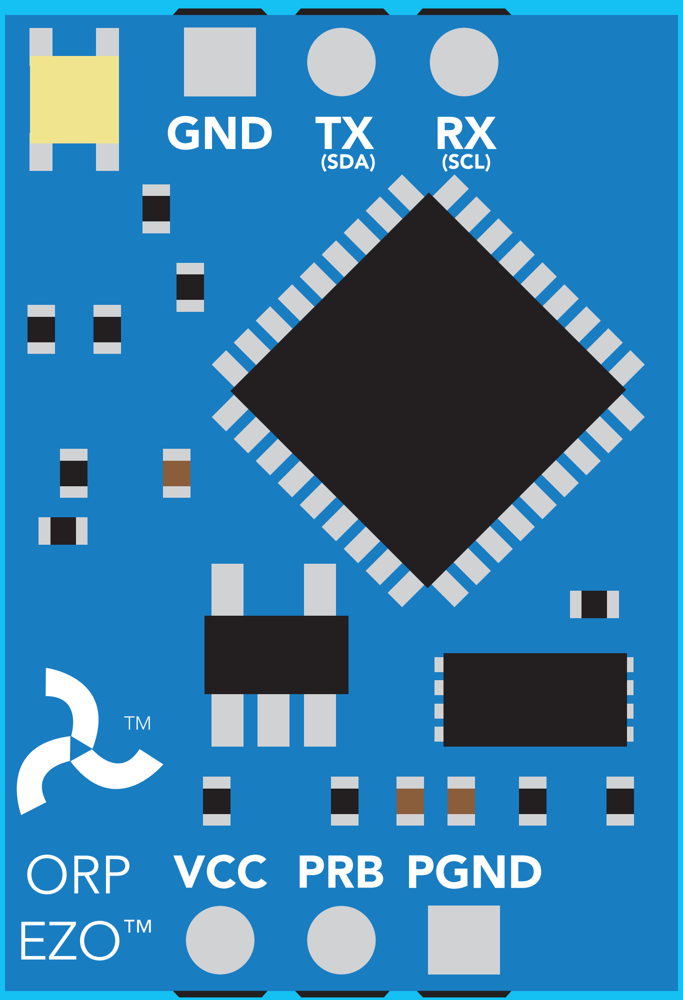
AtlasScientific™
Environmental Robotics

V 4.0

Revised 2/16/18

ORP EZO™ Circuit

Reads	**ORP**
Range	**-1019.9mV – 1019.9mV**
Accuracy	**+/– 1mV**
Max rate	**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.

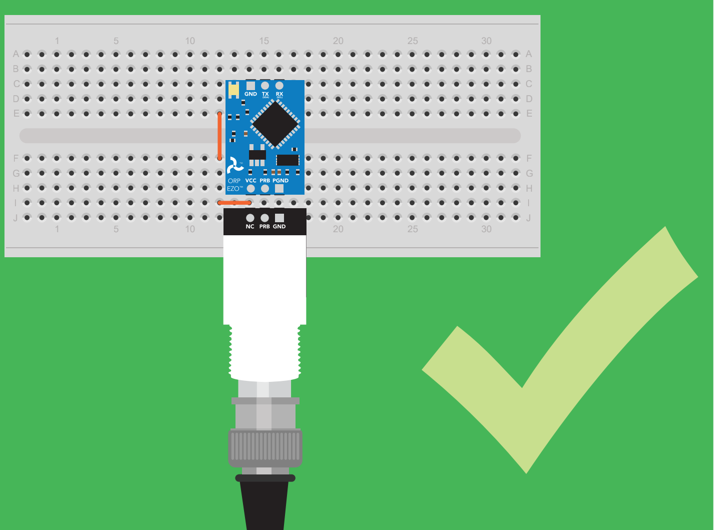
STOP

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!

