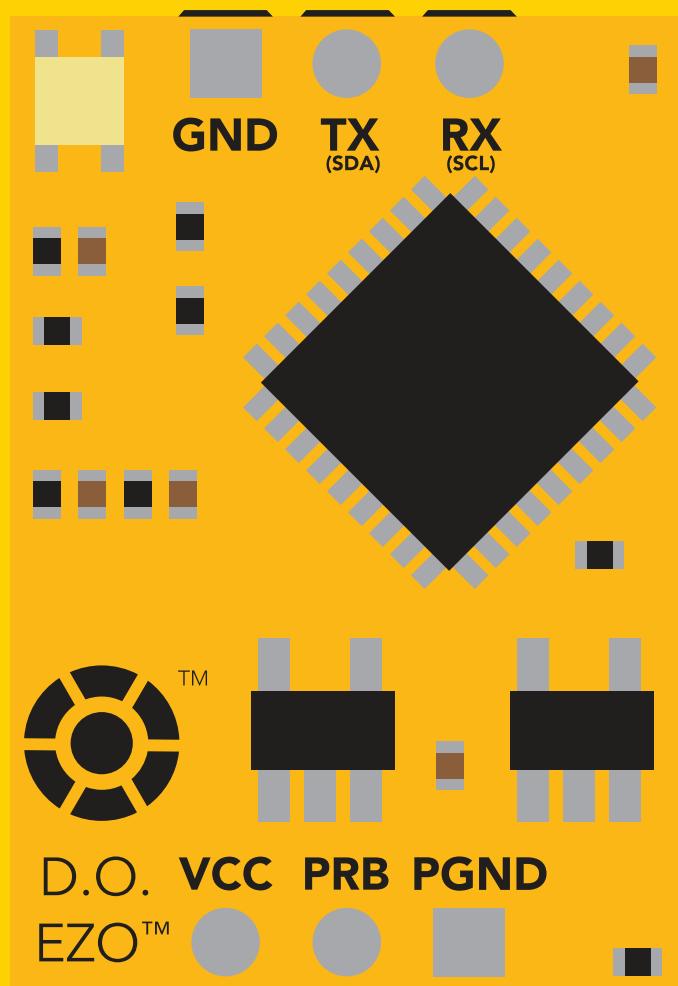


# EZO-DO™

Embedded Dissolved Oxygen Circuit

Reads	<b>Dissolved Oxygen</b>
Range	<b>0.01 – 100+ mg/L 0.1 – 400+ % saturation</b>
Accuracy	<b>+/- 0.05 mg/L</b>
Response time	<b>1 reading per sec</b>
Supported probes	<b>Any galvanic probe</b>
Calibration	<b>1 or 2 point</b>
Temperature, salinity and pressure compensation	<b>Yes</b>
Data protocol	<b>UART &amp; I<sup>2</sup>C</b>
Default I <sup>2</sup> C address	<b>97 (0x61)</b>
Operating voltage	<b>3.3V – 5V</b>
Data format	<b>ASCII</b>



PATENT PROTECTED



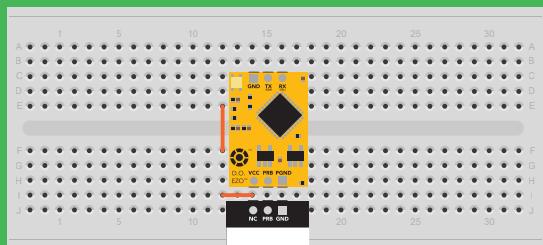
# STOP

**SOLDERING THIS DEVICE VOIDS YOUR WARRANTY.**

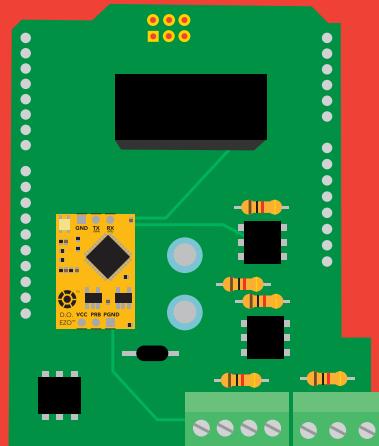
**This is sensitive electronic equipment. Get this device working in a solderless breadboard first. Once this device has been soldered it is no longer covered by our warranty.**

**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

Circuit dimensions	4	Correct wiring	8
Power consumption	4	Calibration theory	9
Absolute max ratings	4	Preserve calibration solution	12
Operating principle	5	Default state	13
Power and data isolation	6	Available data protocols	14

