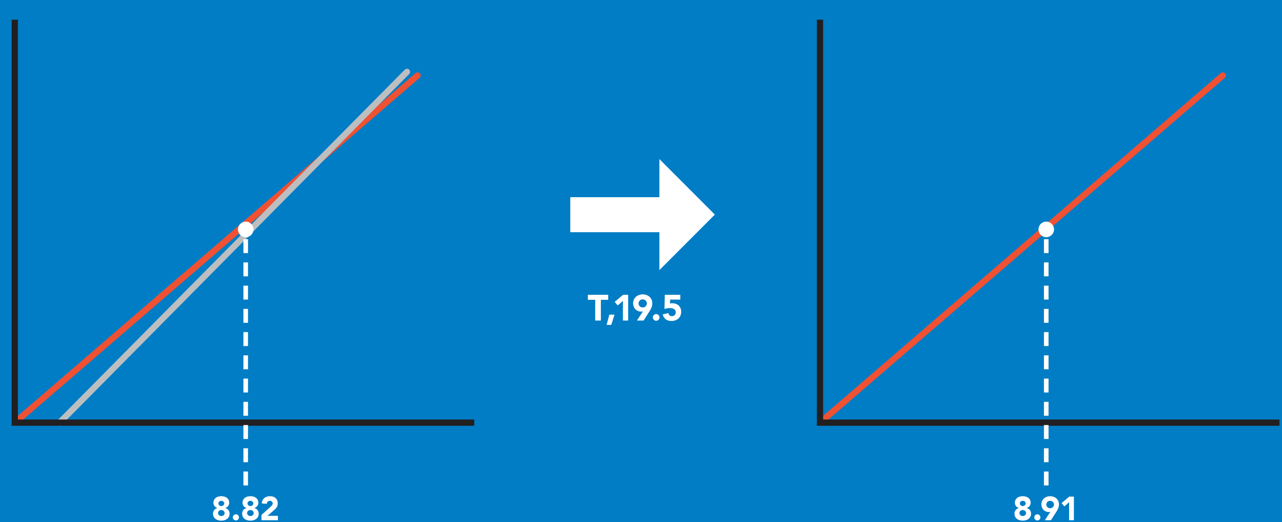
Default temperature = 20°C
Temperature is always in Celsius
Temperature is not retained if power is cut

T,n	n = any value; floating point or int
T,?	compensated temperature value?
RT,n	set temperature compensation and take a reading*

300ms processing delay

This is a new command for firmware V2.13

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)



Salinity compensation

Command syntax

300ms processing delay

S,n	n = any value in microsiemens	default
S,n,ppt	n = any value in ppt	
S,?	compensated salinity value?	

Example	Response
S,50000	Wait 300ms — 1 (Dec) 0 (Null)
S,37.5,ppt	Wait 300ms — 1 (Dec) 0 (Null)
S,?	Wait 300ms — 1 (Dec) ?S,50000,µS (ASCII) 0 (Null) or 1 (Dec) ?S,37.5,ppt (ASCII) 0 (Null)

If the conductivity of your water is less than 2,500µS this command is irrelevant

AtlasScientific
Environmental Robotics

Atmospheric pressure compensation

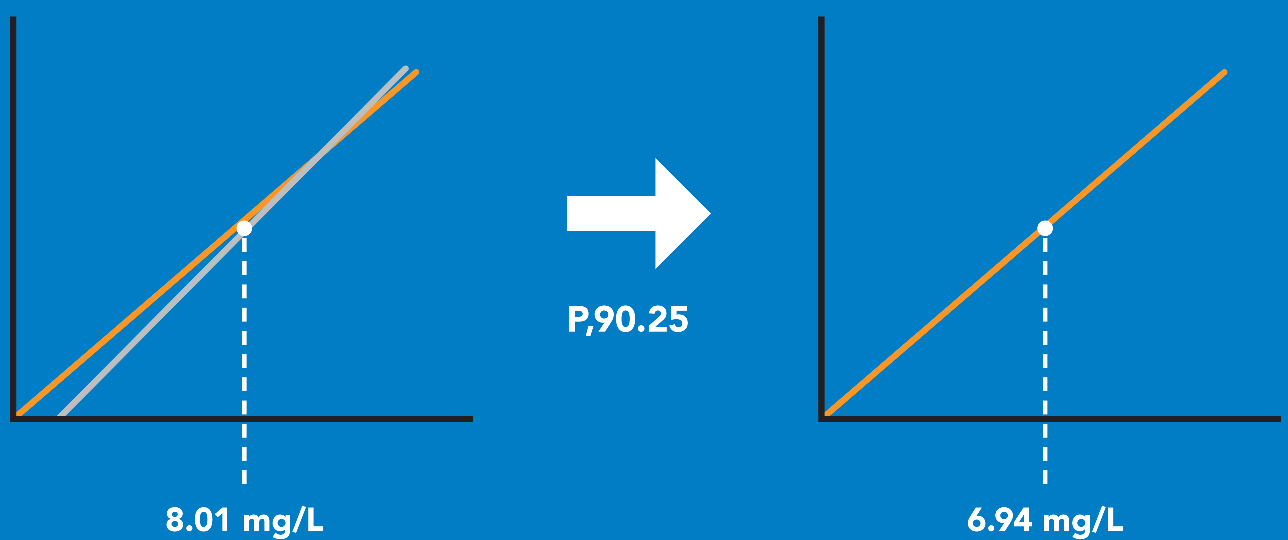
Command syntax

300ms processing delay

P,n n = any value in kPa

P,? compensated pressure value?

Example	Response
P,90.25	Wait 300ms · 1 (Dec) · 0 (Null)
P,?	Wait 300ms · 1 (Dec) · ?,P,90.25 (ASCII) · 0 (Null)



Enable/disable parameters from output string

Command syntax

300ms processing delay

O, [parameter],[1,0]	enable or disable output parameter
O,?	enabled parameter?

Example	Response	
O,mg,1 / O,mg,0	Wait 300ms — 1 Dec, 0 Null	enable / disable mg/L
O,%,1 / O,%,0	Wait 300ms — 1 Dec, 0 Null	enable / disable percent saturation
O,?	Wait 300ms — 1 Dec, ?,O,%,mg ASCII, 0 Null	if both are enabled

Parameters

mg	mg/L
%	percent saturation

Followed by 1 or 0

1	enabled
0	disabled

* If you disable all possible data types your readings will display "no output".

Naming device

300ms processing delay

Command syntax

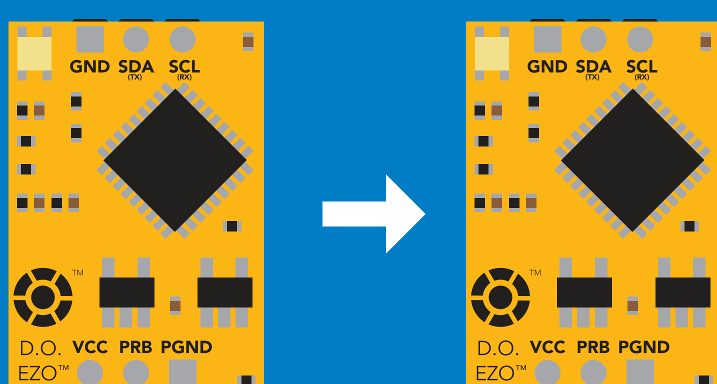
Do not use spaces in the name

Name,n	set name	n = _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ (1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16)
Name,	clears name	
Name,?	show name	

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 Name,?