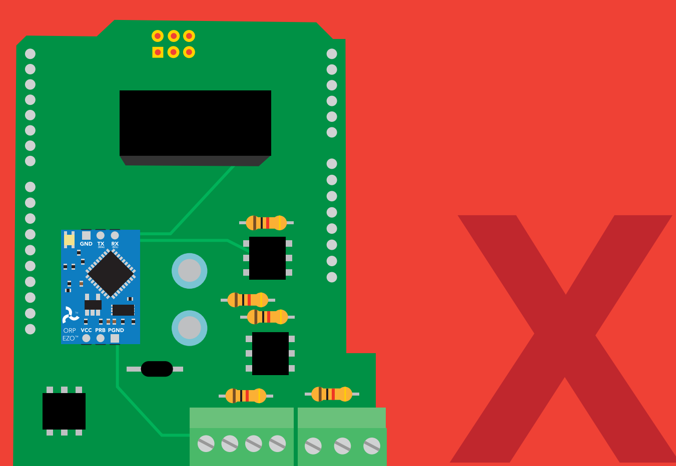


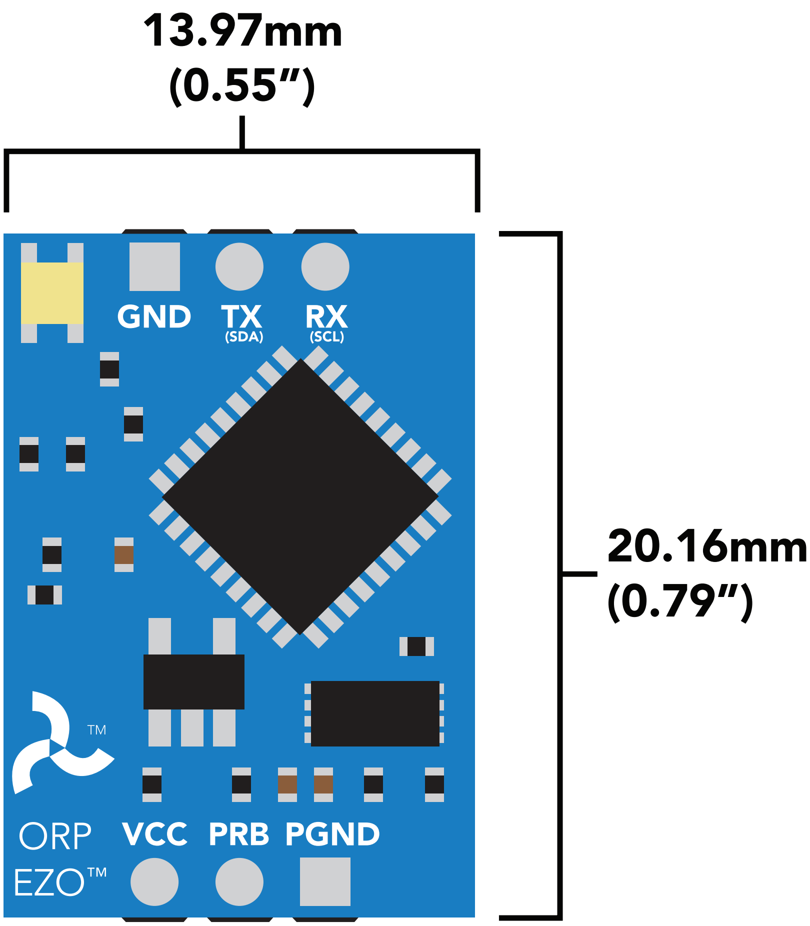
Table of contents

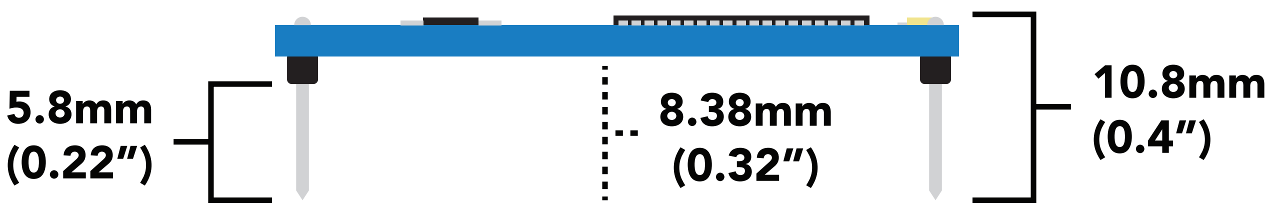
UART

I²C

AtlasScientific™ Environmental Robotics

EZO™ circuit dimensions





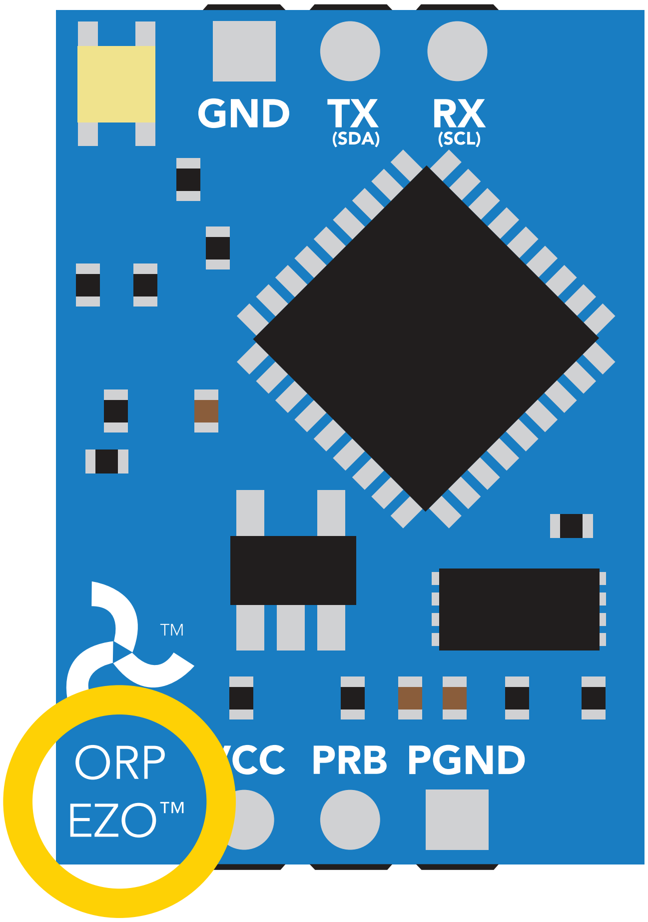
Power consumption

	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	

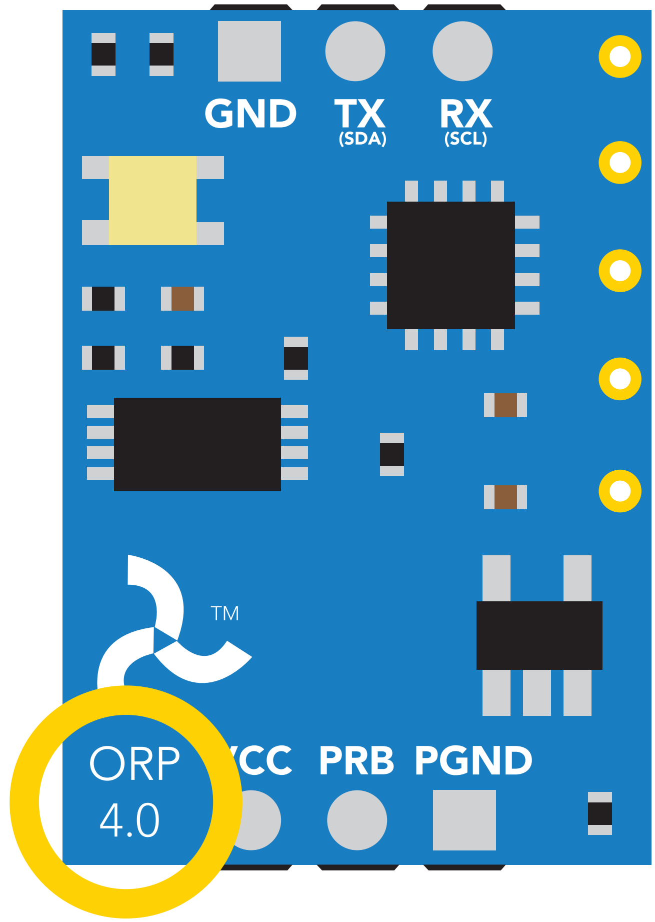
Absolute max ratings

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

EZO™ circuit identification



EZO™ ORP circuit



Legacy ORP circuit

Viewing correct datasheet

Viewing incorrect datasheet

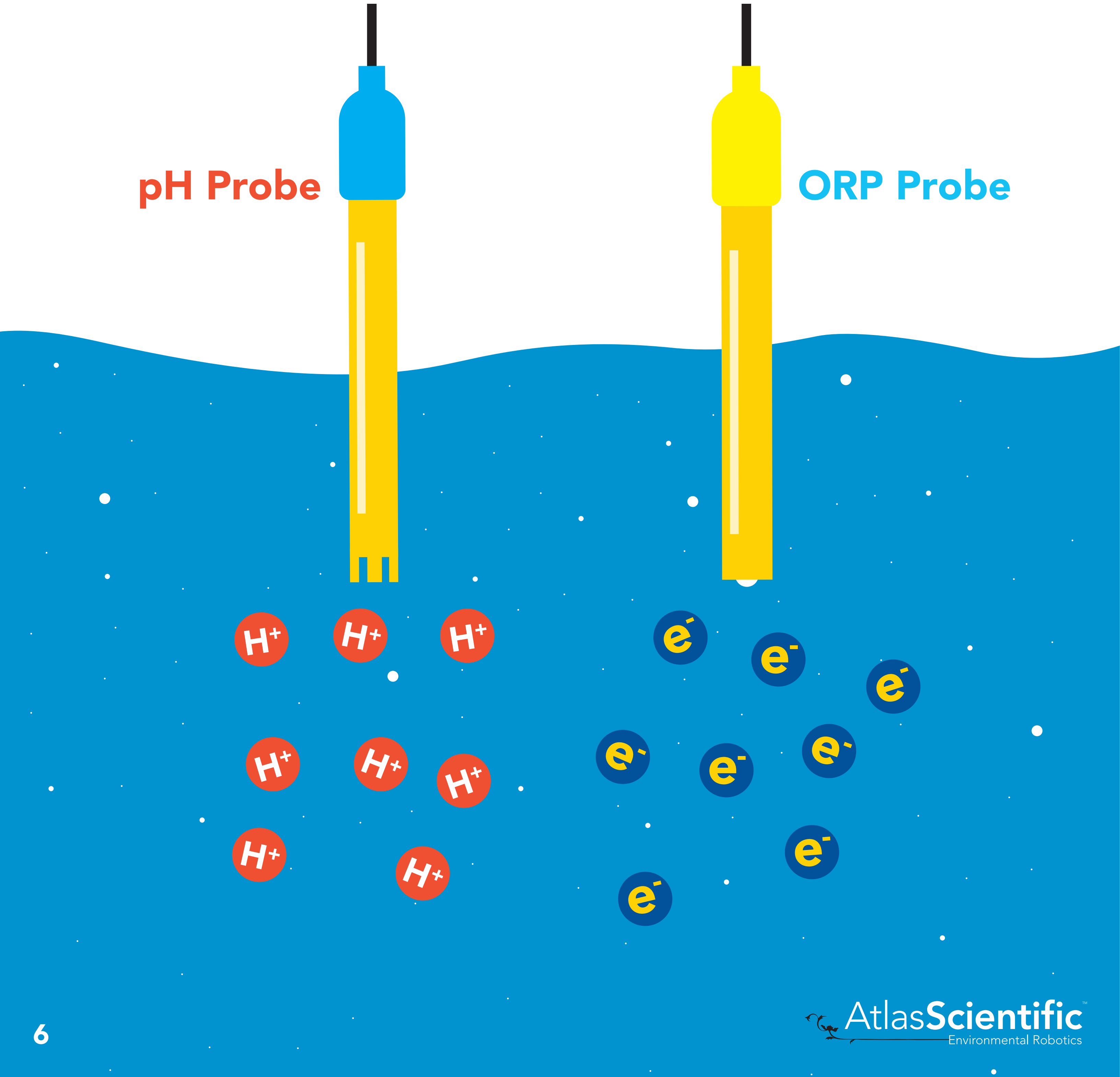
Click here to view legacy datasheet

Atlas Scientific™
Environmental Robotics
r 0.1

Operating principle

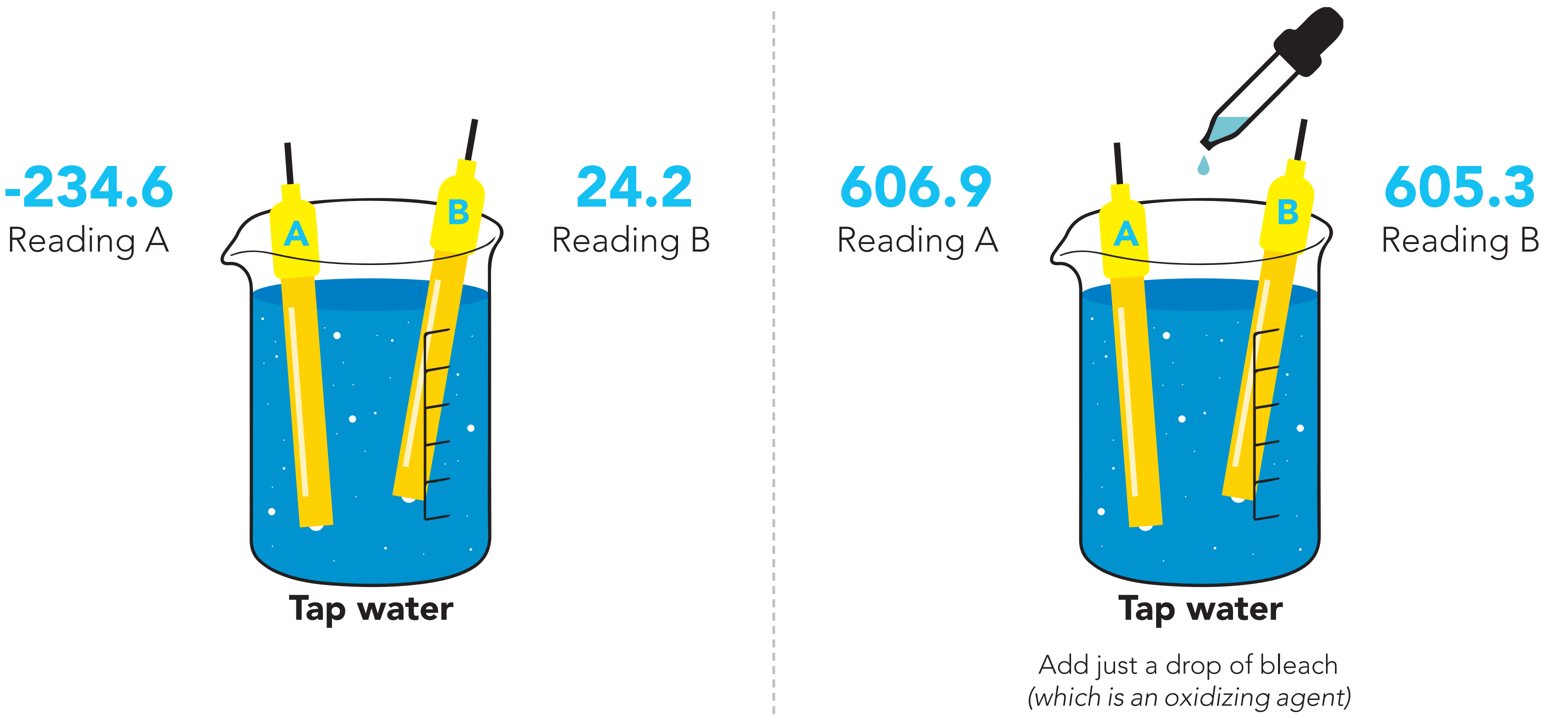
ORP stands for ***oxidation/reduction potential***. Oxidation is the loss of electrons and reduction is the gain of electrons. The output of the probe is represented in millivolts and can be positive or negative.

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.

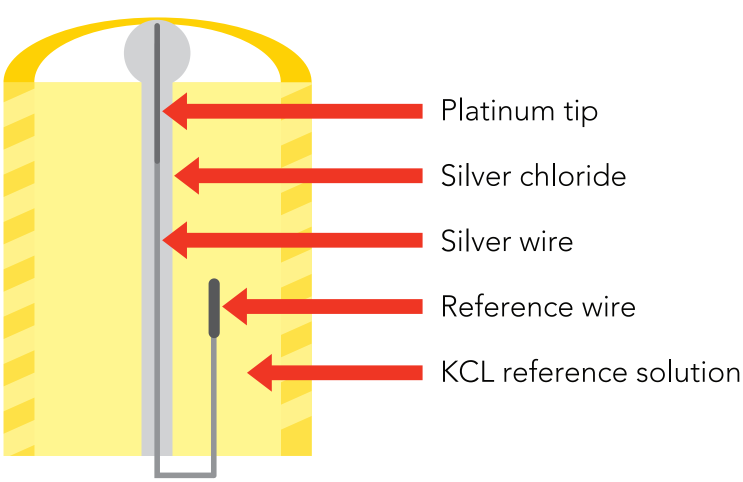


When reading the ORP of a liquid that has very few electrons available for transfer ORP readings can appear to be inconsistent.

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.

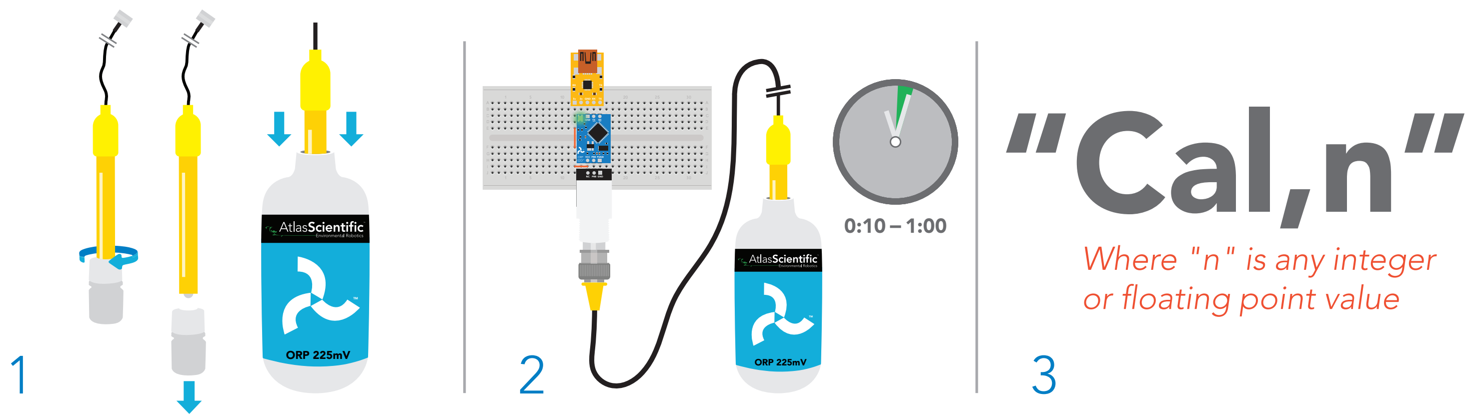


Calibration theory

Calibration should be done at least once per year.
If the ORP that's being read is continuously on the extremes of the scale *(around -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.

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

Single point calibration



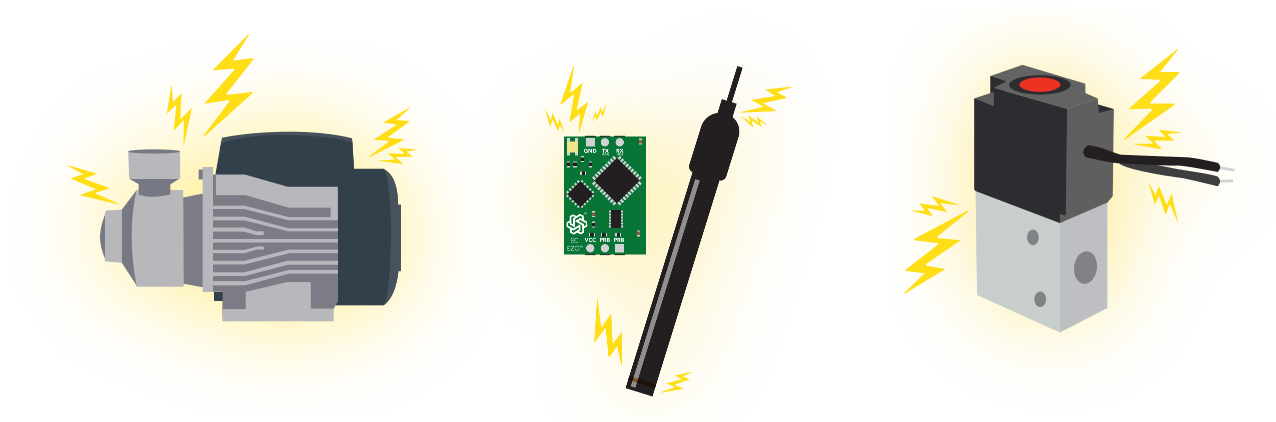
1. Remove soaker bottle and place probe in ORP calibration solution.
2. Let the probe sit in calibration solution untill readings stabalize *(10 – 60 seconds).*
3. Calibrate to the value of the calibration solution using the command "Cal,n".

(If you are using the Atlas Scientific ORP calibration solution, calibrate to 225mV; "Cal,225").

Power and data isolation

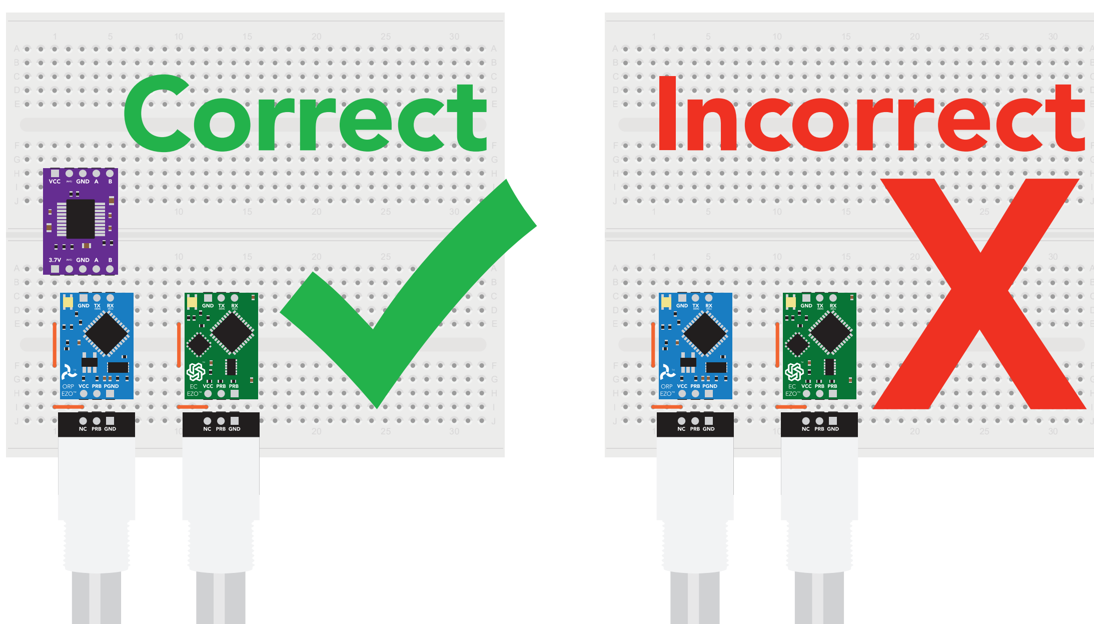
The Atlas Scientific EZO™ ORP circuit is a very sensitive device. This sensitivity is what gives the ORP circuit its accuracy. This also means that the ORP circuit is capable of reading micro-voltages that are bleeding into the water from unnatural sources such as pumps, solenoid valves or other probes/sensors.

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.

Basic EZO™
Inline Voltage Isolator



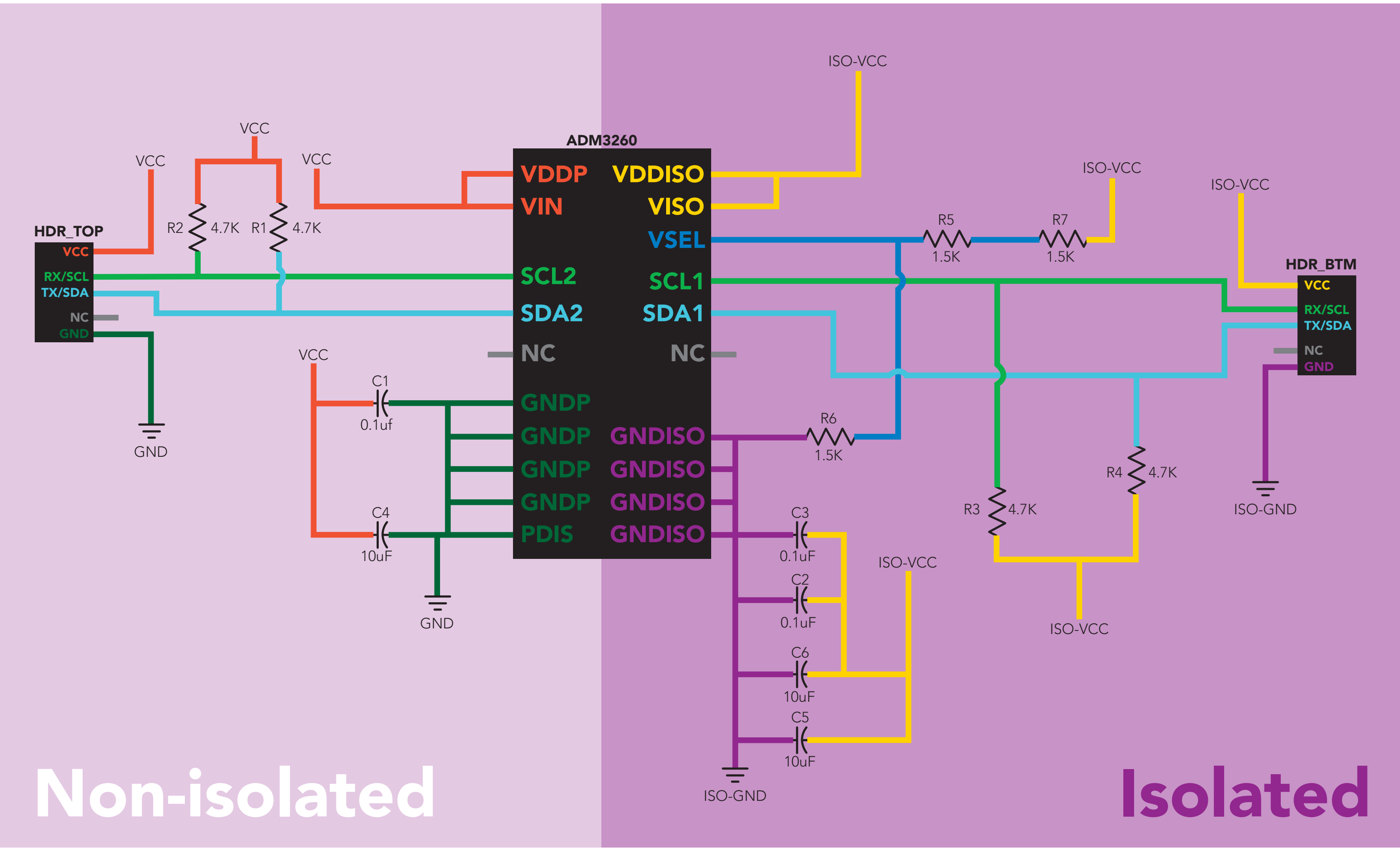
Without isolation, Conductivity and Dissolved Oxygen readings will effect ORP accuracy.