## UART

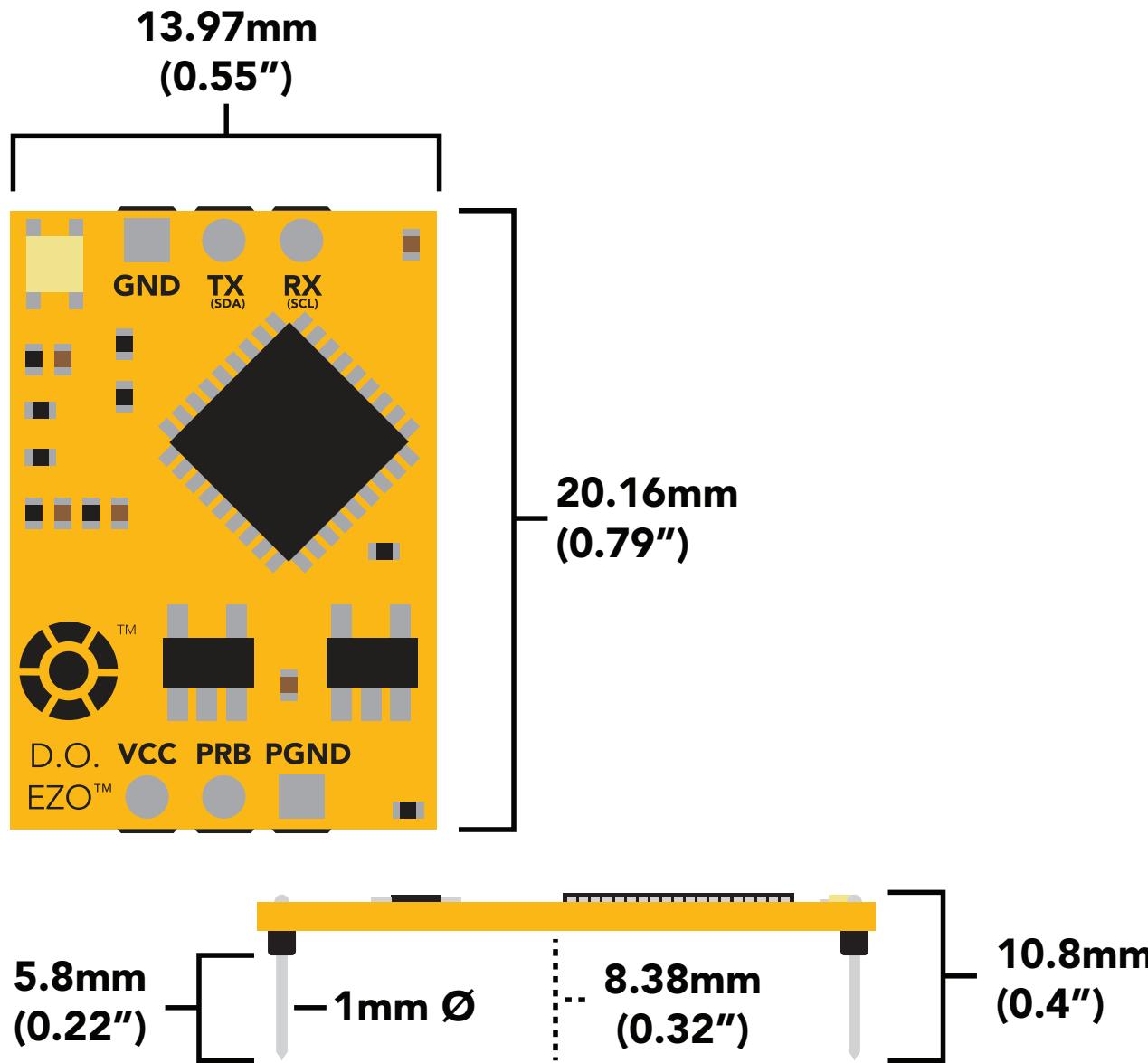
UART mode	16
Receiving data from device	17
Sending commands to device	18
LED color definition	19
UART quick command page	20
LED control	21
Find	22
Continuous reading mode	23
Single reading mode	24
Calibration	25
Export calibration	26
Import calibration	27
Temperature compensation	28
Salinity compensation	29
Pressure compensation	30
Enable/disable parameters	31
Naming device	32
Device information	33
Response codes	34
Reading device status	35
Sleep mode/low power	36
Change baud rate	37
Protocol lock	38
Factory reset	39
Change to I <sup>2</sup> C mode	40
Manual switching to I <sup>2</sup> C	41

## I<sup>2</sup>C

I <sup>2</sup> C mode	43
Sending commands	44
Requesting data	45
Response codes	46
LED color definition	47
I <sup>2</sup> C quick command page	48
LED control	49
Find	50
Taking reading	51
Calibration	52
Export calibration	53
Import calibration	54
Temperature compensation	55
Salinity compensation	56
Pressure compensation	57
Enable/disable parameters	58
Device information	59
Reading device status	60
Sleep mode/low power	61
Protocol lock	62
I <sup>2</sup> C address change	63
Factory reset	64
Change to UART mode	65
Manual switching to UART	66