1 0 1 ?Name,zzt 0

Atlas Scientific™ Environmental Robotics

Device information

Command syntax

300ms processing delay

i device information

Example	Response
i	Wait 300ms · 1 (Dec) · ?i,D.O.,1.98 (ASCII) · 0 (Null)

Response breakdown

?i, D.O., 1.98

D.O. — Device

1.98 — Firmware

Reading device status

Command syntax

300ms processing delay

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

Example

Status

Response

Wait 300ms

1	?Status,P,5.038	0
Dec	ASCII	Null

Response breakdown

?Status,	P,	5.038
	Reason for restart	Voltage at Vcc

Restart codes

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

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	13.1 mA	0.66 mA
3.3V	12 mA	0.3 mA

Standby Sleep

Atlas Scientific Environmental Robotics

Protocol lock

Command syntax

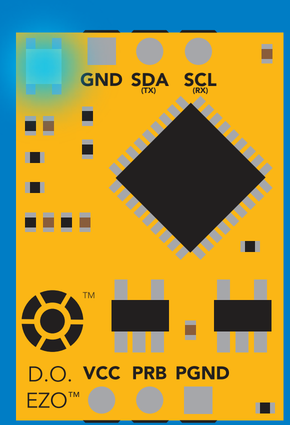
300ms processing delay

Plock,1	enable Plock	
Plock,0	disable Plock	default
Plock,?	Plock on/off?	

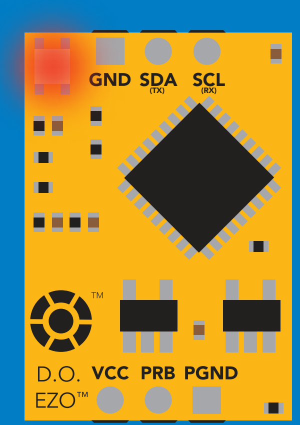
Locks device to I²C mode.

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)

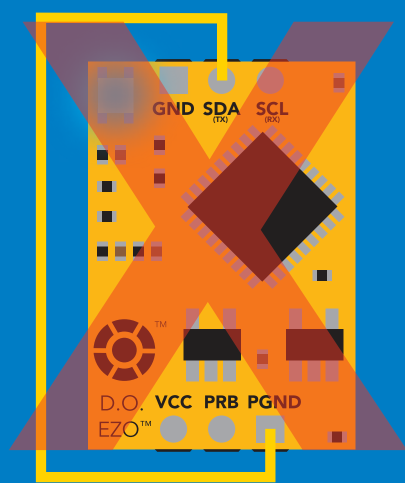
Plock,1



Baud, 9600



cannot change to UART



cannot change to UART

I²C address change

Command syntax

300ms processing delay

I2C,n	sets I²C address and reboots into I²C mode

Example	Response
I2C,100	device reboot (no response given)

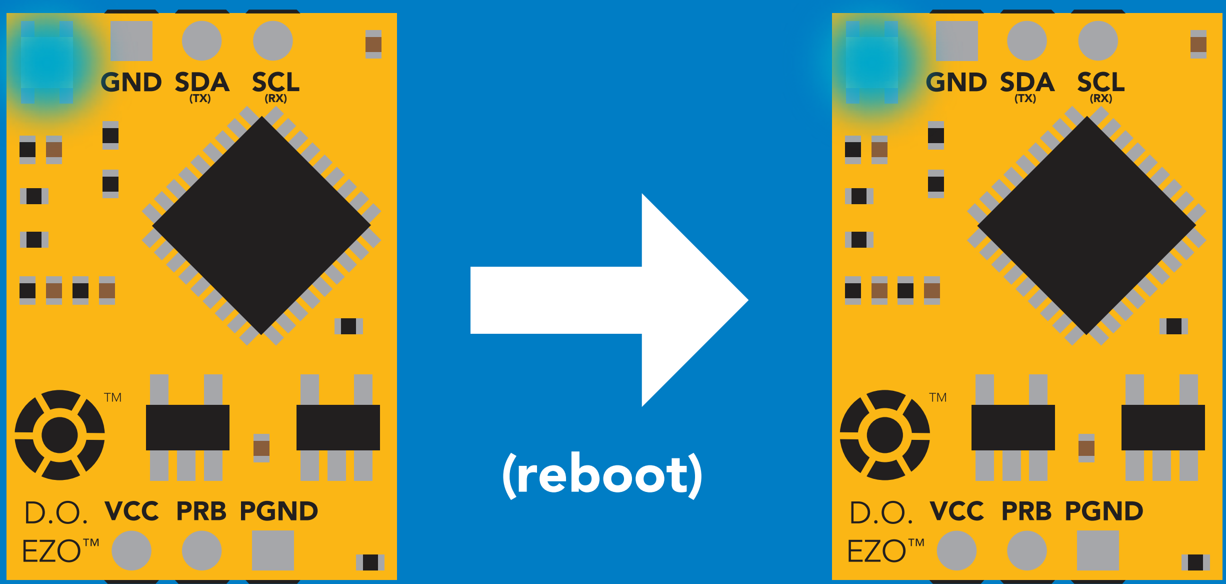
Warning!

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

Default I²C address is 97 (0x61).

n = any number 1 – 127

I2C,100



Factory reset

Command syntax

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

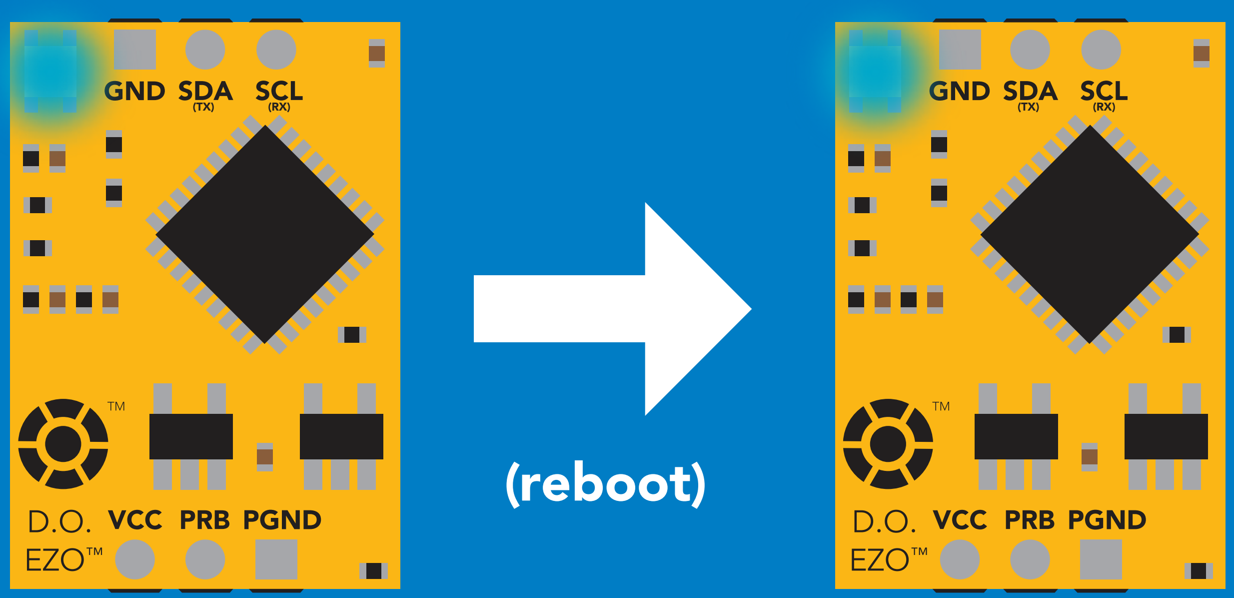
Factory enable factory reset

I²C address will not change

Example	Response
Factory	device reboot (no response given)