r 0.1

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

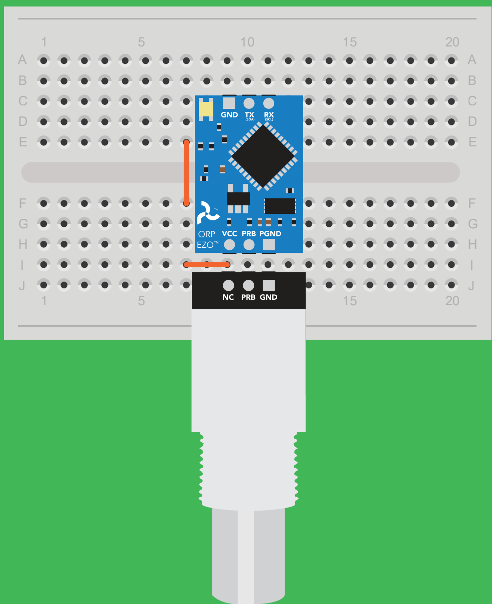
This technology works by using tiny transformers to induce the voltage across an air gap. PCB layout requires special attention for EMI/EMC and RF Control, having proper ground planes and keeping the capacitors as close to the chip as possible are crucial for proper performance. The two data channels 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 R,7) this produces a voltage of 3.7V regardless of your input voltage.

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

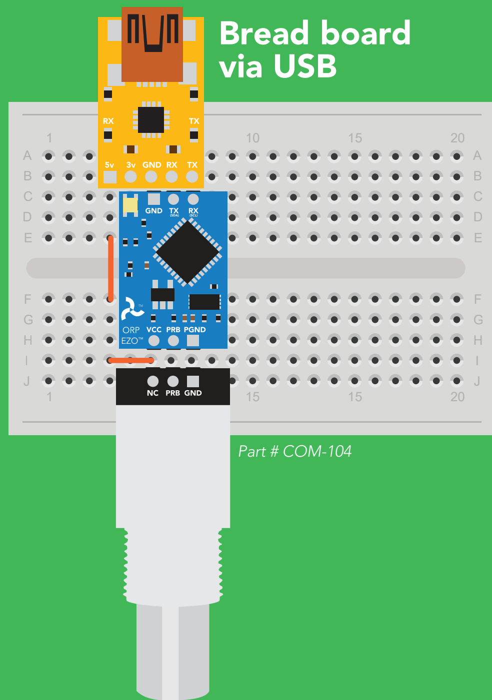


Correct wiring

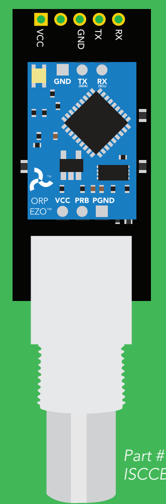
Bread board



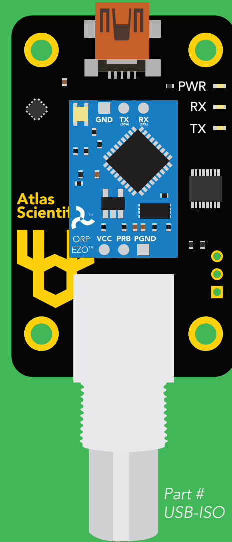
Bread board via USB



Carrier board



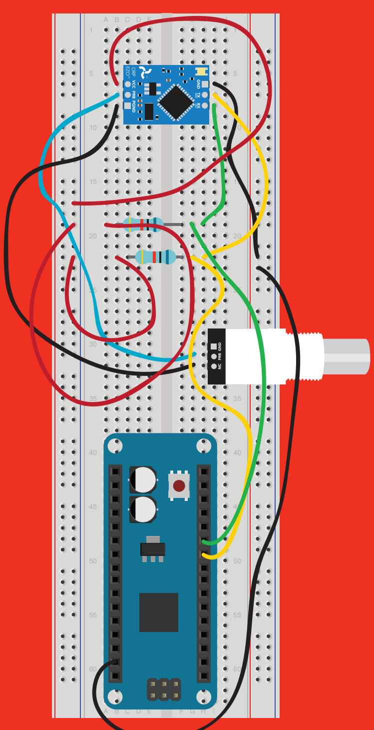
USB carrier board



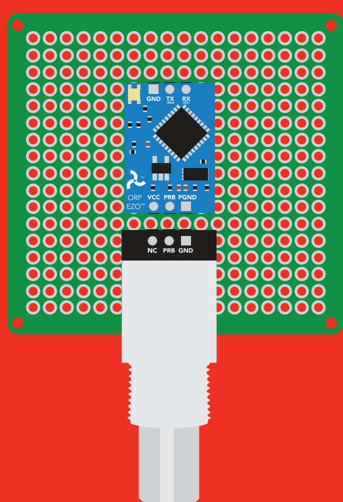
Incorrect wiring

Extended leads

Sloppy setup

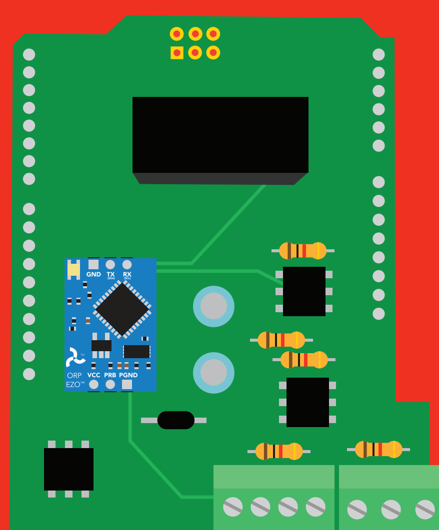


Perfboards or Protoboards



NEVER
use Perfboards or Protoboards

*Embedded into your device

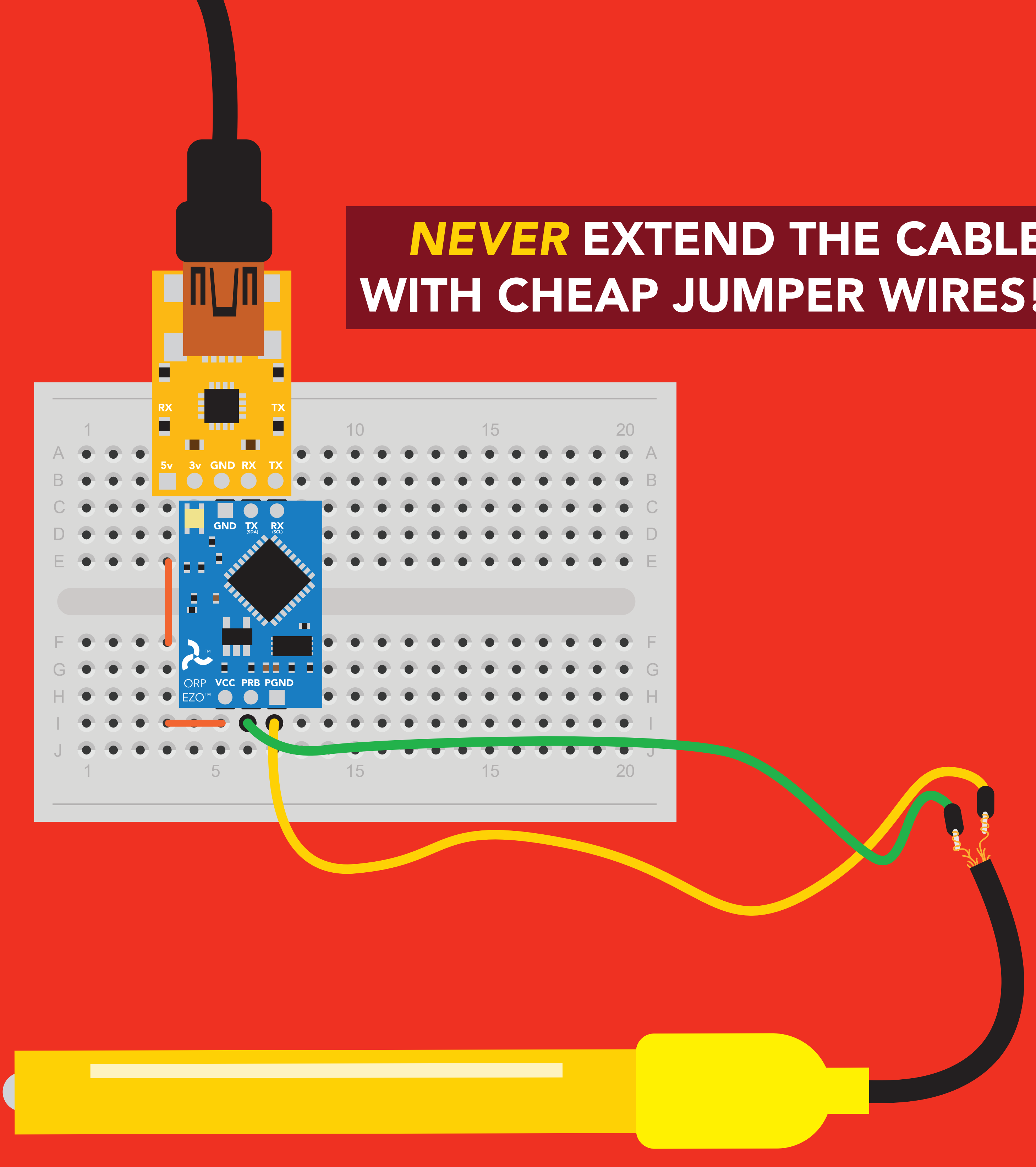


***Only after you are familar with EZO™ circuits operation**

AtlasScientific™
Environmental Robotics

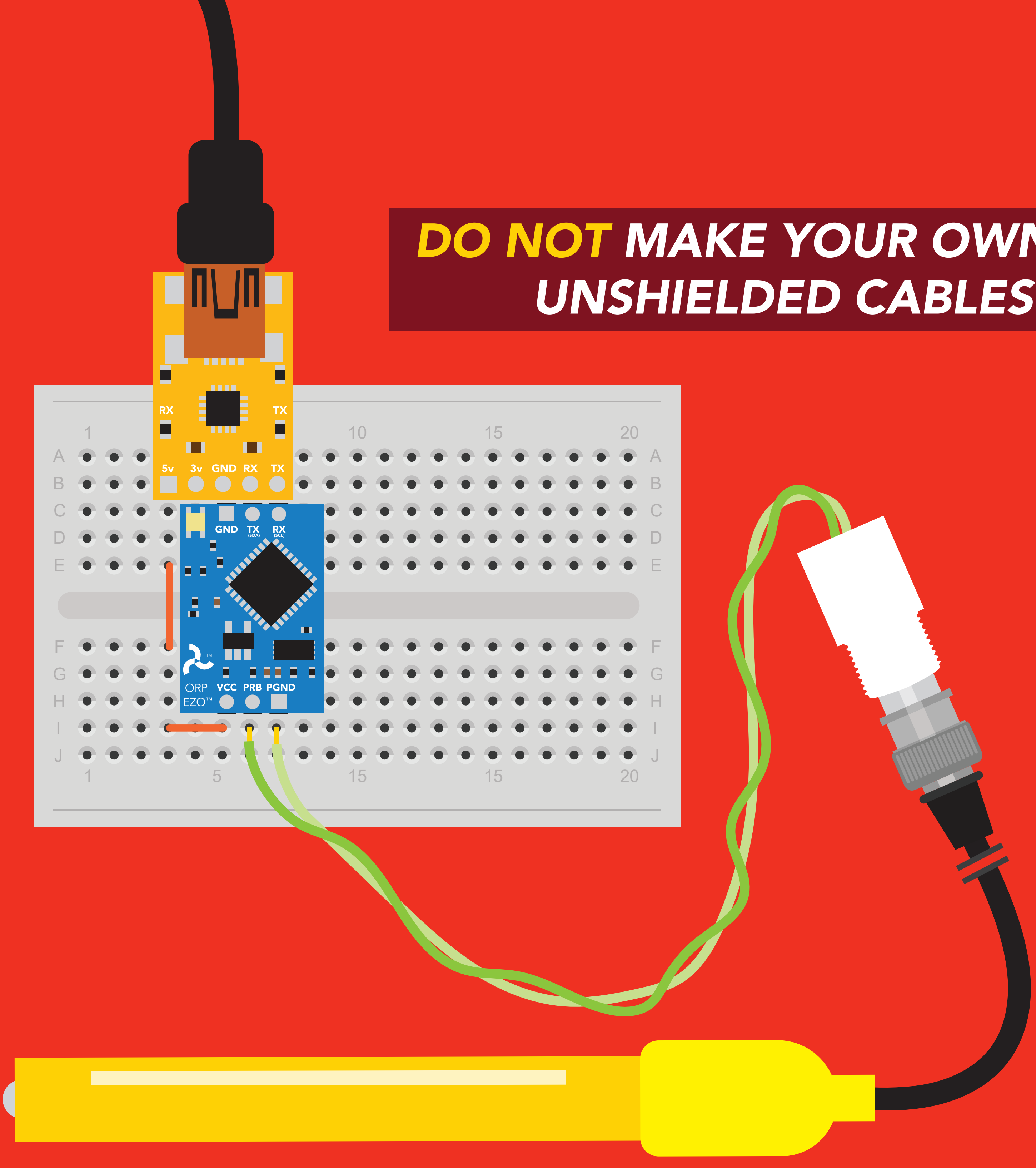
r 0.1

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!*

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

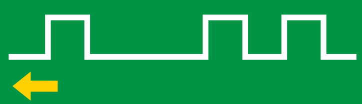
AtlasScientific
Environmental Robotics

UART mode

8 data bits — no parity
1 stop bit — no flow control

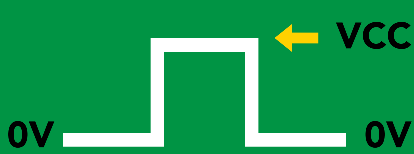
Baud
300
1,200
2,400
9,600 default
19,200
38,400
57,600
115,200