Circuit footprint	67
Datasheet change log	68
Warranty	72

# EZO™ circuit dimensions



## Power consumption

## Absolute max ratings

	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	

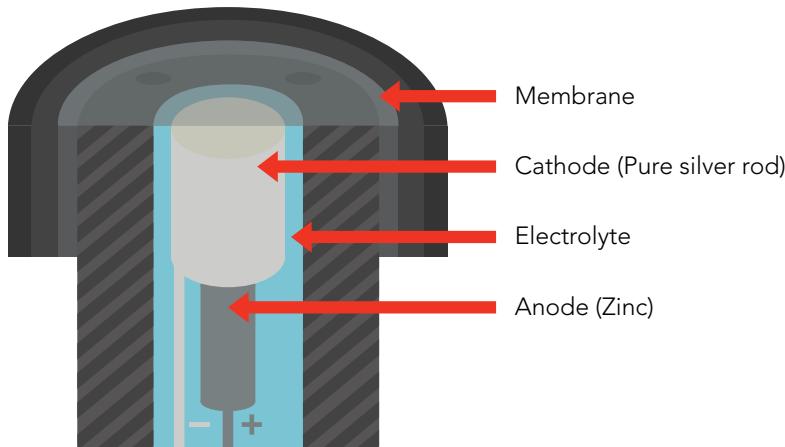
Parameter	MIN	TYP	MAX
Storage temperature (EZO™ D.O.)	-65 °C		125 °C
Operational temperature (EZO™ D.O.)	-40 °C	25 °C	85 °C
VCC	3.3V	5V	5.5V

# Operating principle

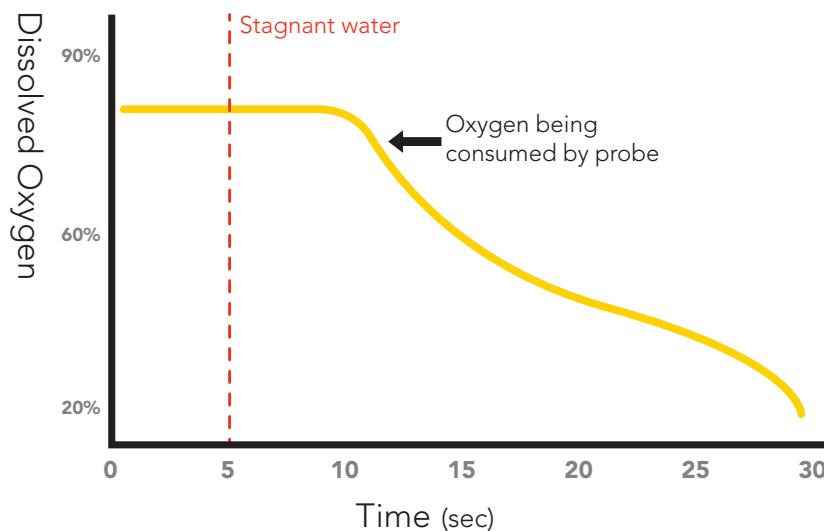
The Atlas Scientific™ EZO™ Dissolved Oxygen circuit works with:

- |                              |  |
|------------------------------|--|
| <b>X Optical probe</b>       | Slow response, requires external power, expensive. |
| <b>X Polar Graphic probe</b> | Requires external power, output in $\mu\text{A}$ . |
| <b>✓ Galvanic probe</b>      | Requires no external power, output in mV.          |

A galvanic dissolved oxygen probe consists of a PTFE membrane, an anode bathed in an electrolyte and a cathode. Oxygen molecules diffuse through the probe's membrane at a constant rate (without the membrane the reaction happens too quickly). Once the oxygen molecules have crossed the membrane they are reduced at the cathode and a small voltage is produced. If no oxygen molecules are present, the probe will output 0 mV. As the oxygen increases so does the mV output from the probe. Each probe will output a different voltage in the presence of oxygen. The only thing that is constant is that **0mV = 0 Oxygen**. (A galvanic dissolved oxygen probe can also be used to detect the Oxygen content in gases).



## Flow Dependence

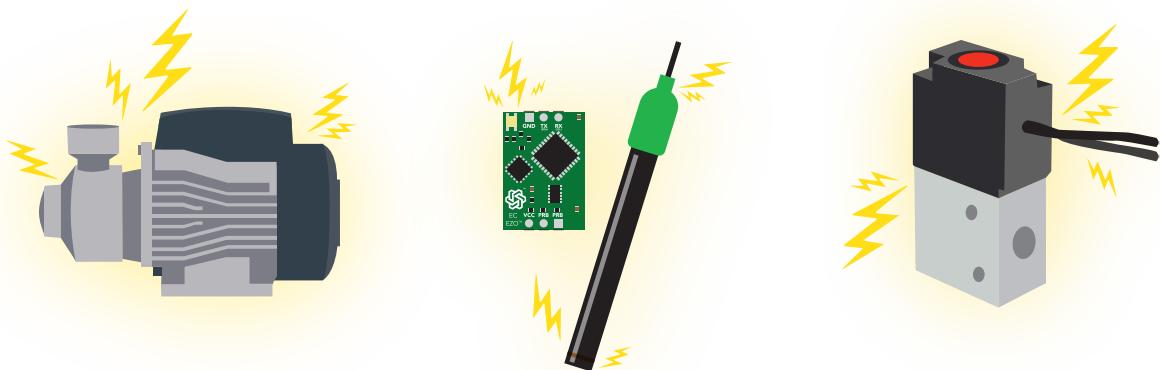


One of the drawbacks from using a galvanic probe is that it consumes a **VERY** small amount of the oxygen it reads. Therefore, a small amount of water movement is necessary to take accurate readings. **Approximately 60 ml/min.**