Clears calibration
LED on
Response codes enabled

Factory



AtlasScientific Environmental Robotics

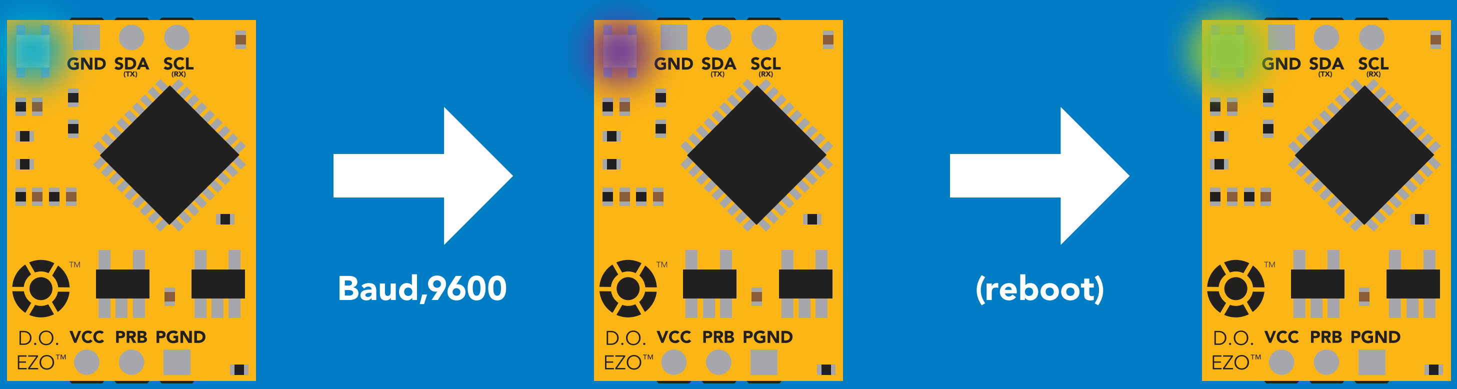
Change to UART mode

Command syntax

Baud,n switch from I^{2}C to UART

Example	Response
Baud,9600	reboot in UART mode (no response given)

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

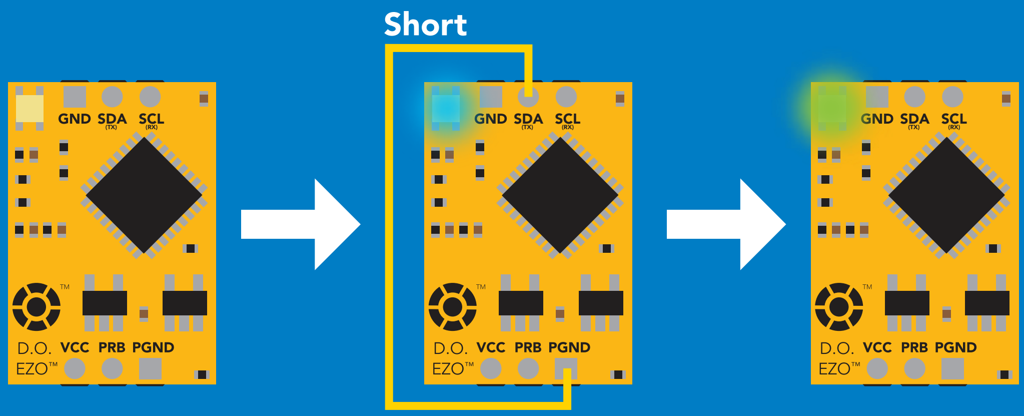


Changing to UART mode

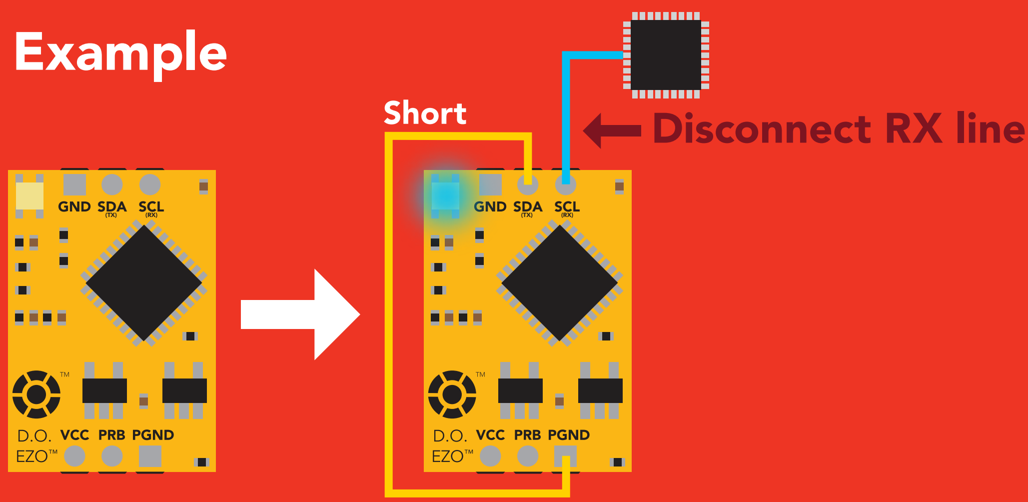
Manual switching to UART

- 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).

Confirm the D.O. probe is working correctly

Take readings in air first.

Readings > 10

✓

Readings < 5 or > 25

Refer to probes datasheet for instructions.

Calibrate first, compensate later

Compensating for temperature, pressure, and salinity will change your calibrated readings to a value that cannot easily be predicted. This makes it difficult to know if the probe has been calibrated correctly.

Default compensation values	Known calibration value
Temp = 20 °C **Pressure = 101 kPa** **Salinity = 0**	**9.09 Mg/L**
Temp = 29 °C **Pressure = 93 kPa** **Salinity = 5** *(too many variables)*	**???** **(6.84 Mg/L)**

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! ⚠

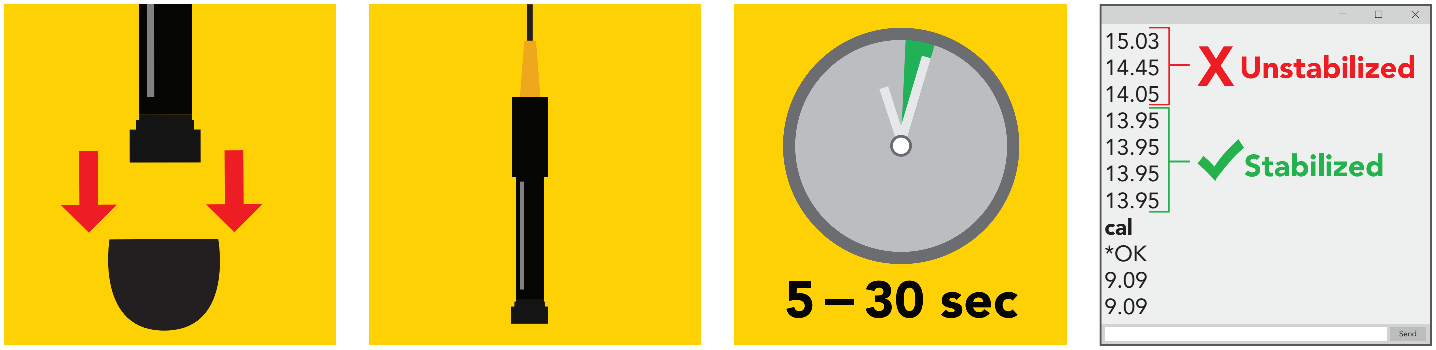
Issuing a calibration command before the readings stabilize will result in drifting readings.