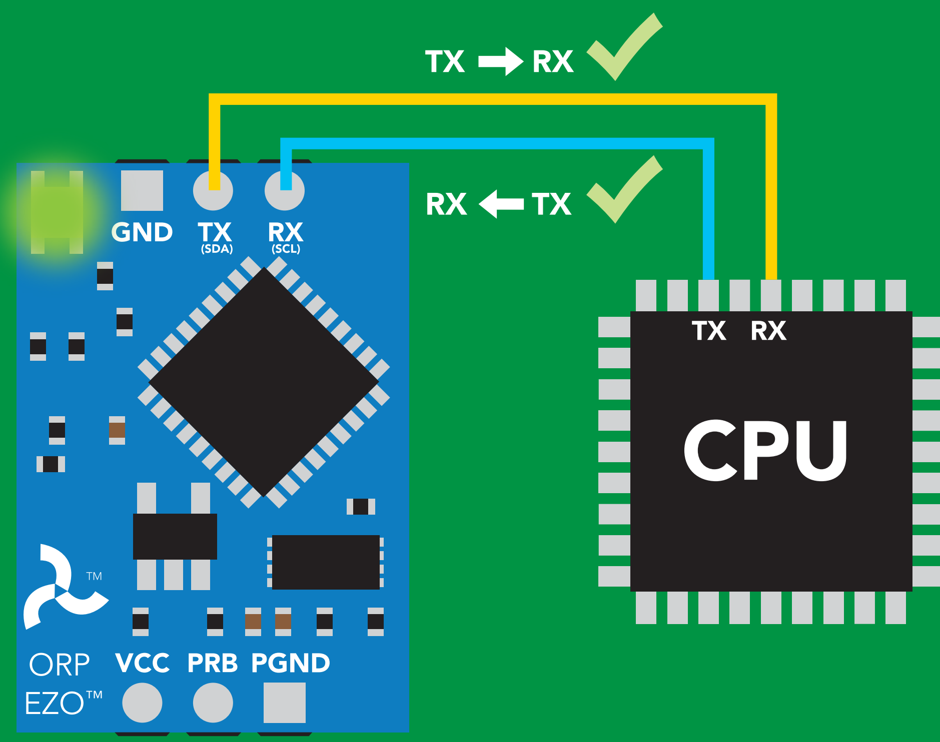
RX
Data in



TX
Data out

Vcc 3.3V – 5.5V



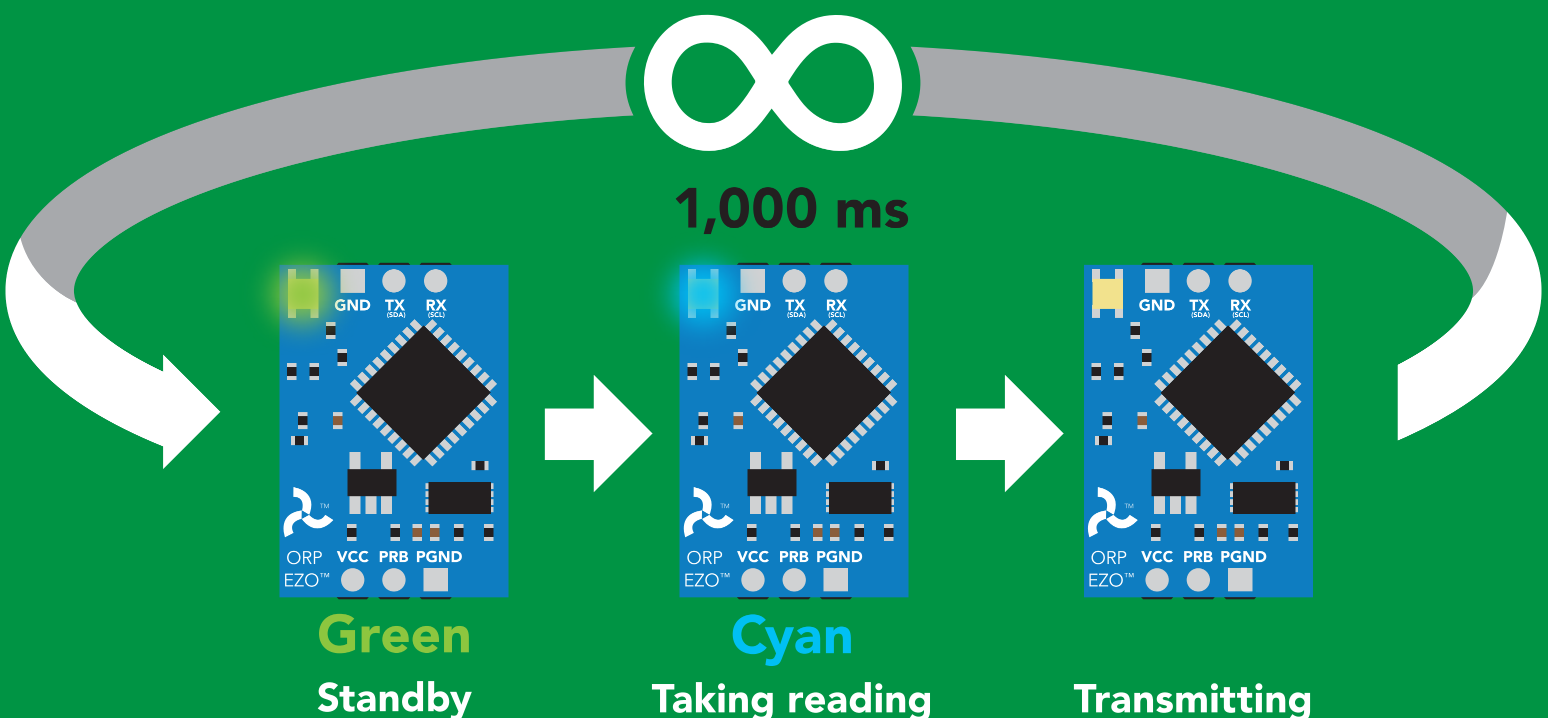


Data format

Reading	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		

Default state

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

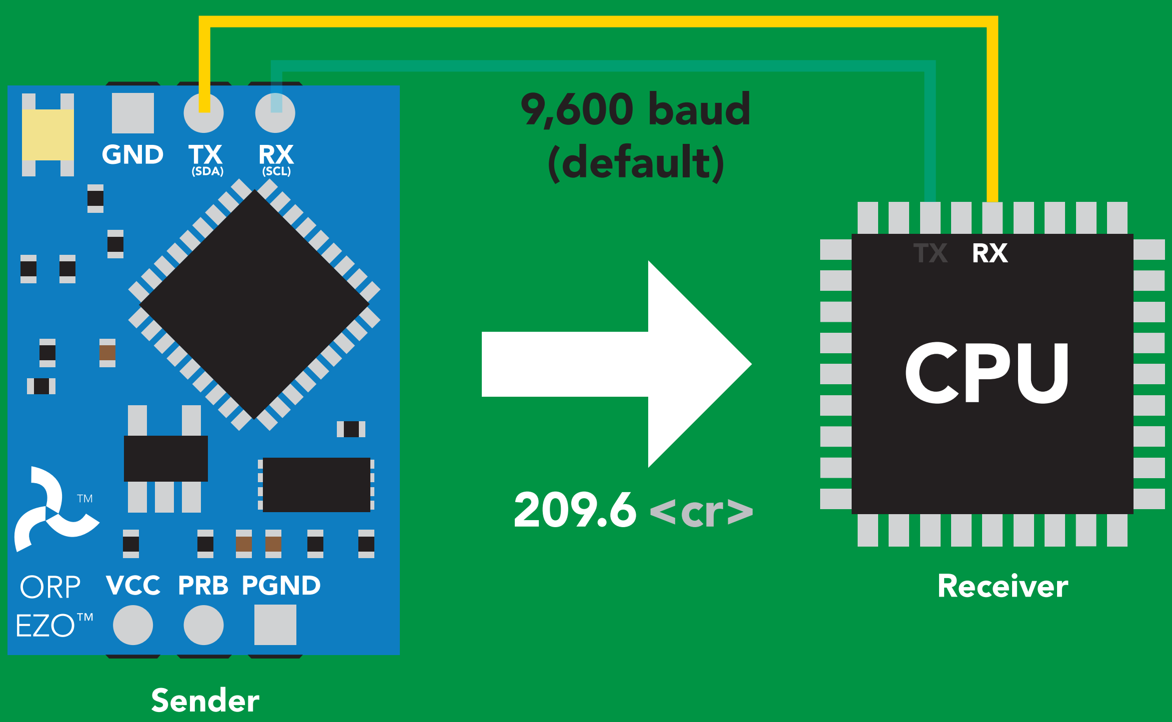


Receiving data from device

2 parts

ASCII data string — Command

Carriage return <cr> — Terminator



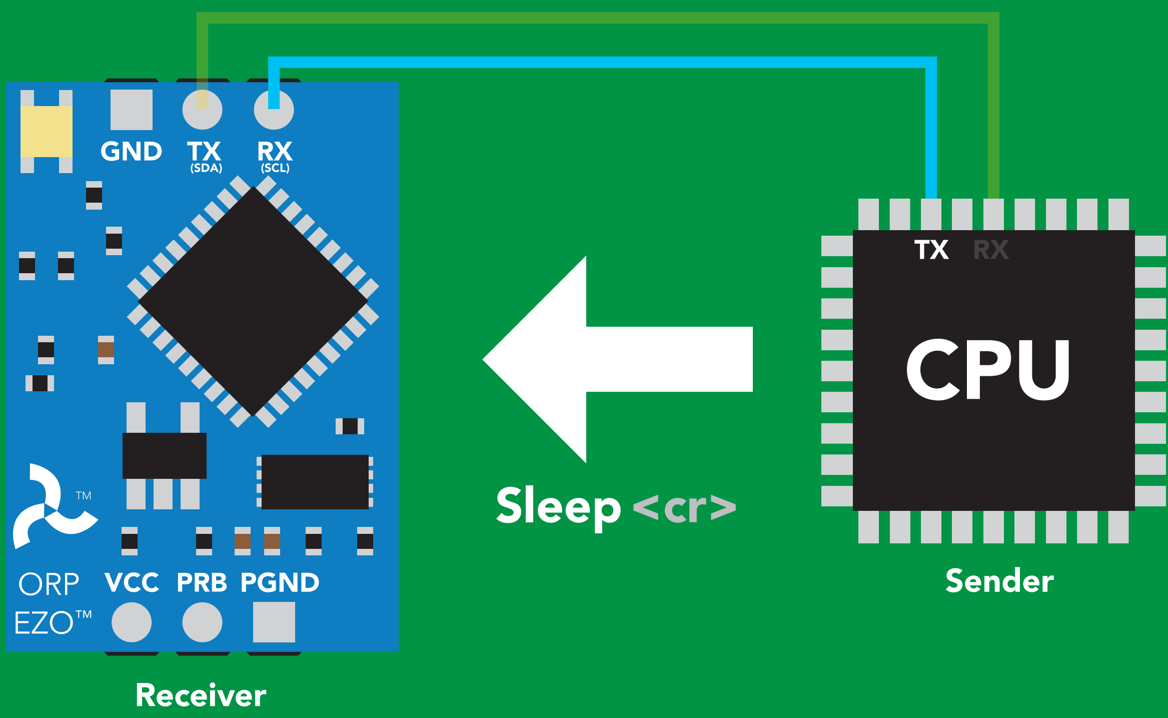
Advanced

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

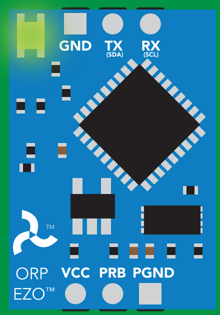
Command (not case sensitive)	Carriage return <cr>
ASCII data string	Terminator



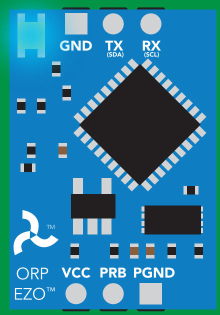
Advanced

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

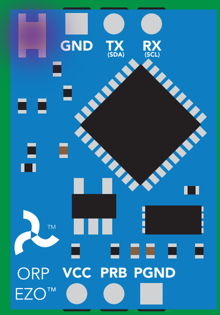
LED color definition



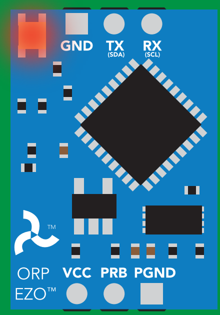
Green
UART standby



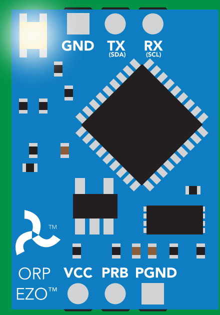
Cyan
Taking reading



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

LED control

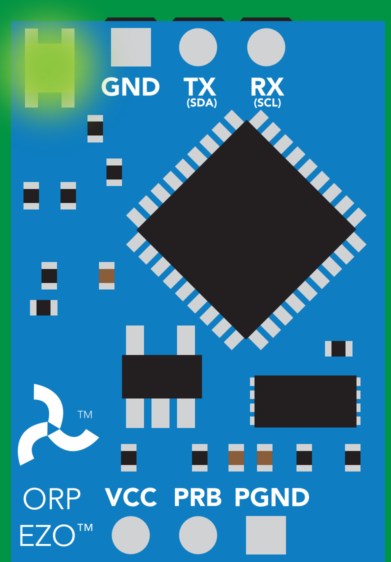
Command syntax

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

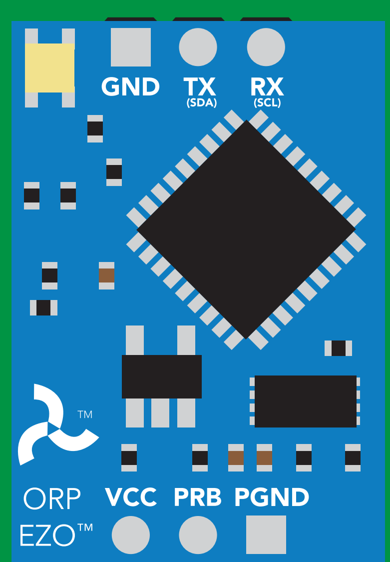
L,0 <cr> LED off

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

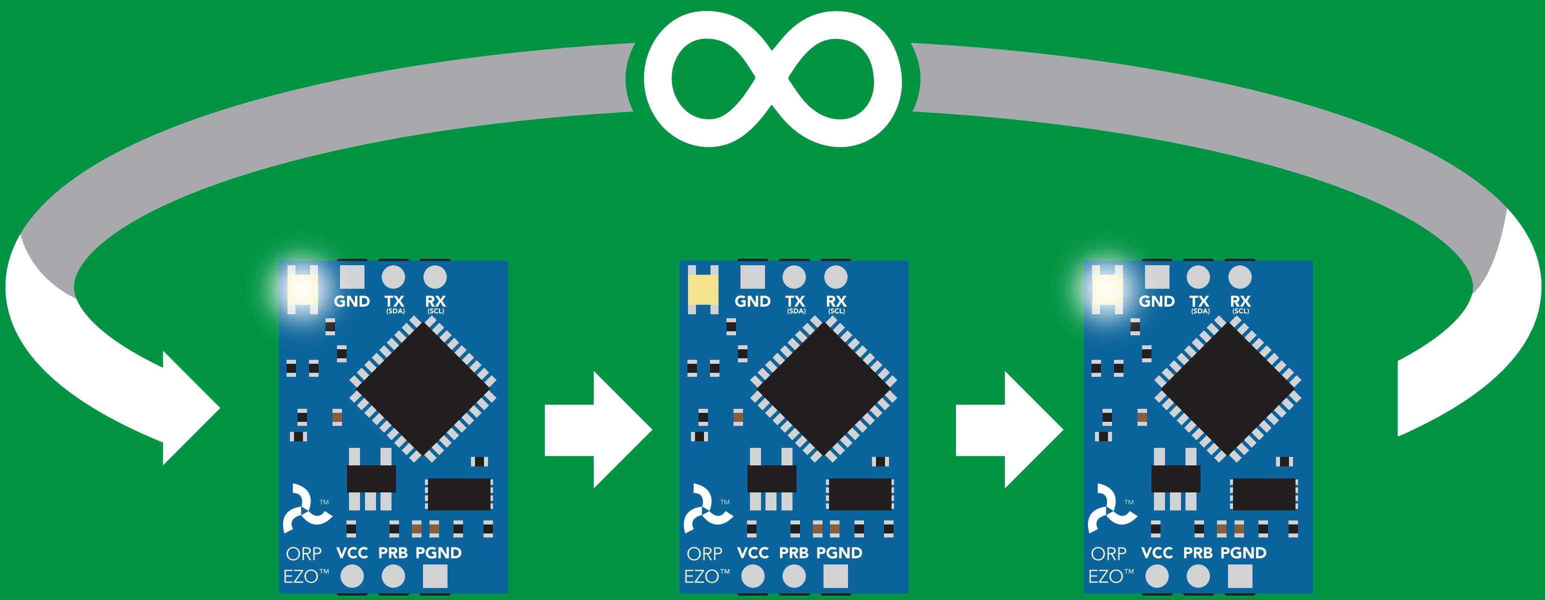
Command syntax

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*

*This command is only available for firmware version 2.10 and above.