# Power and data isolation

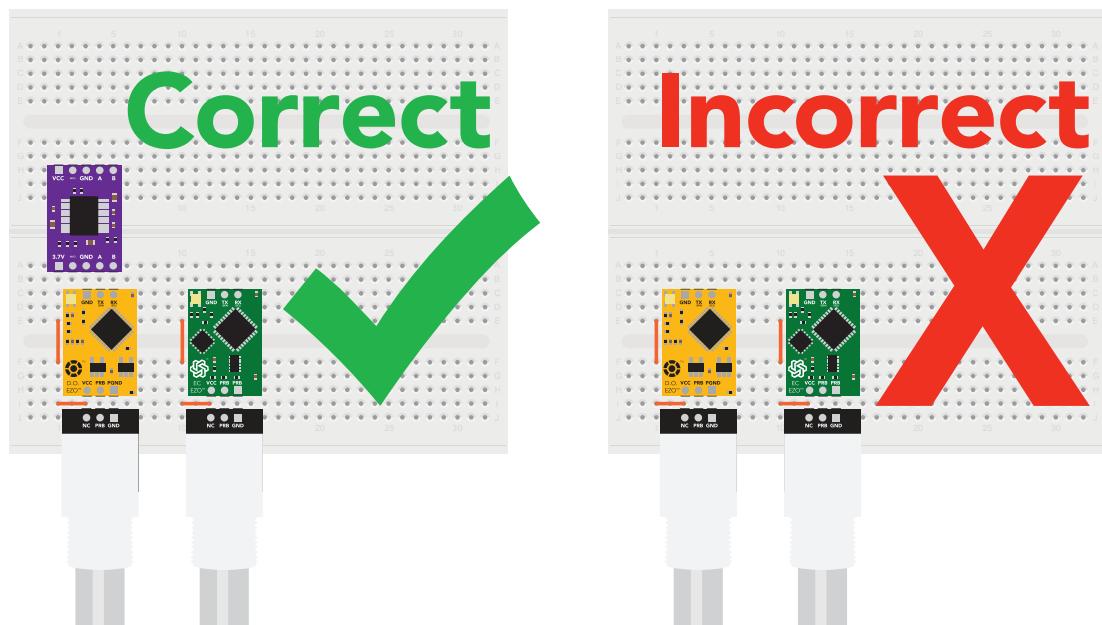
The Atlas Scientific EZO™ Dissolved Oxygen circuit is a very sensitive device. This sensitivity is what gives the Dissolved Oxygen circuit its accuracy. This also means that the Dissolved Oxygen circuit is capable of reading micro-voltages that are bleeding into the water from unnatural sources such 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.



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

Basic EZO™  
Inline Voltage Isolator



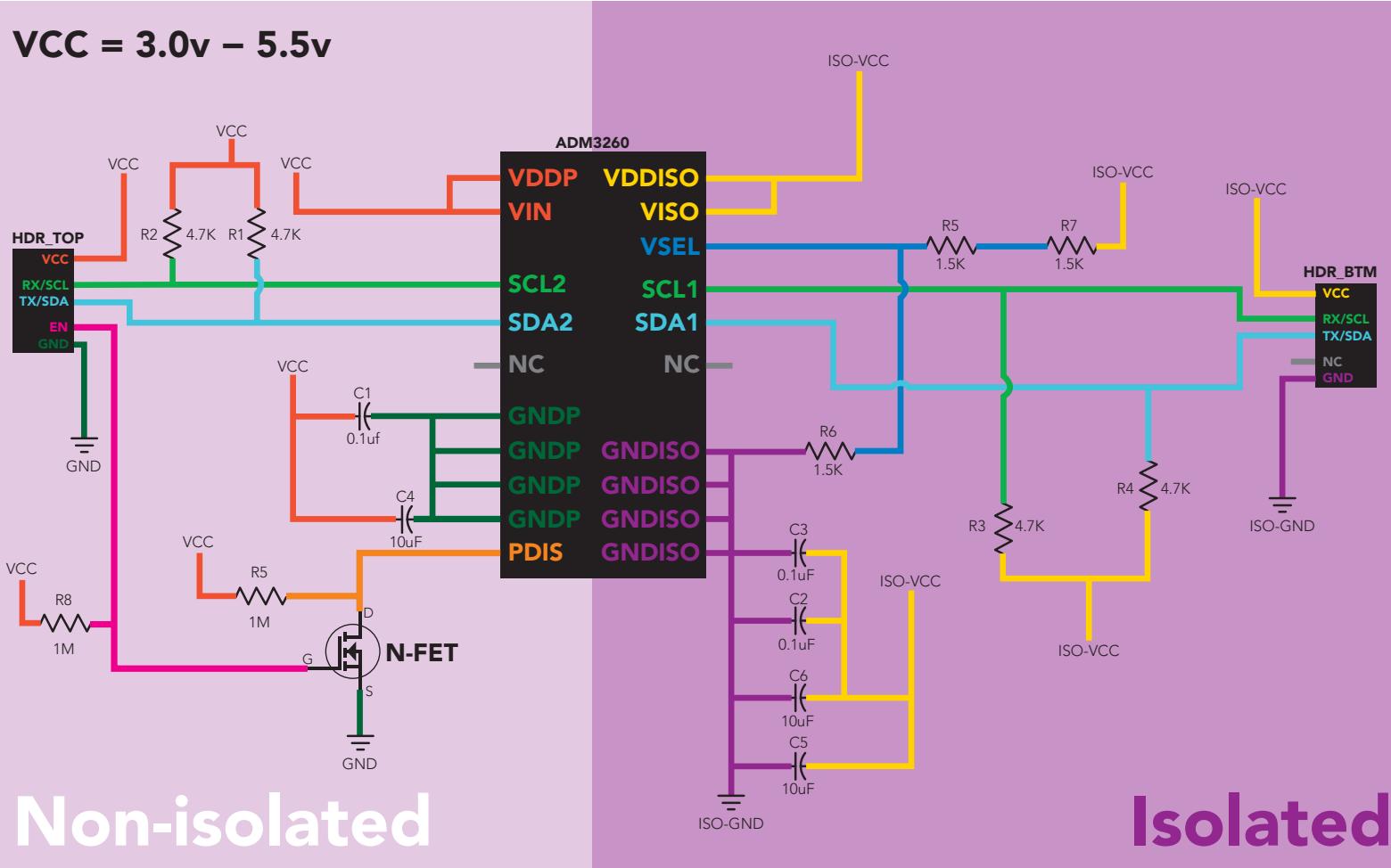
**Without isolation, Conductivity readings  
will effect Dissolved Oxygen accuracy.**