AtlasScientific Environmental Robotics

Calibration order

High point calibration

Remove the Dissolved Oxygen probe's cap and let the probe sit, exposed to air until the readings stabilize. *(small movement from one reading to the next is normal).*

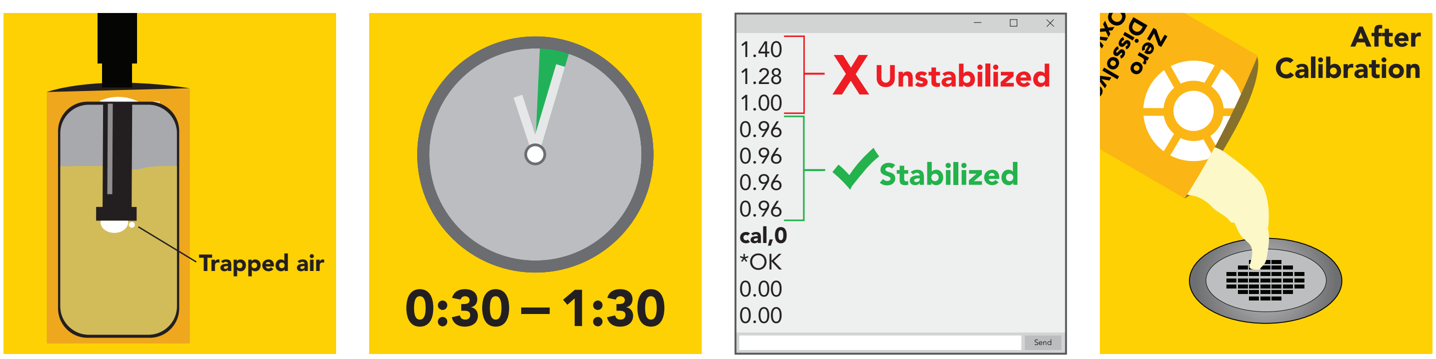


After calibration is complete, you should see readings between **9.09 – 9.1X mg/L.**
(only if temperature, salinity and pressure compensation are at default values)

High point calibrated

Low point calibration

After you have calibrated the EZO™ Dissolved Oxygen circuit using the "Cal" command; Remove the top of the Zero Dissolved Oxygen calibration solution pouch, and Insert the probe and stir it around to remove any trapped air (which could cause readings to go high). Let the probe sit in Zero D.O. calibration solution until readings stabilize.
(small movement from one reading to the next is normal).



Low point calibrated

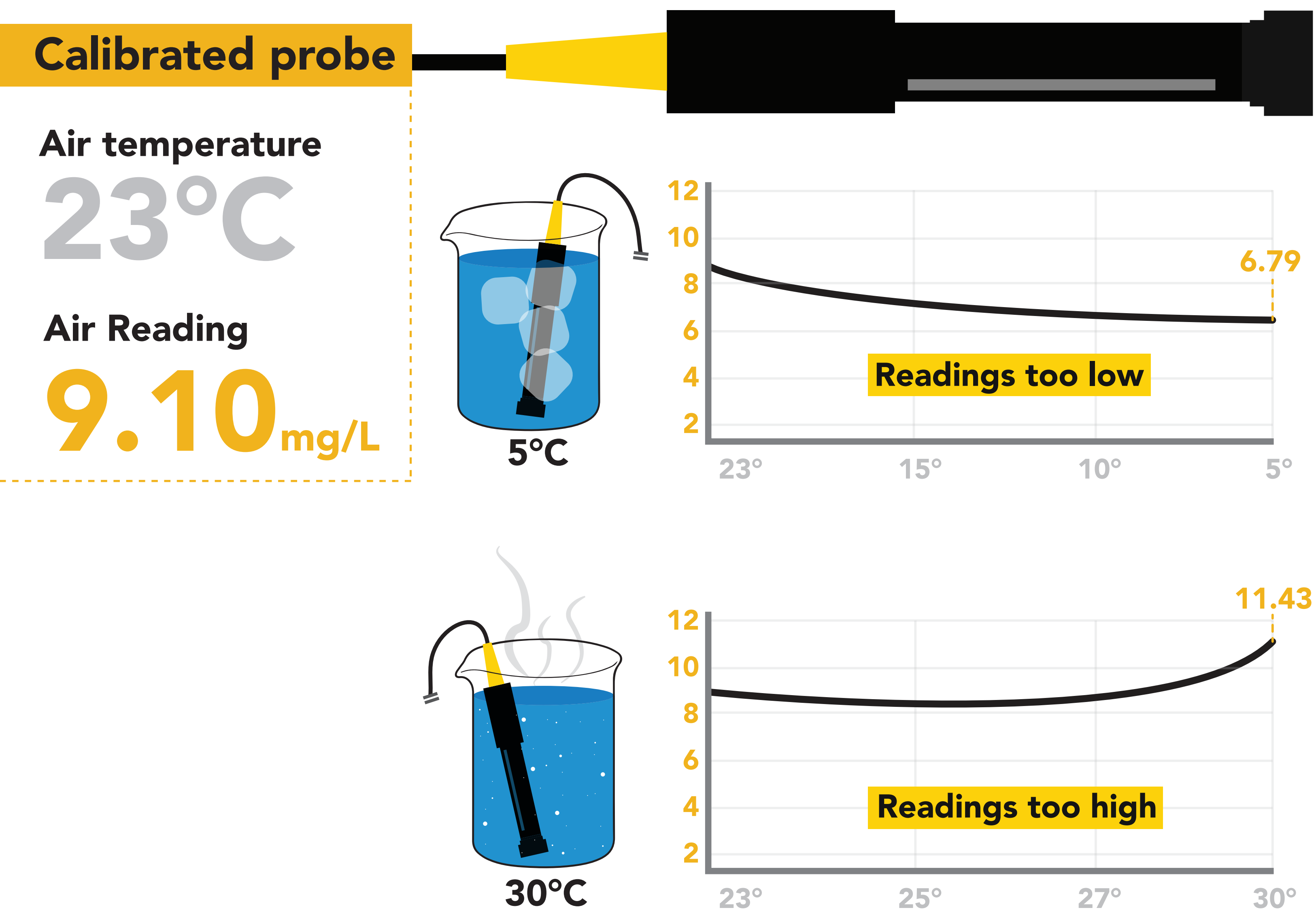
AtlasScientific™
Environmental Robotics

Advanced calibration

Probe temperature calibration

Probe temperature calibration ≠ Temperature compensation.

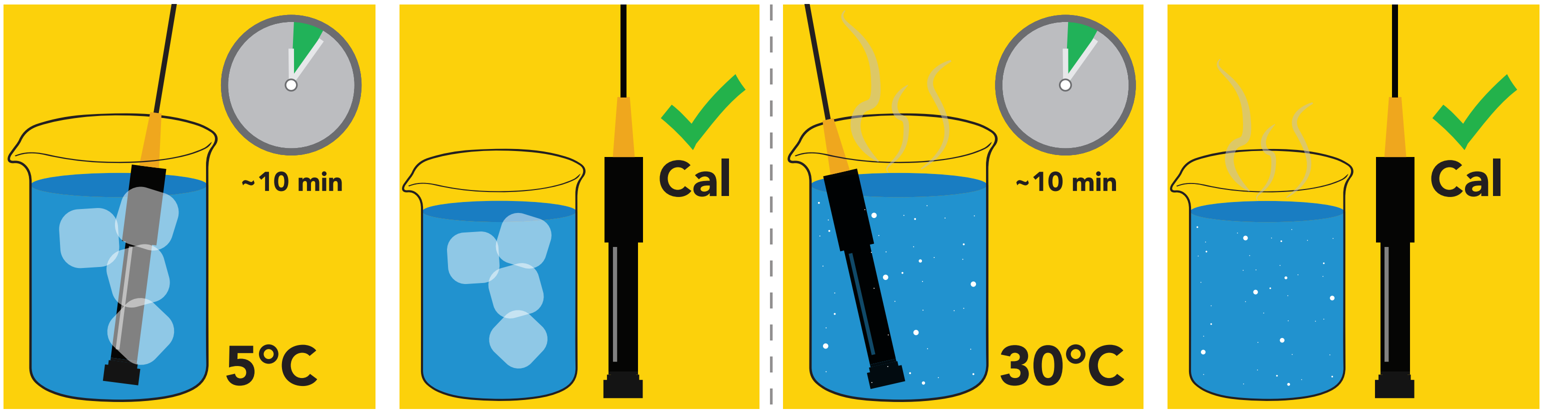
When a Dissolved Oxygen probe is calibrated, it is calibrated to the oxygen level and ambient temperature. As a D.O. probe is heated or cooled, its response curve will change. A small temperature change (≤ 5 °C) will not affect the probe. However, a large temperature change will be noticeable.



Advanced calibration

What to do:

After the Dissolved Oxygen probe has been properly calibrated, another calibration can be done to account for the probe temperature.