Example	Response
Find <cr>	*OK <cr>



AtlasScientific™
Environmental Robotics
r 0.1

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?	

*This command is only available for firmware version 2.10 and above.

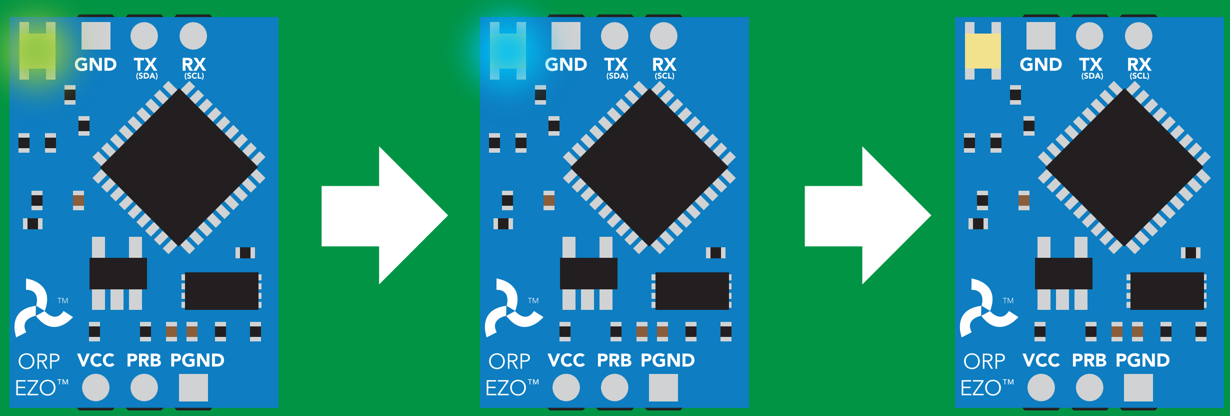
Example	Response
C,1 <cr>	*OK <cr> ORP (1 sec) <cr> ORP (2 sec) <cr> ORP (n sec) <cr>
C,30 <cr>	*OK <cr> ORP (30 sec) <cr> ORP (60 sec) <cr> ORP (90 sec) <cr>
C,0 <cr>	*OK <cr>
C,? <cr>	?C,1 <cr> or ?C,0 <cr> or ?C,30 <cr> *OK <cr>

Single reading mode

Command syntax

R <cr> **takes single reading**

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



Green
Standby

Cyan
Taking reading

Transmitting

800 ms

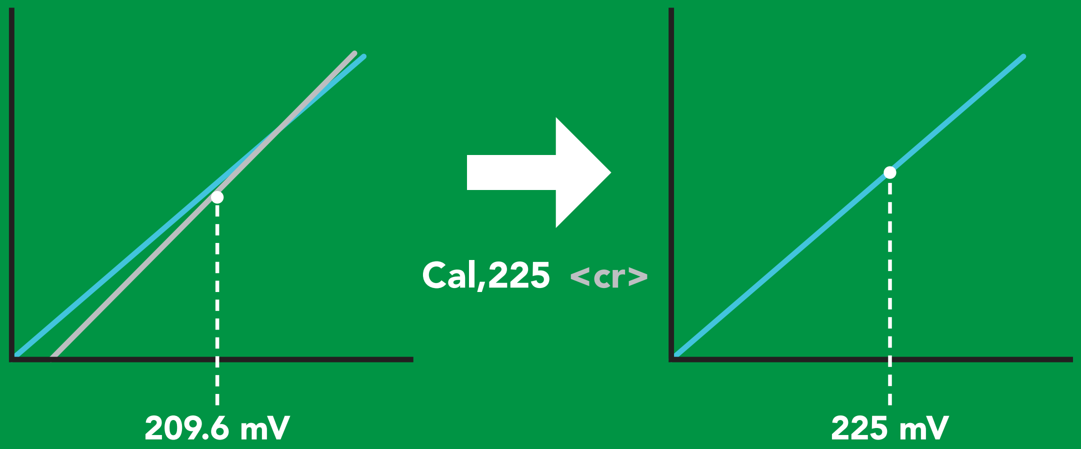
Calibration

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/import calibration

Command syntax

Export: Use this command to save calibration settings
Import: Use this command to load calibration settings to one or more devices.

Export	<cr>	export calibration string from calibrated device*
Import	<cr>	import calibration string to new device*
Export,?	<cr>	calibration string info*

*This command is only available for firmware version 2.10 and above.

Example	Response
Export,? <cr>	10,120 <cr>
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
Import, n (FIFO)	Import, 59 6F 75 20 61 72 <cr> (1 of 10)

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>

Disabling *OK simplifies this process

Naming device

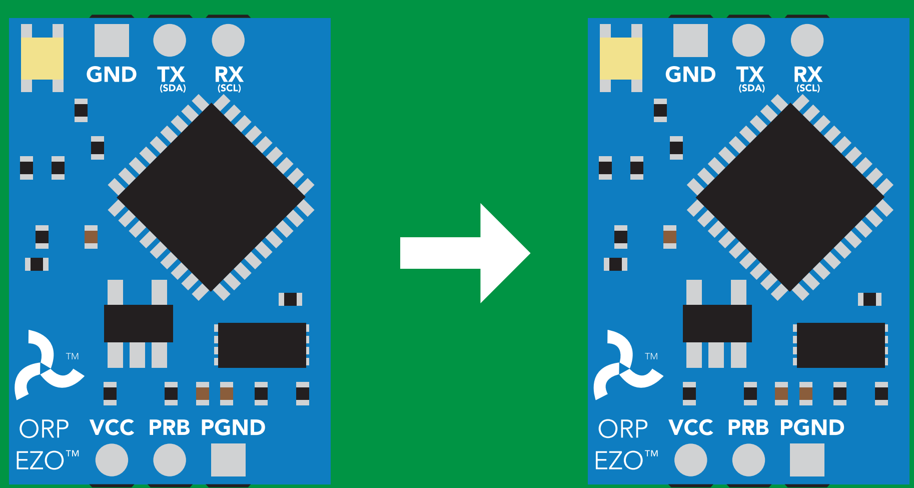
Command syntax

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

Example	Response
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,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

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>	*SL
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

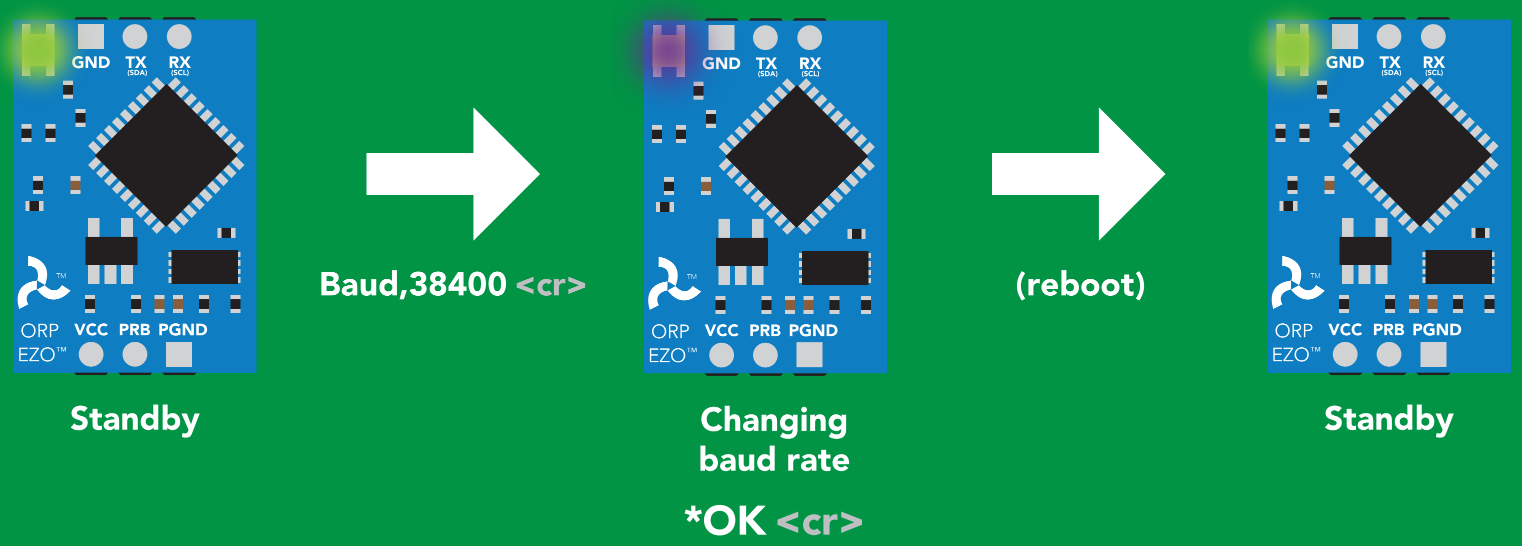
Change baud rate

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.

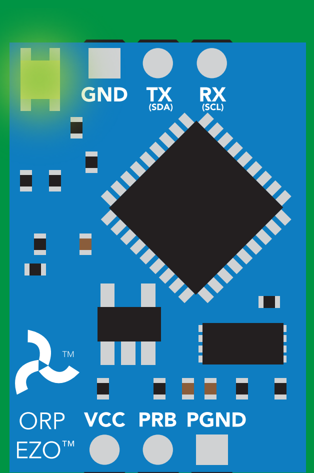
Plock,1 <cr> enable Plock

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

Plock,? <cr> Plock on/off?

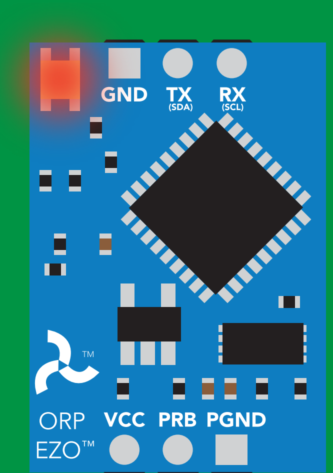
Example	Response
Plock,1 <cr>	*OK <cr>
Plock,0 <cr>	*OK <cr>
Plock,? <cr>	?Plock,1 <cr> or ?Plock,0 <cr>

Plock,1



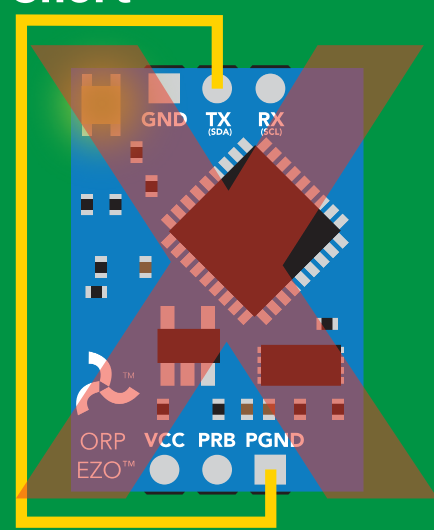
*OK <cr>

I2C,100



cannot change to I²C
*ER <cr>

Short



cannot change to I²C

Factory reset

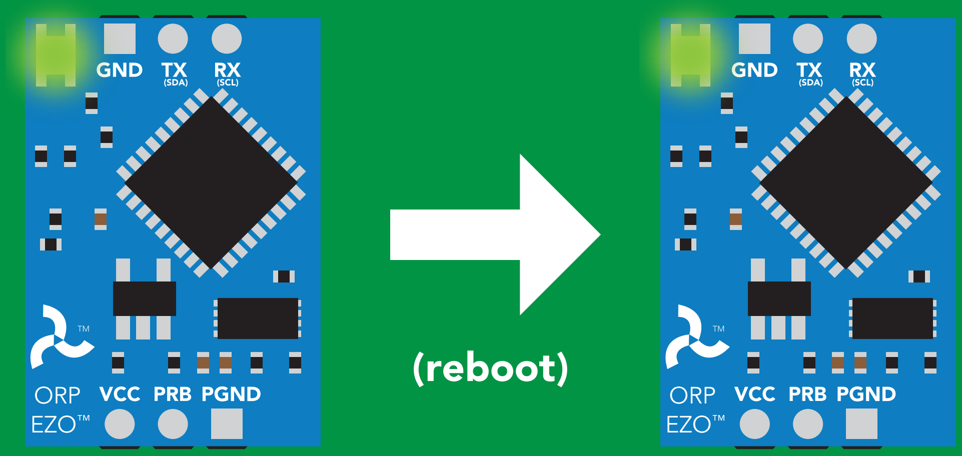
Command syntax

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

AtlasScientific™ Environmental Robotics

Change to I²C mode

Command syntax

Default I²C address 98 (0x62)