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  $4.7\text{k}\Omega$  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.**

**VCC = 3.0v – 5.5v**

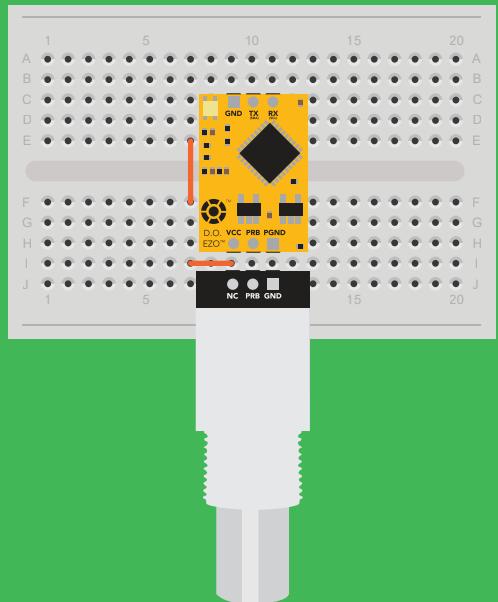


**Non-isolated**

**Isolated**

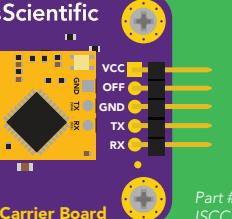
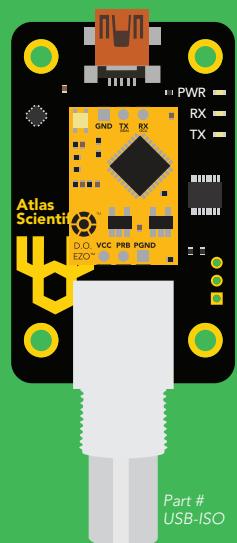
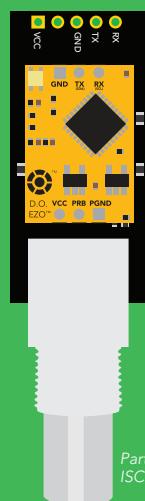
# ✓ Correct wiring

## Bread board



Bread board  
via USB

## Carrier board



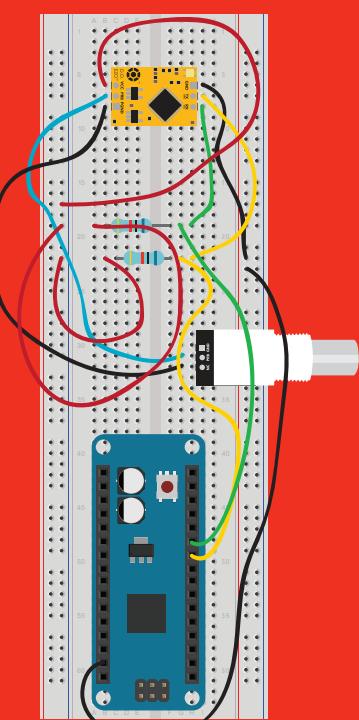
Electrically Isolated  
EZO™ Carrier Board