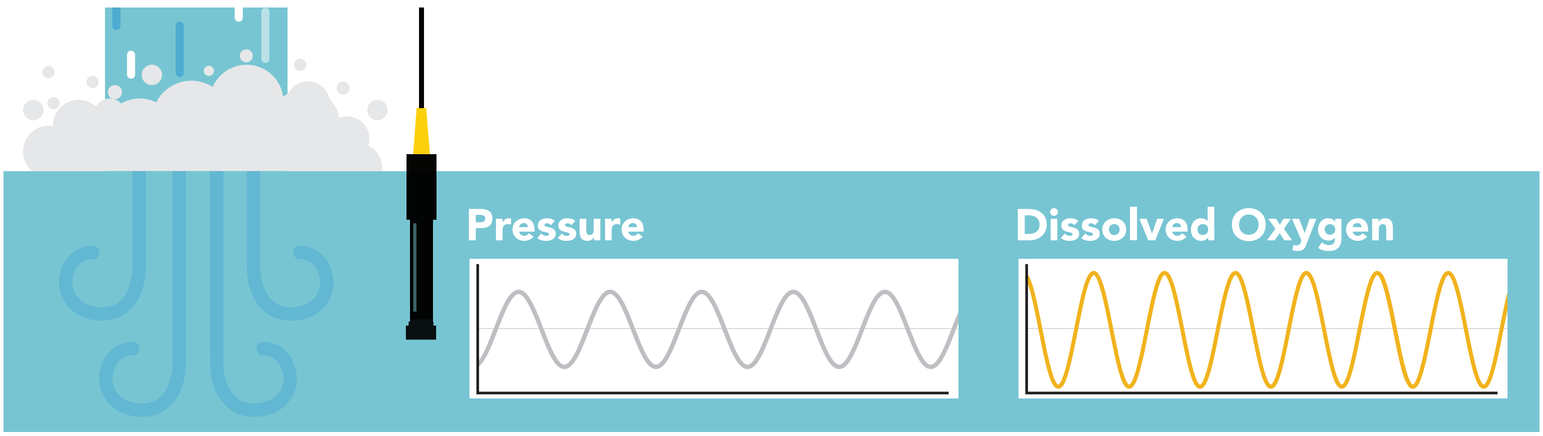


Let the probe acclimate to its operating temperature and then recalibrate. Once the probe has been calibrated at its intended operating temperature, using temperature compensation will give accurate readings.

Understanding D.O. measurements

Most chemical sensors do not directly measure the parameter they are designed for. Dissolved oxygen is no exception. A galvanic D.O. probe is actually an oxygen pressure sensor. It only measures the partial pressure of oxygen.

Keep this in mind when choosing a spot to place the probe.



It just so happens that partial pressure of oxygen is the same in water as it is in air.

(While the pressure is the same, the amount is not. Pure water at sea level can only hold ~9 mg/L of oxygen, while the atmosphere holds ~300mg/L)

By comparing oxygens pressure to its solubility in water, the mg/L are derived.

There are three factors that affect waters ability to hold oxygen.

Temperature
Salinity
Atmospheric Pressure

Temperature

Water temperature has the largest effect; the colder the water, the more oxygen it holds. As water heats up, its ability to hold oxygen goes down.

Pure water at 1°C can hold 14.2 mg/L

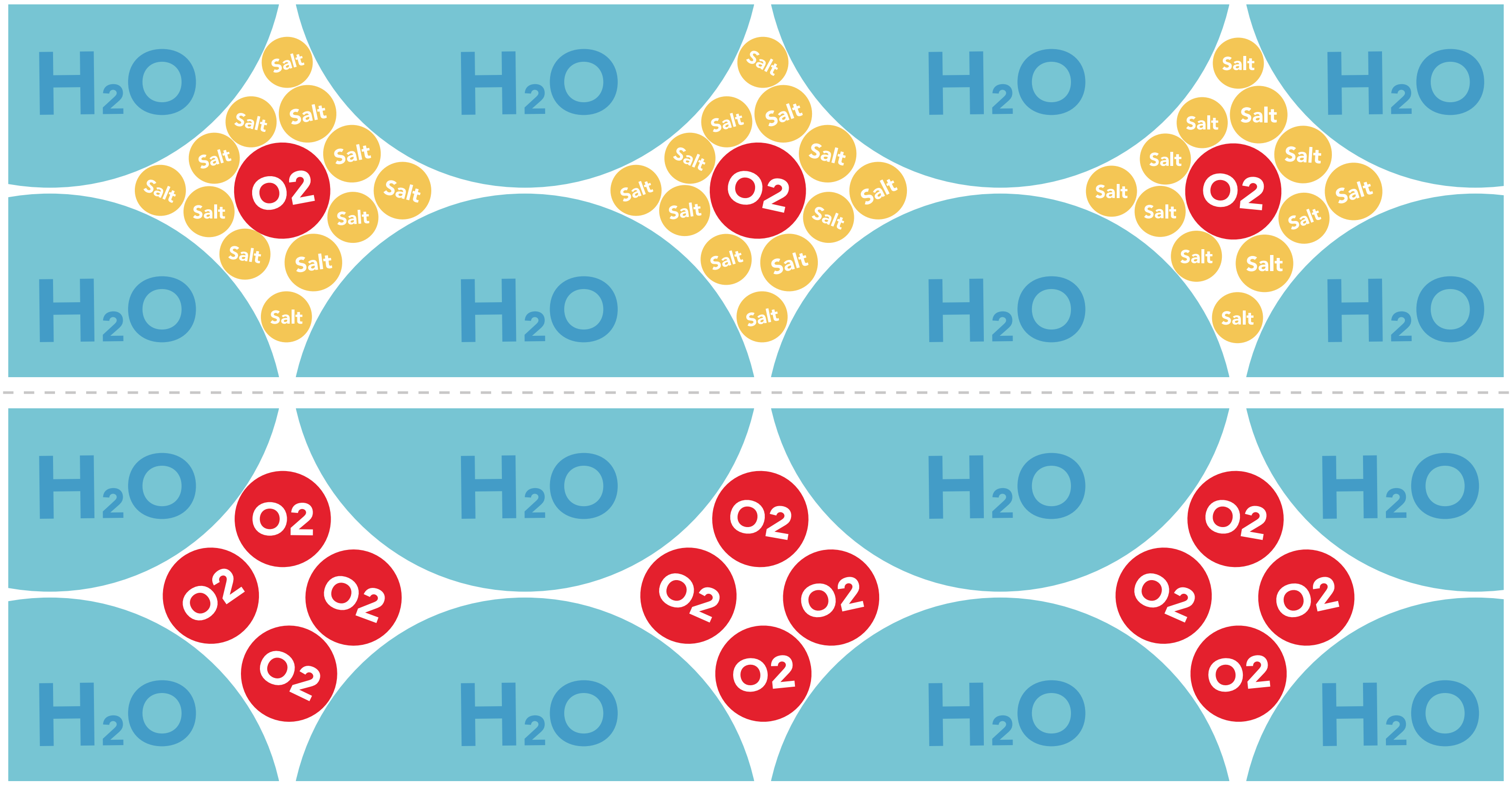
And at 40°C it can only hold 6.4 mg/L

Understanding D.O. measurements

Salinity

When salt is added to water, it drives out oxygen by competing for the same space.

Sea water at 1°C can only hold 10.7 mg/L
Pure water at 1°C can hold 14.2 mg/L



Atmospheric Pressure

A D.O. probe is an oxygen pressure sensor.

Dissolved oxygen pressure cannot be higher than atmospheric oxygen pressure. This is why the probe is calibrated to the atmosphere; it defines the probe's response to the maximum oxygen pressure available. However, oxygen pressure does not tell us how much oxygen is available to dissolve in the water. That information is derived from atmospheric pressure (where atmospheric pressure = altitude).