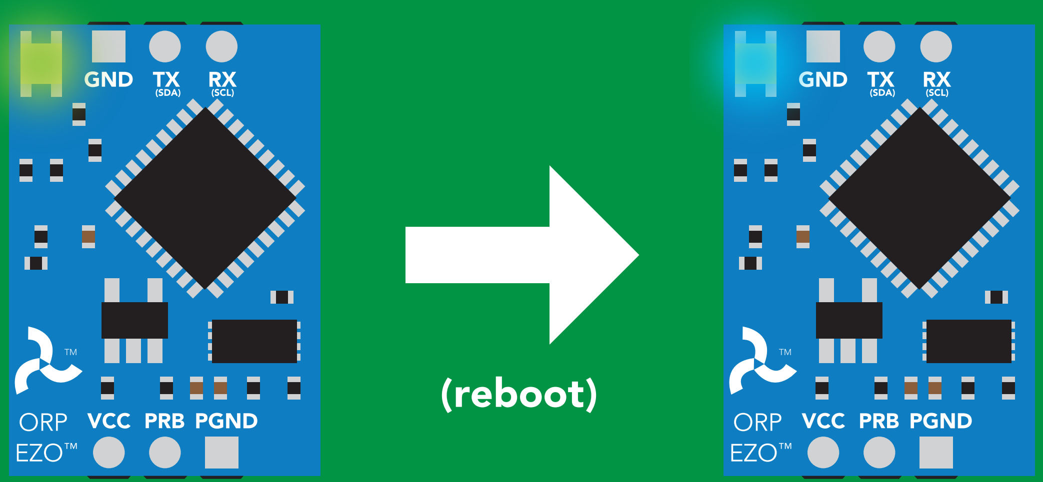
I2C,n <cr> sets I²C address and reboots into I²C mode

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

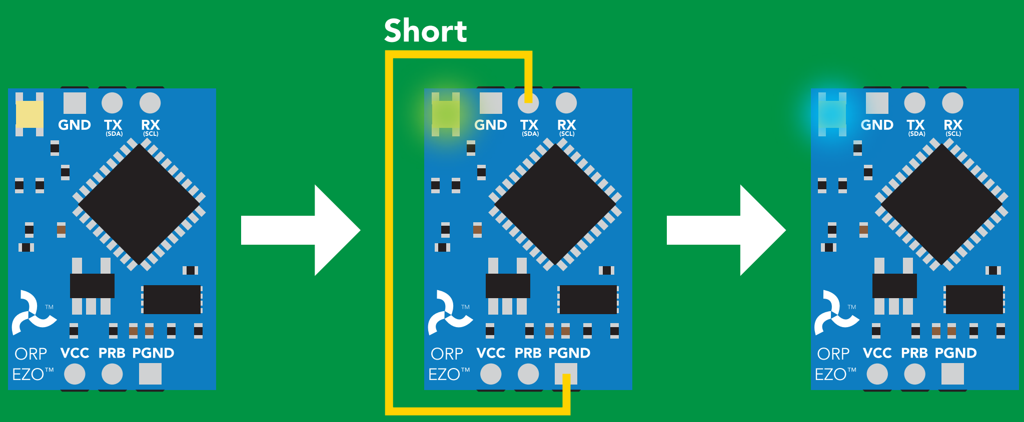
AtlasScientific Environmental Robotics r 0.2

Manual switching to I²C

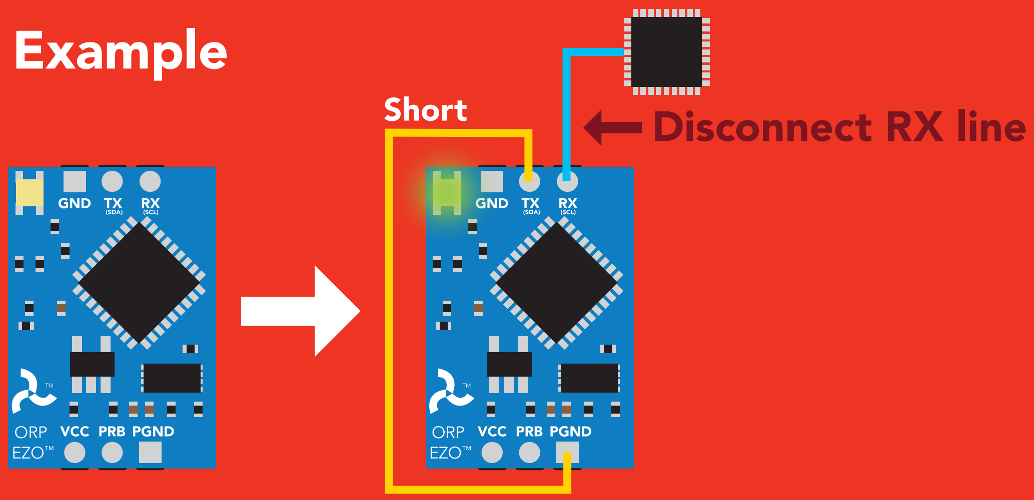
- Make sure Plock is set to 0
- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to PGND
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from Green to Blue
- Disconnect ground (power off)
- Reconnect all data and power

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

Example



Wrong Example



AtlasScientific
Environmental Robotics

I²C mode

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

To set your EZO™ 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

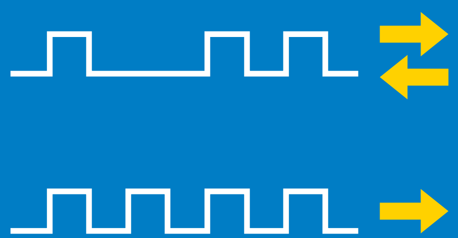
AtlasScientific™
Environmental Robotics

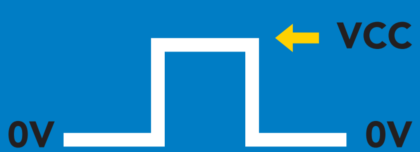
I²C mode

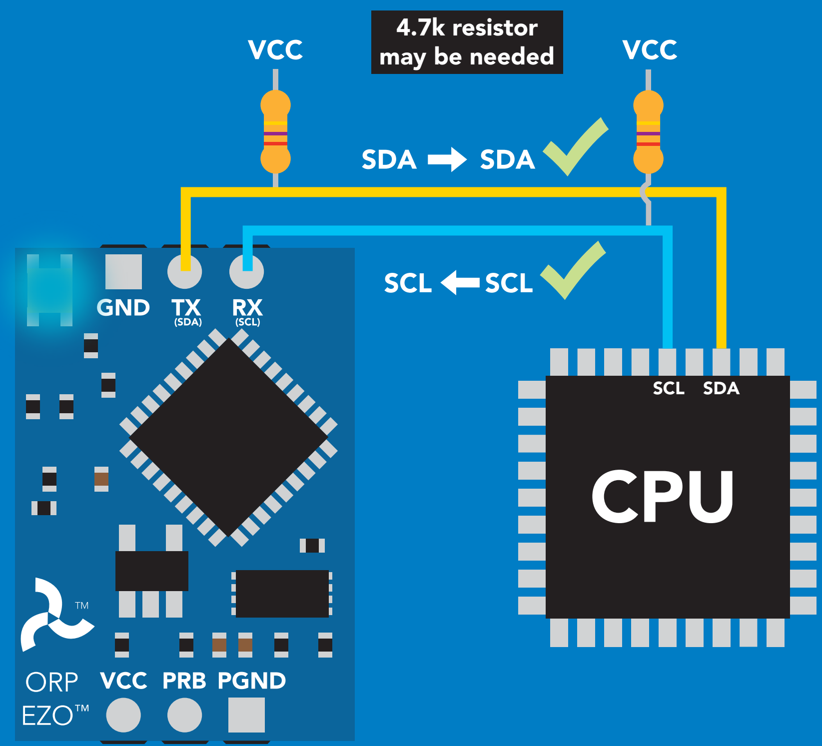
I²C address	(0x01 – 0x7F) 98 (0x62) default
Vcc	3.3V – 5.5V
Clock speed	100 – 400 kHz

SDA

SCL







Data format

Reading	ORP	Data type	floating point
Units	mV	Decimal places	1
Encoding	ASCII	Smallest string	2 characters
Format	string	Largest string	399 characters

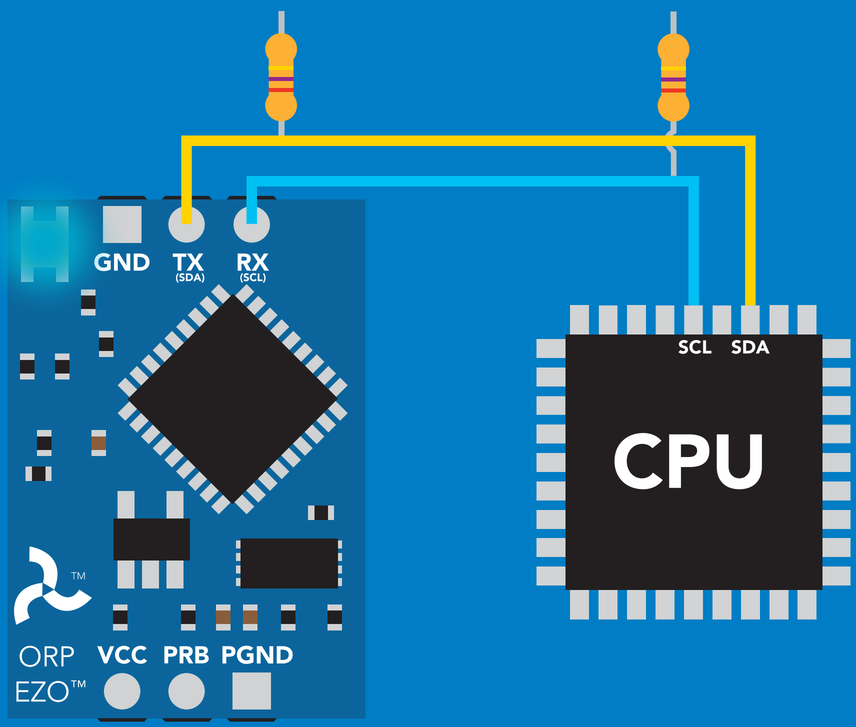
Sending commands to device

5 parts

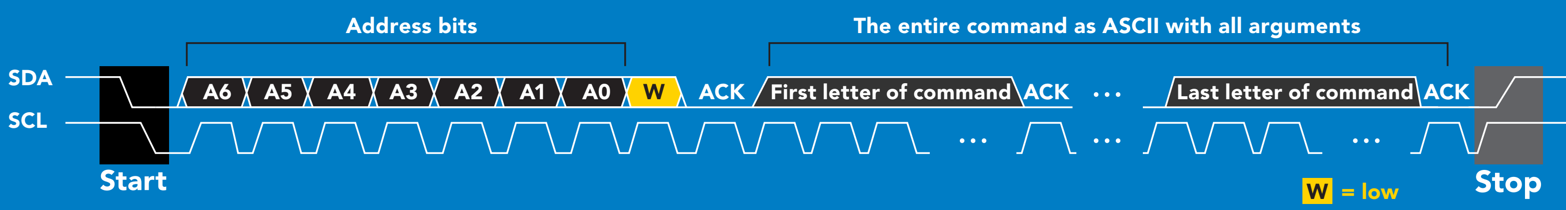
Start	I²C address	Write	Command (not case sensitive)	Stop
	98 (0x62)		ASCII command string	

Example

Start	98 (0x62)	Write	Sleep	Stop
	I²C address		Command	



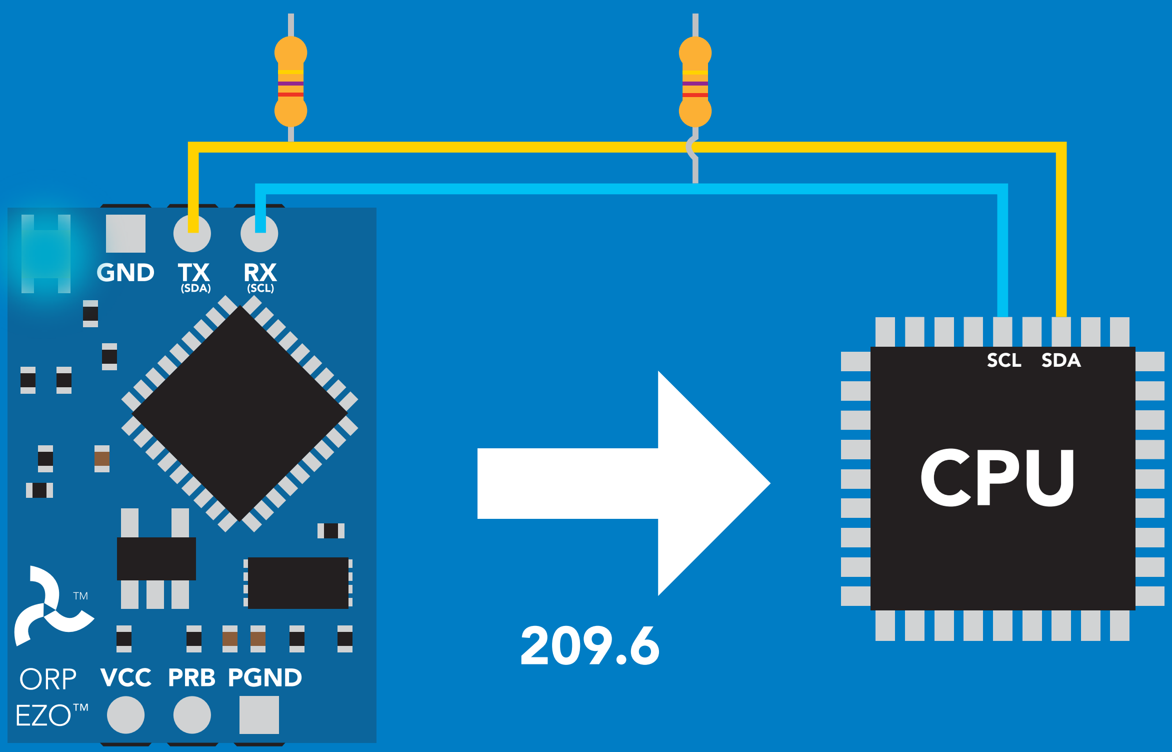
Advanced



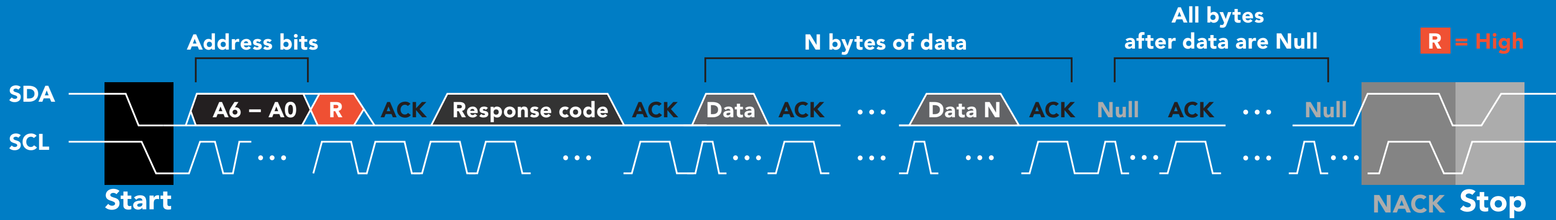
Requesting data from device

7 parts

Start	I²C address	Read	Response code	Data string	Null	Stop
	98 (0x62)		1 byte	"209.6"	Terminator (Dec 0)	