# ✗ Incorrect wiring

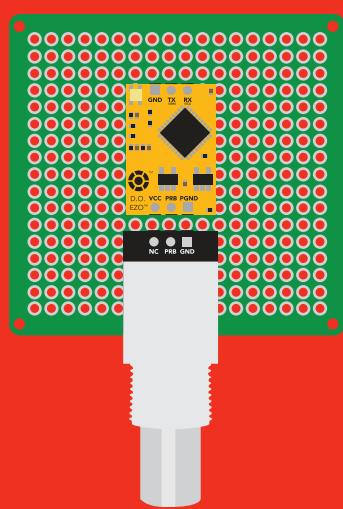
## Extended leads



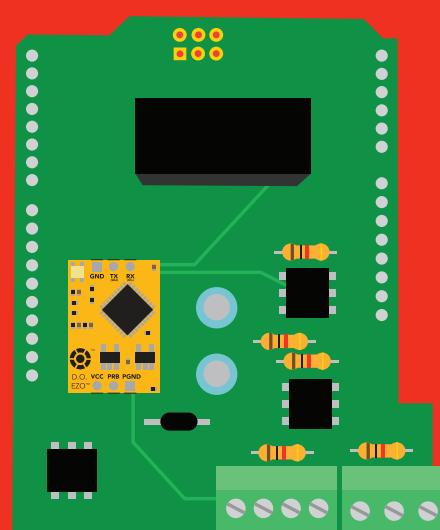
## Sloppy setup



## Perfboards or Protoboards



## \*Embedded into your device



**NEVER**  
use Perfboards  
or Protoboards

\*Only after you are familiar  
with EZO™ circuits operation

# Calibration theory

Simple calibration

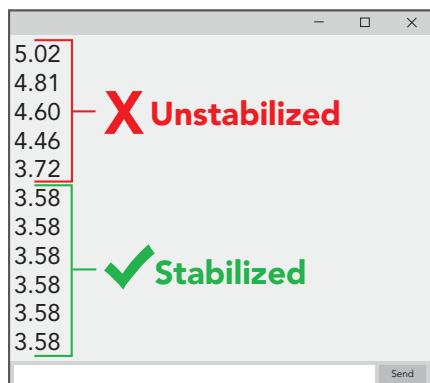
**UART**  
mode

Continuous readings

Advanced calibration

**I<sup>2</sup>C**  
mode

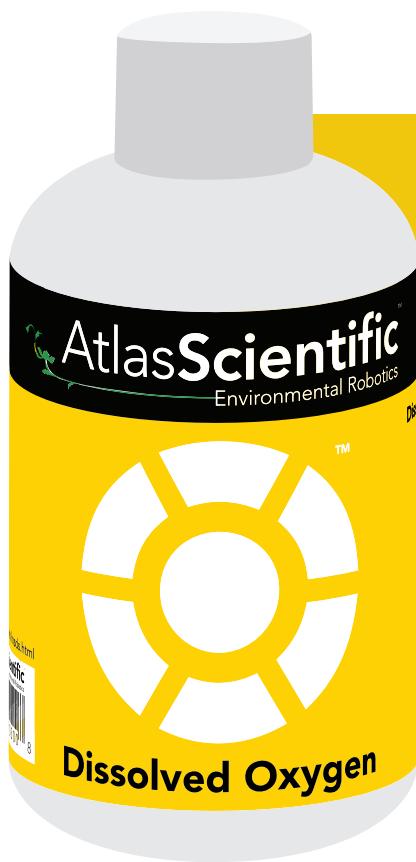
Continuously request readings



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 I<sup>2</sup>C mode after calibration **will not** affect the stored calibration. If the device must be calibrated in I<sup>2</sup>C mode be sure to **continuously request readings** so you can see the output from the probe.



The Atlas Scientific EZO™ Dissolved Oxygen circuit, has a flexible calibration protocol, allowing for **single point** or **dual point (optional)** calibration.

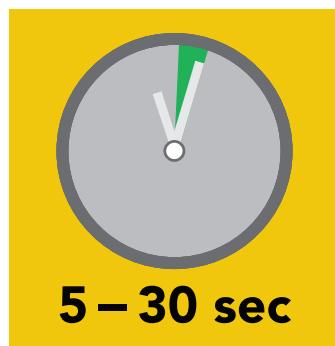
**Calibrate first, compensate later.**

Temperature, salinity and pressure compensation values have no effect on calibration.



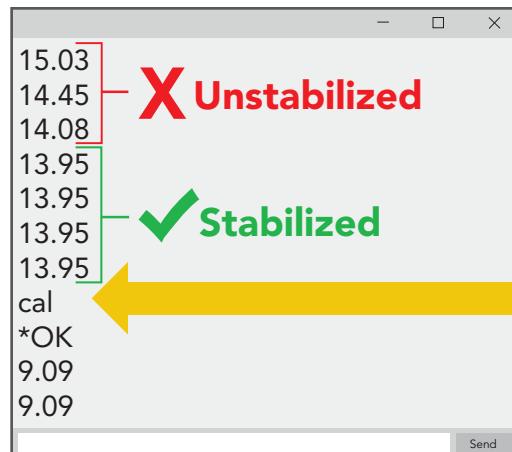
# Single point calibration

Carefully pull off and discard the cap from the Dissolved Oxygen probe. Let the Dissolved Oxygen probe sit, exposed to air until the readings stabilize. (small movement from one reading to the next is normal).



## Do not unscrew!

Cap is only used to protect the probe during shipping.