As altitude increases, oxygen concentration decreases, and because D.O. readings are expressed in Mg/L, the oxygen concentration must be known.

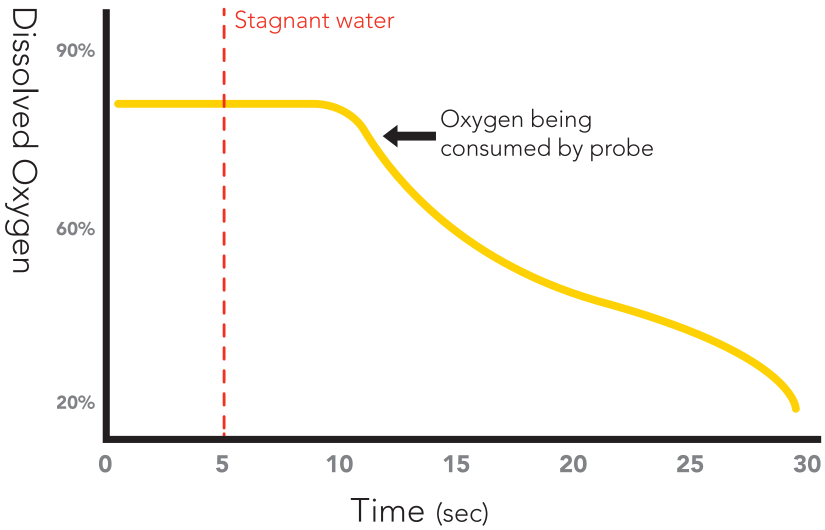
At sea level, 1°C pure water can hold 14.2 mg/L

At 1,500 meters, 1°C pure water can hold 11.7 mg/L

At -1,200 meters, 1°C pure water can hold 16.2 mg/L

AtlasScientific™
Environmental Robotics

Flow Dependence



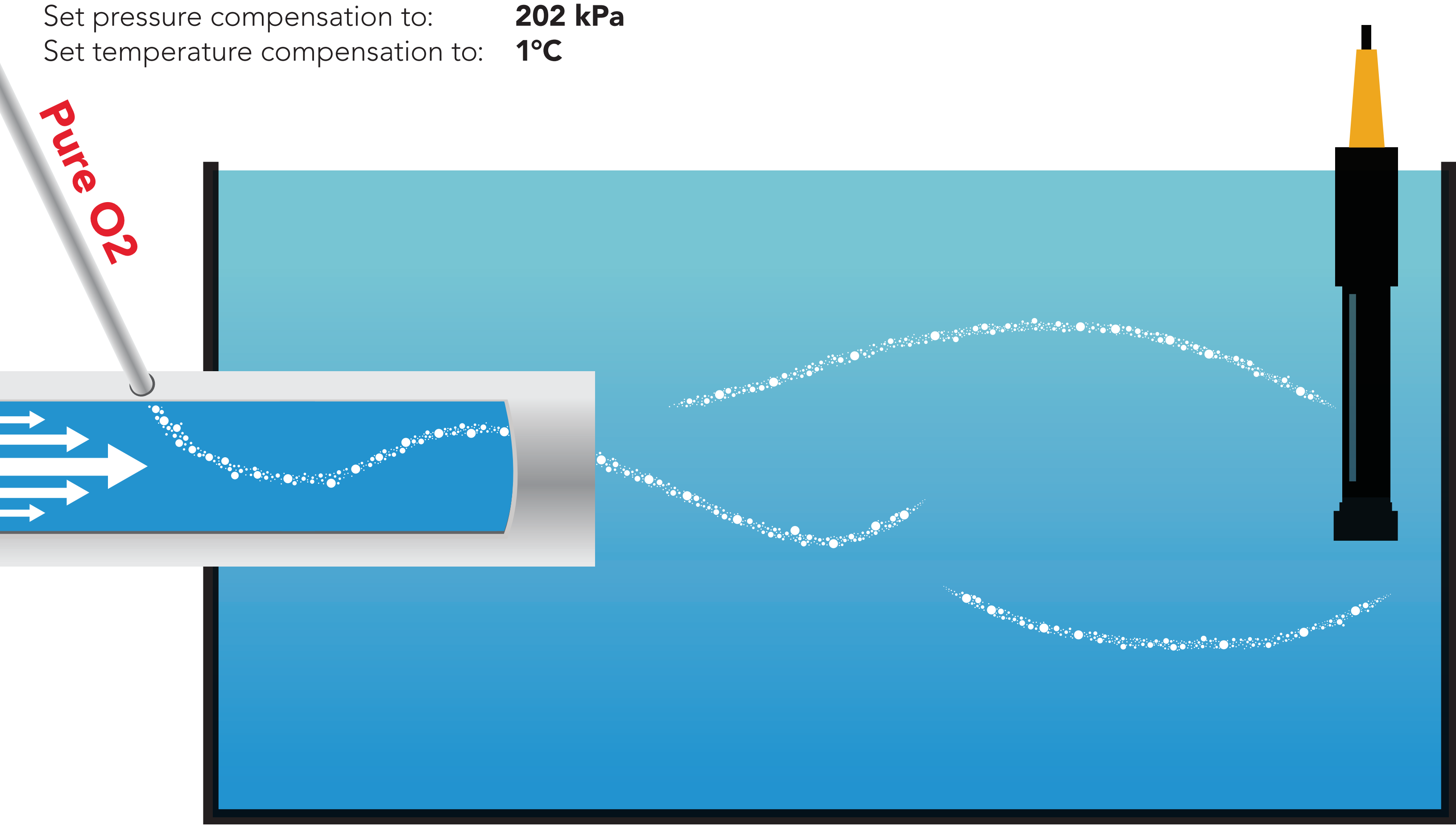
One of the drawbacks from using a galvanic probe is that it consumes a ***VERY*** small amount of the oxygen it reads. Therefore, a small amount of water movement is necessary to take accurate readings. **Approximately 60 ml/min**.

Hyper saturation with pure oxygen

Dissolved oxygen measurements are based on natural occurring oxygen levels. However, some applications may require pure oxygen to achieve extremely high saturation levels. Because injecting pure oxygen into water is not a naturally occurring event, you will need to change some compensation parameters to achieve extremely high readings.

To reach 100mg/L and a saturation of 350%

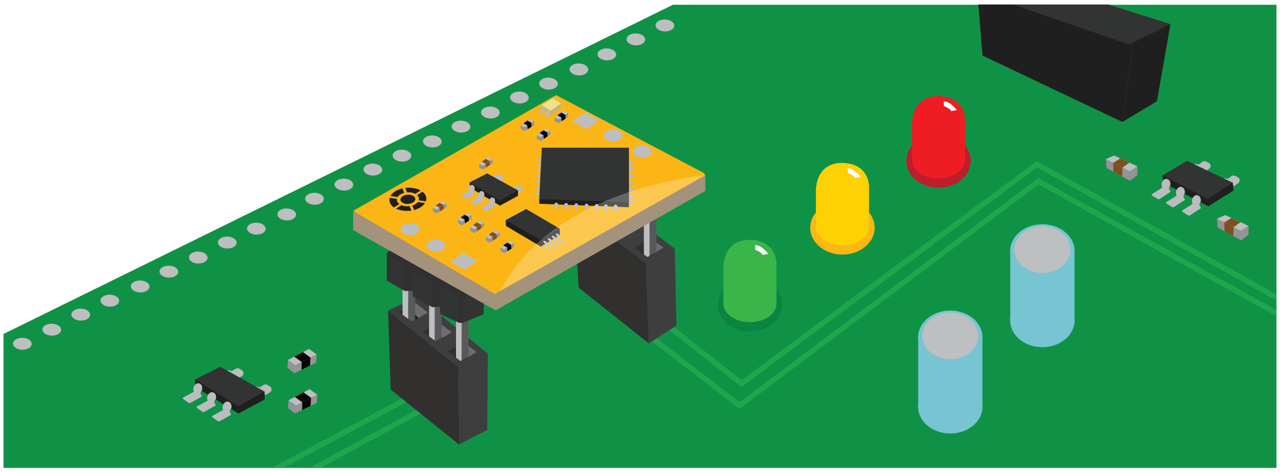
Set pressure compensation to: **202 kPa**
Set temperature compensation to: **1°C**