Advanced



1	50	48	57	46	54	0	= 209.6
Dec	ASCII					Dec	

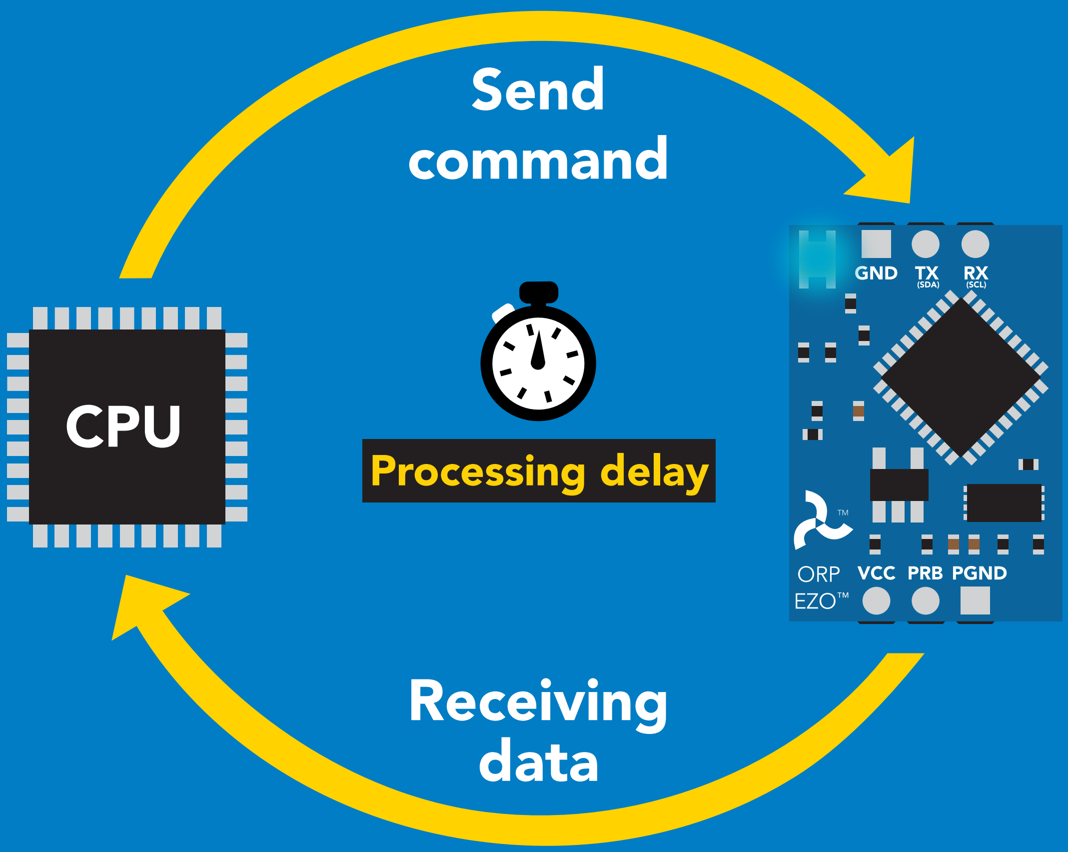
AtlasScientific™
Environmental Robotics

r 0.1

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;
```

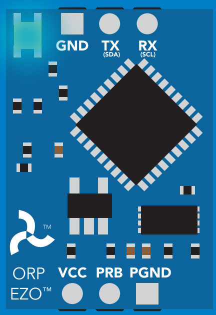
If there is no processing delay or the processing delay is too short, the response code will always be 254.

Response codes

Single byte, not string

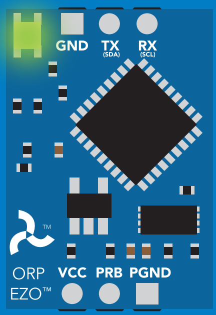
255	no data to send
254	still processing, not ready
2	syntax error
1	successful request

LED color definition



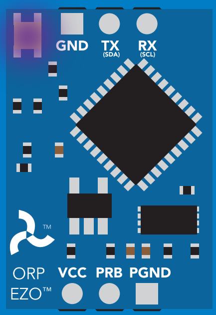
Blue

I²C standby



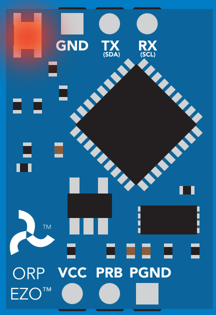
Green

Taking reading



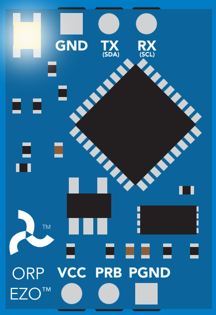
Purple

Changing I²C ID#



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. 56
Cal	performs calibration	pg. 48
Export/import	export/import calibration	pg. 49
Factory	enable factory reset	pg. 55
Find	finds device with blinking white LED	pg. 46
i	device information	pg. 50
I2C	change I^2C address	pg. 54
L	enable/disable LED	pg. 45
Plock	enable/disable protocol lock	pg. 53
R	returns a single reading	pg. 47
Sleep	enter sleep mode/low power	pg. 52
Status	retrieve status information	pg. 51

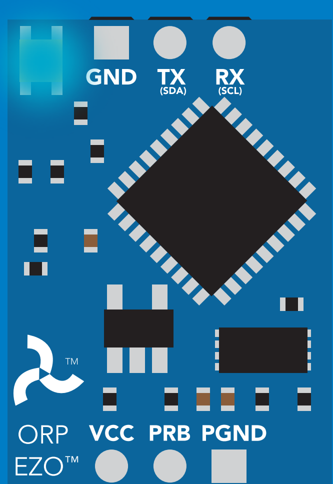
LED control

Command syntax

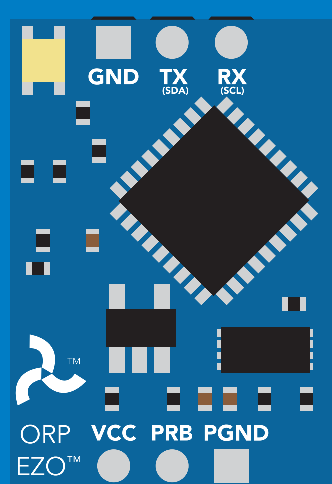
300ms processing delay

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

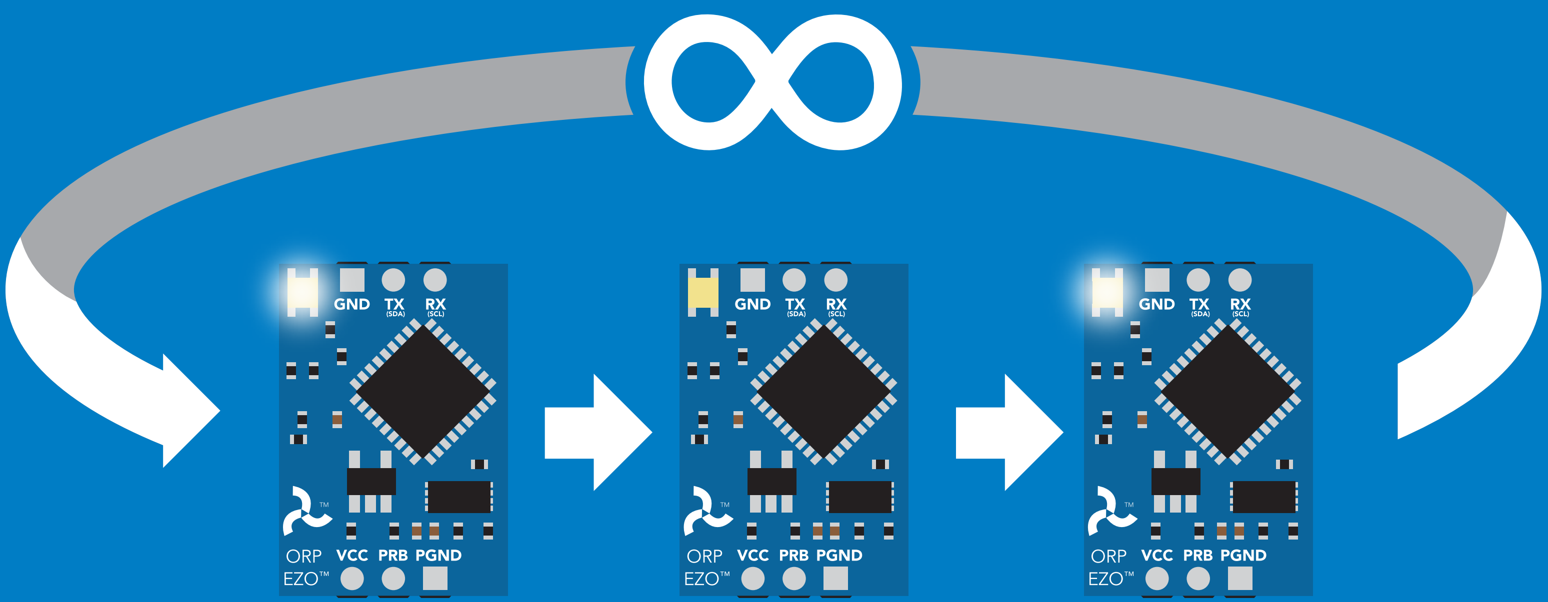
Command syntax

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

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

*This command is only available for firmware version 2.10 and above.

Example	Response
Find <cr>	Wait 300ms · 1 (Dec) · 0 (Null)



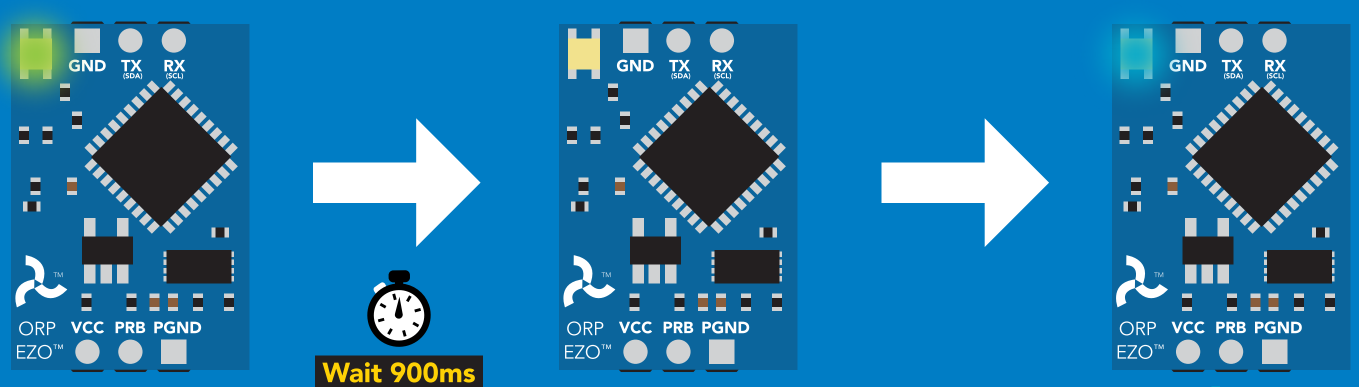
Taking reading

Command syntax

900ms processing delay

R	**return 1 reading**

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



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

Calibration

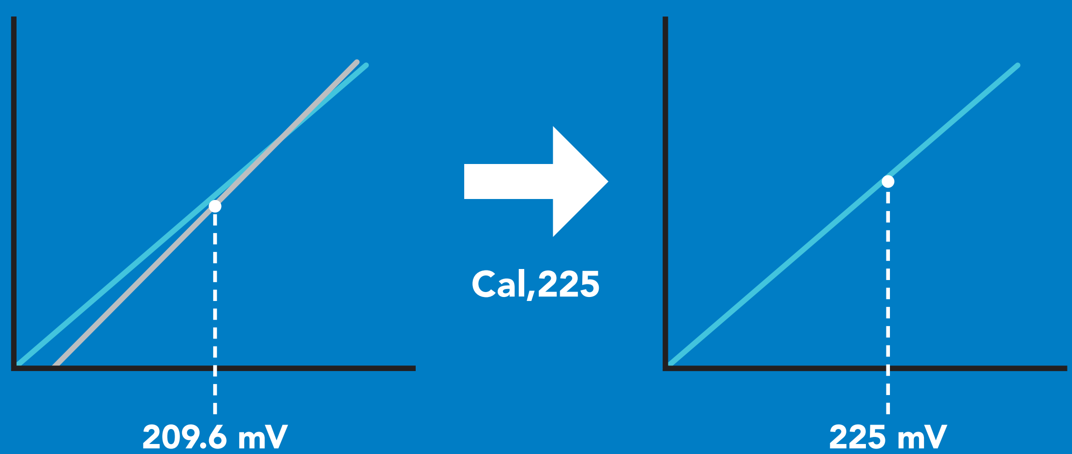
Command syntax

300ms 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)



Export/import calibration

Command syntax

Export: Use this command to save calibration settings
Import: Use this command to load calibration settings to one or more devices.

Export	export calibration string from calibrated device*
Import	import calibration string to new device*
Export,?	calibration string info*

300ms processing delay

*This command is only available for firmware version 2.10 and above.

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)
(8 more)	⋮
Export	Wait 300ms — 1 (Dec) 65 20 61 20 63 6F (ASCII) 0 (Null) (10 of 10)