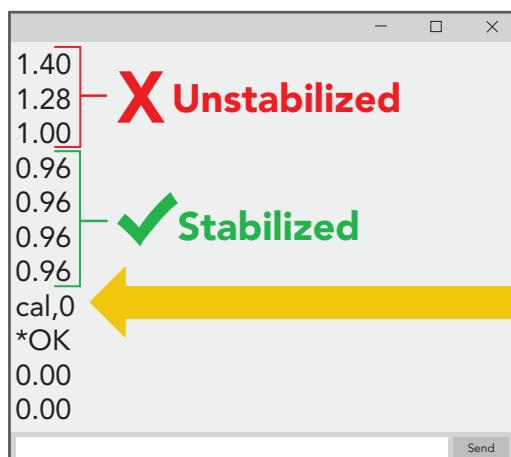
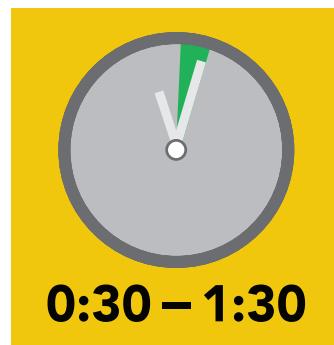
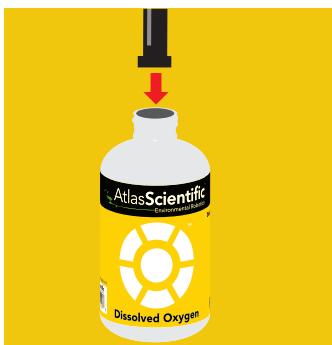
Once the readings have stabilized,  
issue the calibration command.  
**"cal"**

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)*

## Dual point calibration (optional)

Only perform this calibration if you require accurate readings **below** 1.0 mg/L

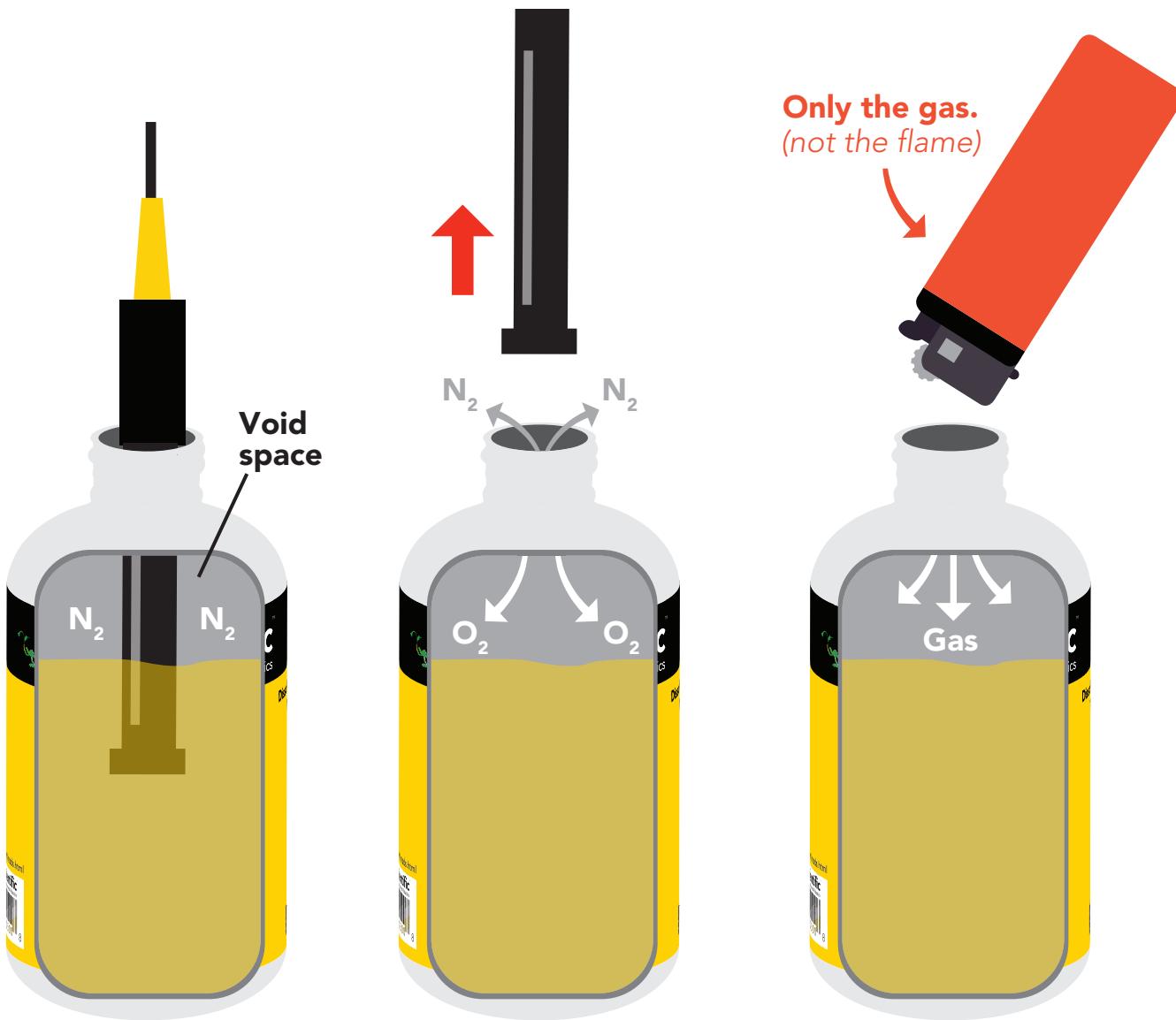
After you have calibrated the EZO™ Dissolved Oxygen circuit using the "Cal" command; Place the probe into the Zero Dissolved Oxygen calibration solution and stir the probe around to remove trapped air (which could cause readings to go high). Let the probe sit in Zero D.O. calibration solution until readings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



Once the readings have stabilized,  
issue the calibration command.  
**"cal,0"**

# How to preserve the Zero D.O. calibration solution

Oxygen is everywhere. The Zero D.O. calibration solution has been designed to chemically absorb oxygen. Once the bottle has been opened the test solution has been exposed to oxygen and will slowly stop working.