Soldering

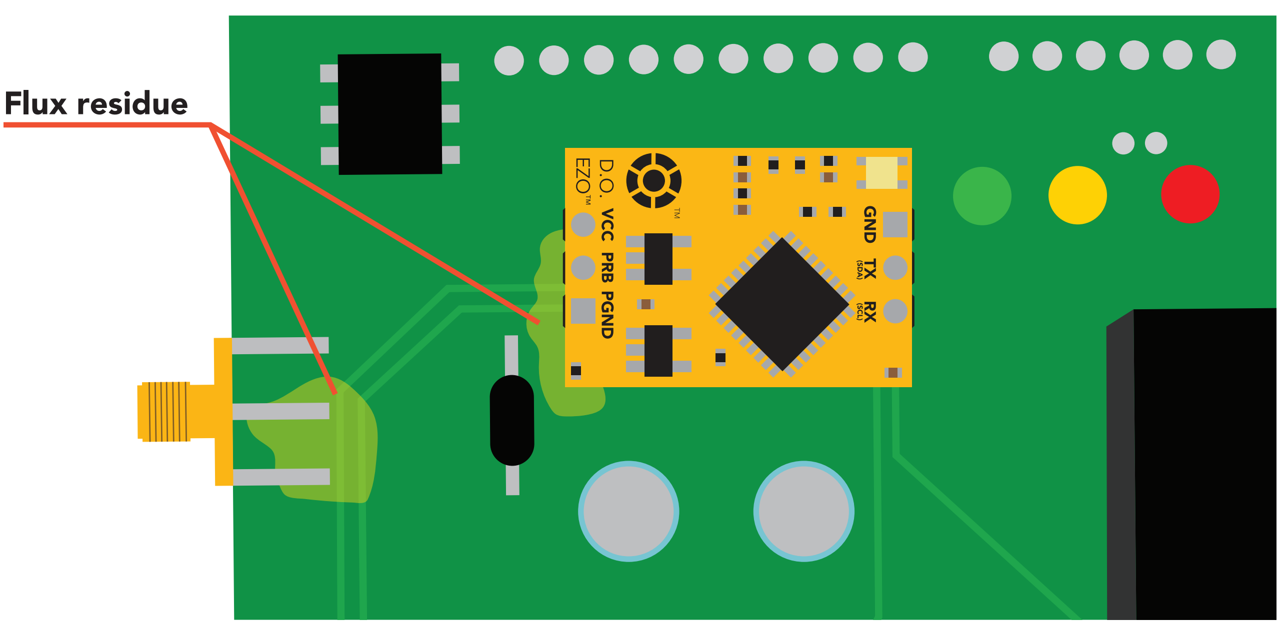
Do not directly solder an EZO circuit to your PCB. If something goes wrong during the soldering process it may become impossible to correct the problem. It is simply not worth the risk.

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.

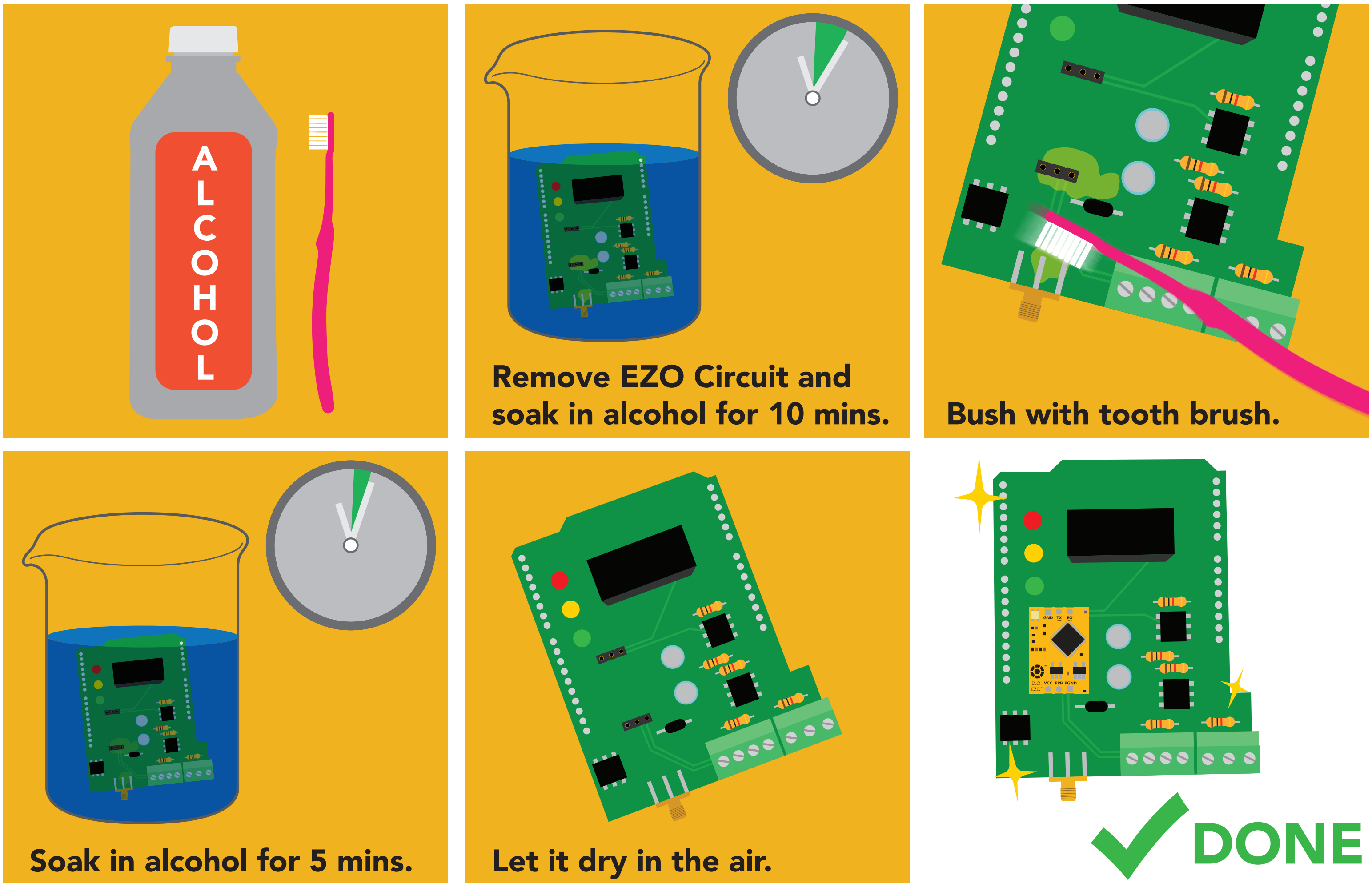
Flux residue will severely affect your readings. Any Flux residue that comes in contact with the PRB pins or your probes connector will cause a “flux short”.