Export	Wait 300ms — 1 (Dec) *DONE (ASCII) 0 (Null)
Import, n (FIFO)	Import, 59 6F 75 20 61 72 (ASCII) (1 of 10)

Response breakdown

10, 120

10 = # of strings to export

120 = # of bytes to export

Export strings can be up to 12 characters long

Device information

Command syntax

300ms processing delay

i	device information

Example	Response
i	Wait 300ms · 1 (Dec) · ?i,ORP, 19.7 (ASCII) · 0 (Null)

Response breakdown

?i, ORP, 1.97

ORP, = Device

1.97 = 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

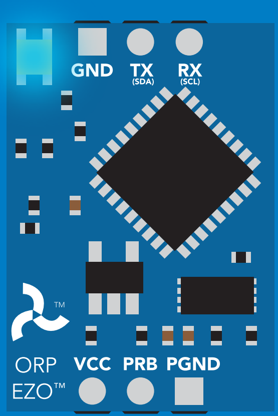
300ms processing delay

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

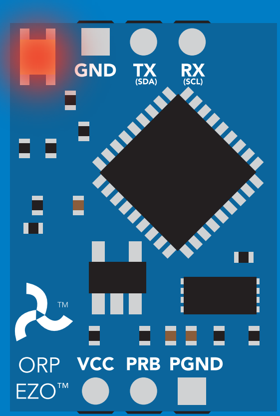
Locks device to I²C mode.

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)

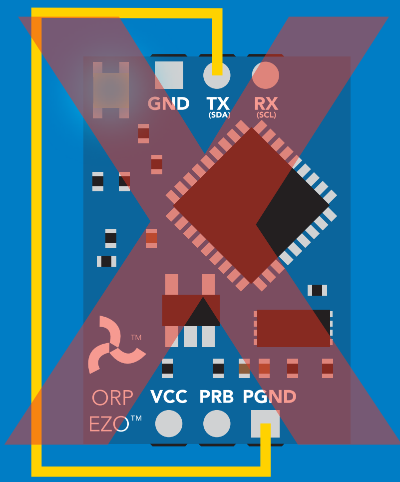
Plock,1



Serial, 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

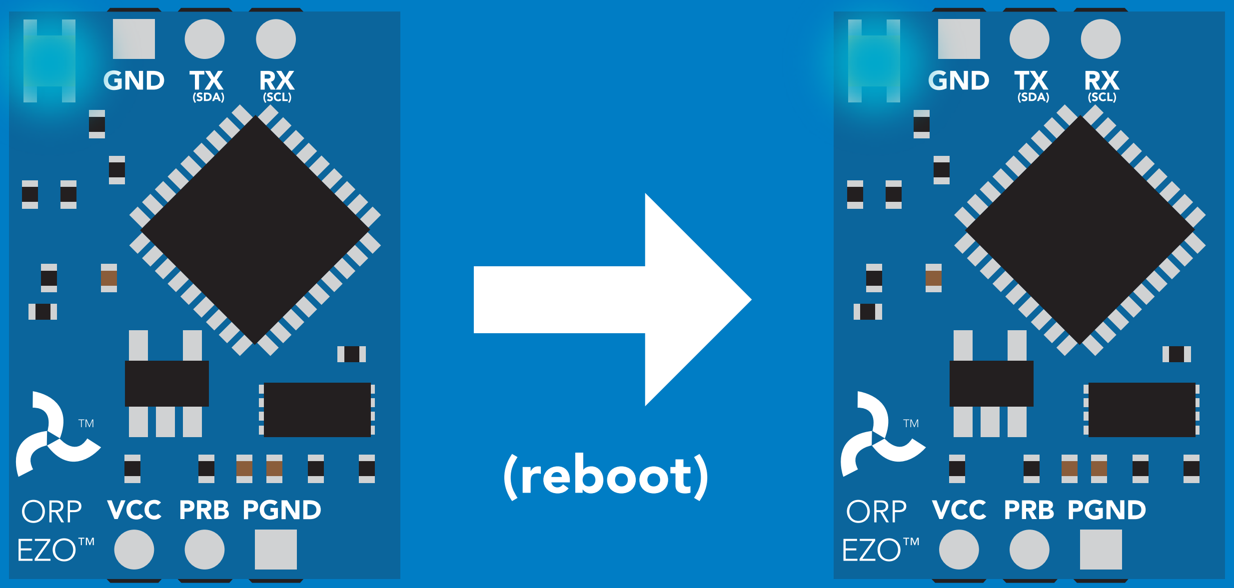
Warning!

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).

n = any number 1 – 127

I2C,100



Factory reset

Command syntax

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

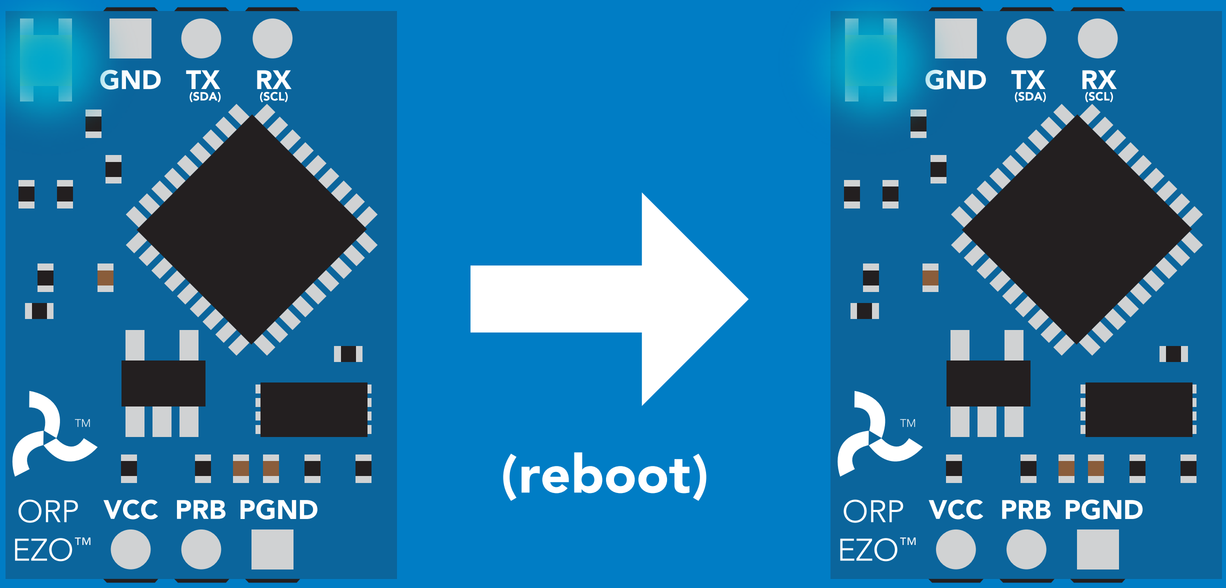
Factory	enable factory reset

I²C address will not change

Example	Response
Factory	device reboot

Clears calibration
LED on
Response codes enabled

Factory



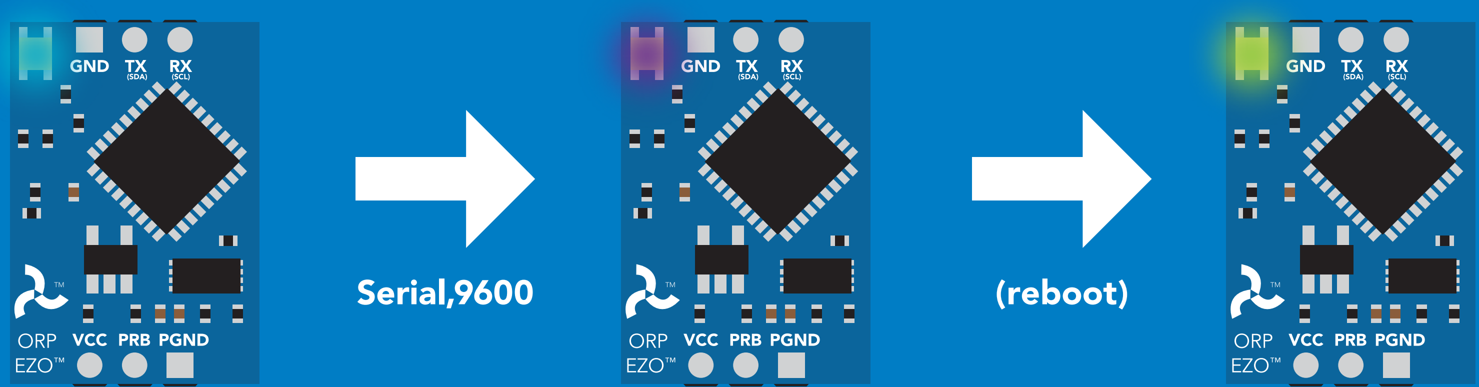
Change to UART mode

Command syntax

Baud,n	switch from I²C to UART

Example	Response
Baud,9600	reboot in UART mode

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



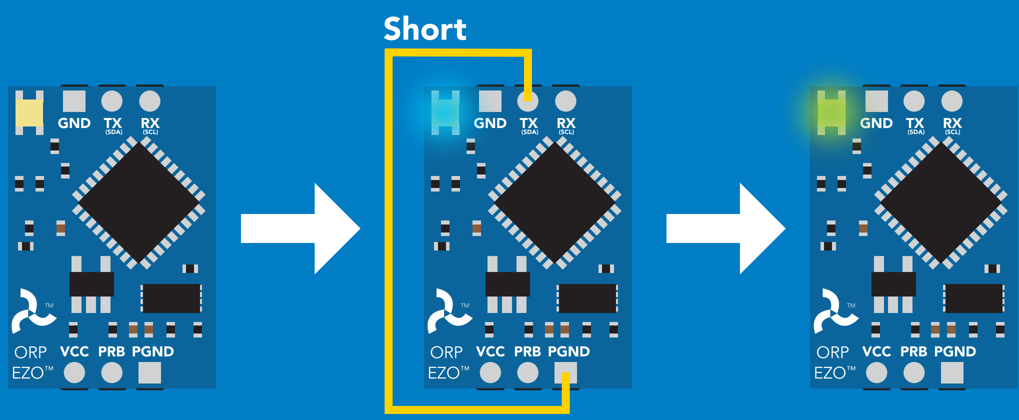
Changing to UART mode

AtlasScientific™
Environmental Robotics
r 0.1

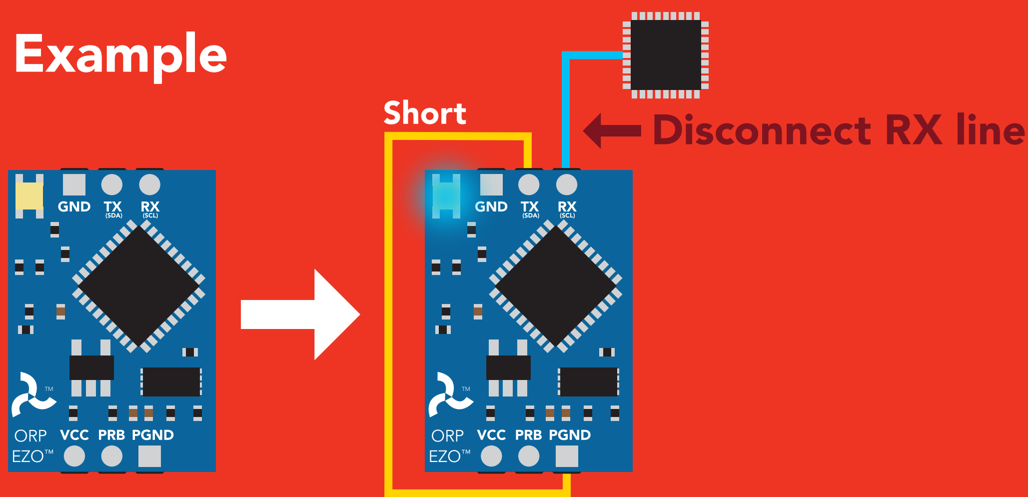
Manual switching to UART

- Make sure Plock is set to 0
- 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

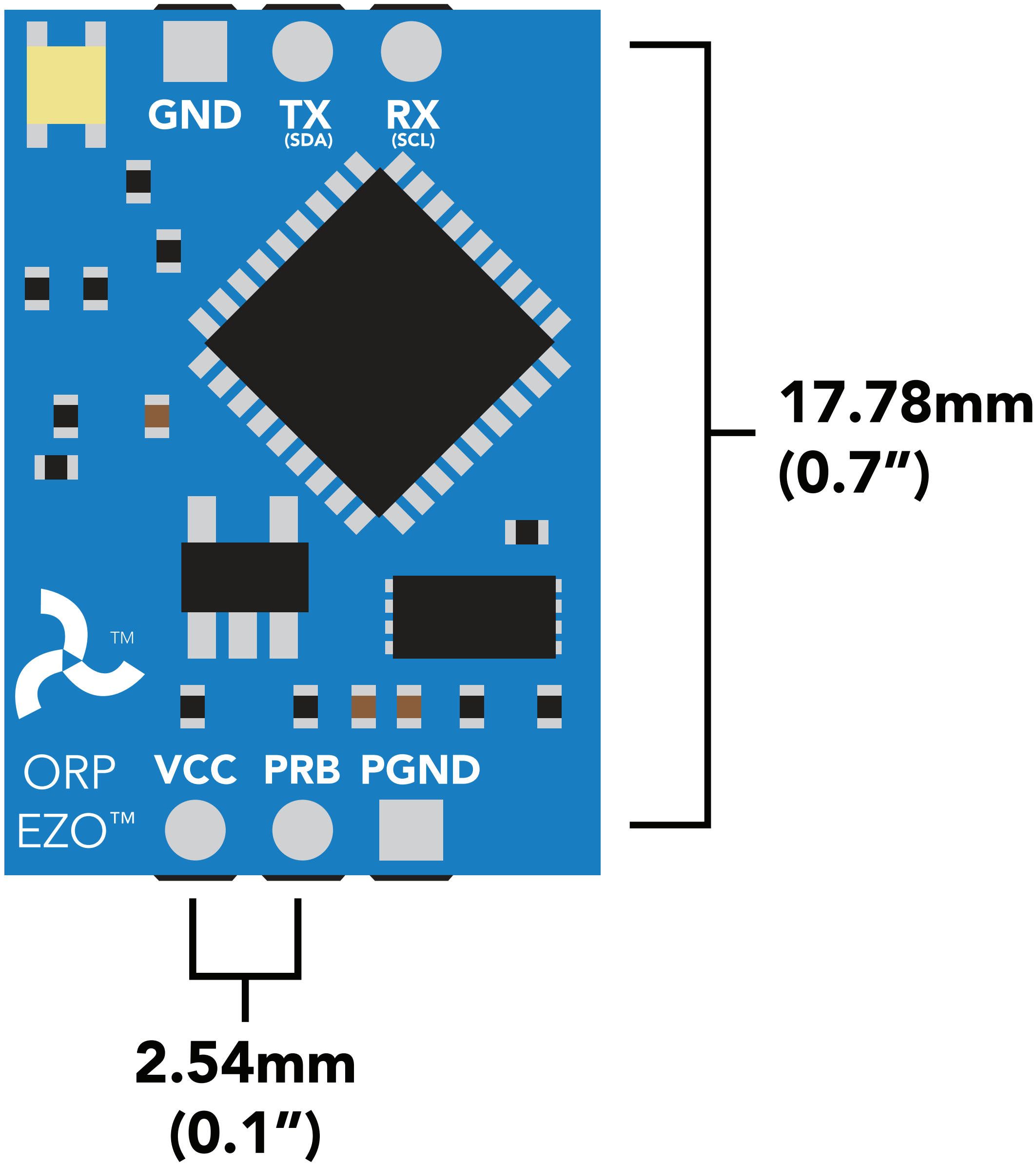
Example



Wrong Example



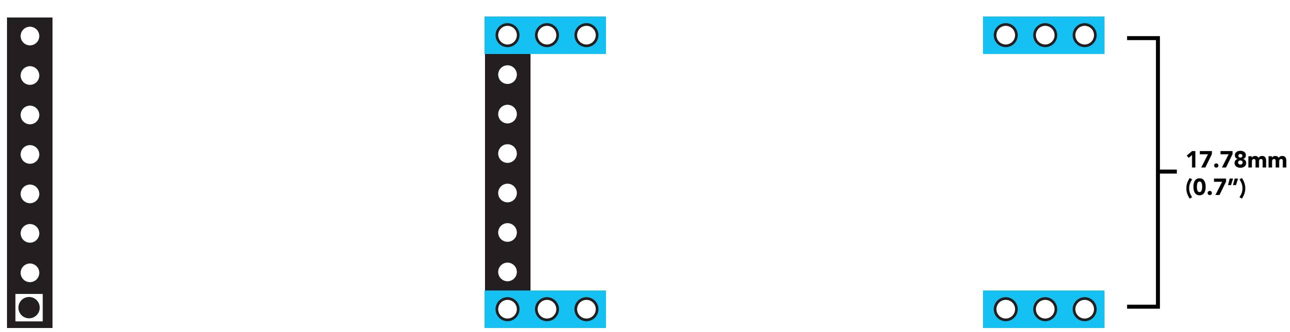
EZO™ circuit footprint



1 In your CAD software place an 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 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 & 46.
"Export/Import calibration" pages 27 & 49.
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

V1.1 – 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 – I²C bug (Dec 1, 2014)
- Fix 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 glitch where EEPROM would get erased if the circuit lost power 900ms into startup

V1.97 – EEPROM (Oct 10, 2016)
- Fixed glitch 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.

AtlasScientific
Environmental Robotics

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.