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

Soldering

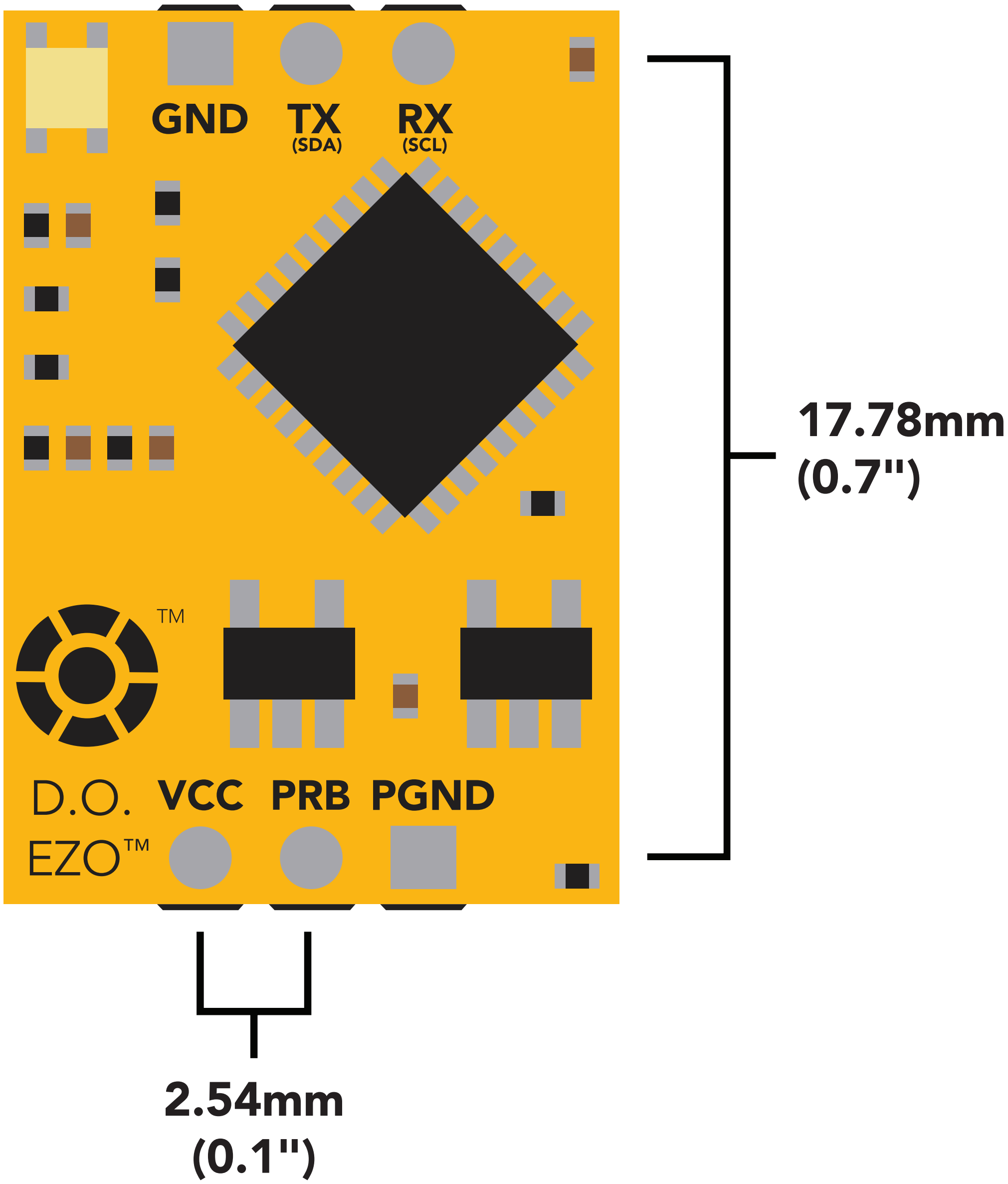
Removing flux residue can be done with commercially available products such as flux off or you can use alcohol and a tooth brush.



What does a flux short look like?

Readings move slowly and take serval minutes to reach the correct value.

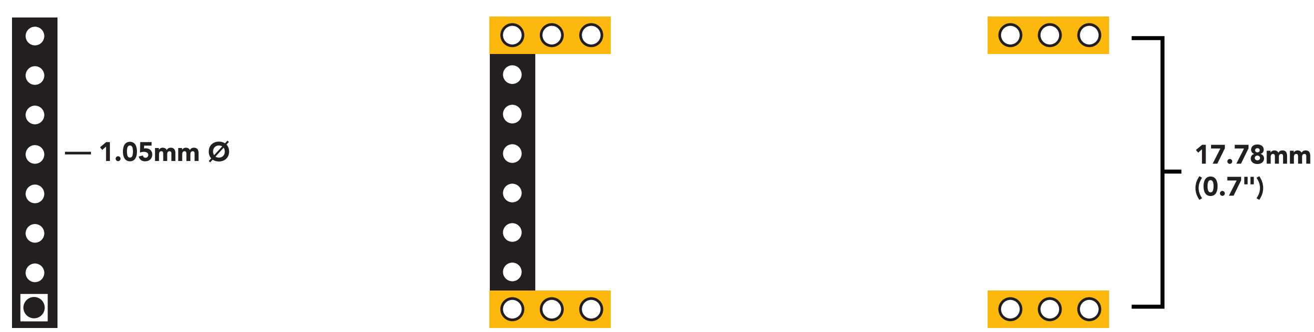
EZO™ circuit footprint



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 5.6

Revised enitre document.

Datasheet V 5.5

Revised naming device info on pages 32 & 59.

Datasheet V 5.4

Revised artwork within datasheet.

Datasheet V 5.3

Moved Default state to pg 13.

Datasheet V 5.2

Updated firmware changes on page 70.

Datasheet V 5.1

Revised response for the sleep command in UART mode on pg 36.

Datasheet V 5.0

Revised calibration theory on page 9, and added more information on the Export calibration and Import calibration commands.

Datasheet V 4.9

Corrected temperature compensation typo on pages 26 & 52.

Datasheet V 4.8

Revised isolation schematic on pg. 10

Datasheet V 4.7

Added new command:
"RT,n" for Temperature compensation located on pages 26 (UART) & 52 (I^2C).
Added firmware information to Firmware update list.

Datasheet change log

Datasheet V 4.6

Added more information about temperature compensation on pages 26 & 52.

Datasheet V 4.5

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

Datasheet V 4.4

Removed note from certain commands about firmware version.

Datasheet V 4.3

Added information to calibration theory on pg 7.

Datasheet V 4.2

Revised definition of response codes on pg 44.

Datasheet V 4.1

Updated firmware changes on pg. 66.

Datasheet V 4.0

Revised Enable/disable parameters information on pages 29 (UART) & 55 (I^2C).

Datasheet V 3.9

Revised information on cover page.

Datasheet V 3.8

Update firmware changes on pg. 66.

Datasheet V 3.7

Revised Plock pages to show default value.

Datasheet change log

Datasheet V 3.6

Added new commands:
"Find" pages 21 (UART) & 48 (I^2C).
"Export/Import calibration" pages 25 (UART) & 51 (I^2C).
Added new feature to continous mode "C,n" pg 22.

Datasheet V 3.5

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

Datasheet V 3.4

Added manual switching to UART information on pg. 59.

Datasheet V 3.3

Updated firmware changes to refeclt V1.99 update.

Datasheet V 3.2

Revised entire datasheet.

Firmware updates

V1.1 – Initial release (Oct 30, 2014)
• Change output to mg/L, then percentage (was previously percentage, then mg/L).

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

V1.6 – I^2C bug (Dec 1, 2014)
• Fixed I^2C bug where the circuit may inappropriately respond when other I^2C 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)
• Fixed bug in the cal clear command, improves how it calculates the DO, adds calibration saving and loading.

V1.98 – EEPROM (Nov 14, 2016)
• Updated firmware for new circuit design.

V1.99 – (Feb 2, 2017)
• Revised "O" command to accept mg.

V2.10 – (April 12, 2017)
• Added "Find" command.
• Added "Export/import" command.
• Modified continuous mode to be able to send readings every "n" seconds.

V2.11 – (Sept 28, 2017)
• Fixed bug where the temperature would default to 0 on startup.

V2.12 – (Dec 19, 2017)
• Improved accuracy of dissolved oxygen equations.

V2.13 – (July 16, 2018)
• Added "RT" command to Temperature compensation.

V2.14 – (June 7, 2019)
• Fixed bug where the output buffer overflows when the cal and cal,0 point are too close together.

Firmware updates

V2.15 – (Sept 8, 2022)
• Internal update for new part compatibility.

Warranty

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

The debugging phase

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

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