Inside each bottle of the calibration solution is a small amount of nitrogen gas that helps displace oxygen out of the bottle during the filling process. When the Dissolved Oxygen probe is removed from the bottle, oxygen will enter the bottle and begin to dissolve into the solution.

In order slow down this process, fill the void space of the bottle with any gas (other than oxygen) to preserve the calibration solution. Gas from a lighter works great if other gases are currently unobtainable.

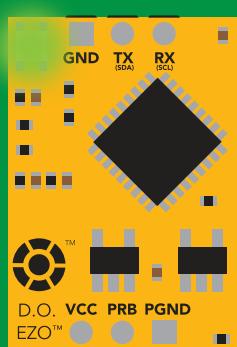
# Default state

# UART mode

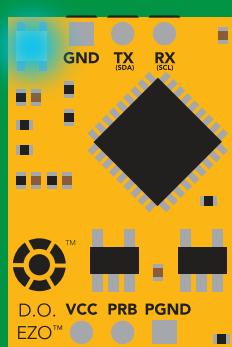
Baud	9,600
Readings	continuous
Speed	1 reading per second
Temperature compensation	20 °C
Salinity compensation	0 (Fresh water)
Pressure compensation	101.3 kPa (Sea level)
LED	on



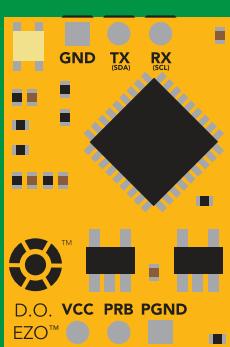
1,000 ms



Green  
Standby



Cyan  
Taking reading



Transmitting

 Available data protocols

**UART**

**Default**

**I<sup>2</sup>C**

 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 parameters  
Enable/disable response codes  
Hardware switch to I<sup>2</sup>C mode  
LED control  
Protocol lock  
Software switch to I<sup>2</sup>C mode

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

Find  
Pressure compensation  
Salinity compensation  
Sleep mode  
Temperature compensation

# UART mode

8 data bits      no parity  
1 stop bit      no flow control

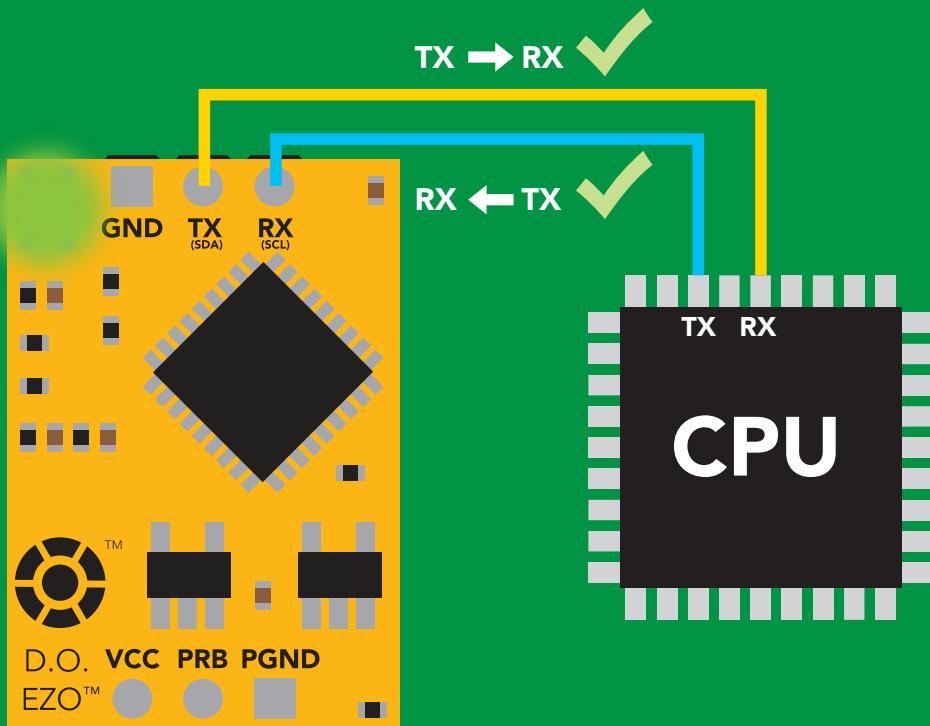
Baud 300  
1,200  
2,400  
**9,600 default**  
19,200  
38,400  
57,600  
115,200

**RX** Data in

**TX** Data out

**Vcc** 3.3V – 5.5V

VCC  
0V



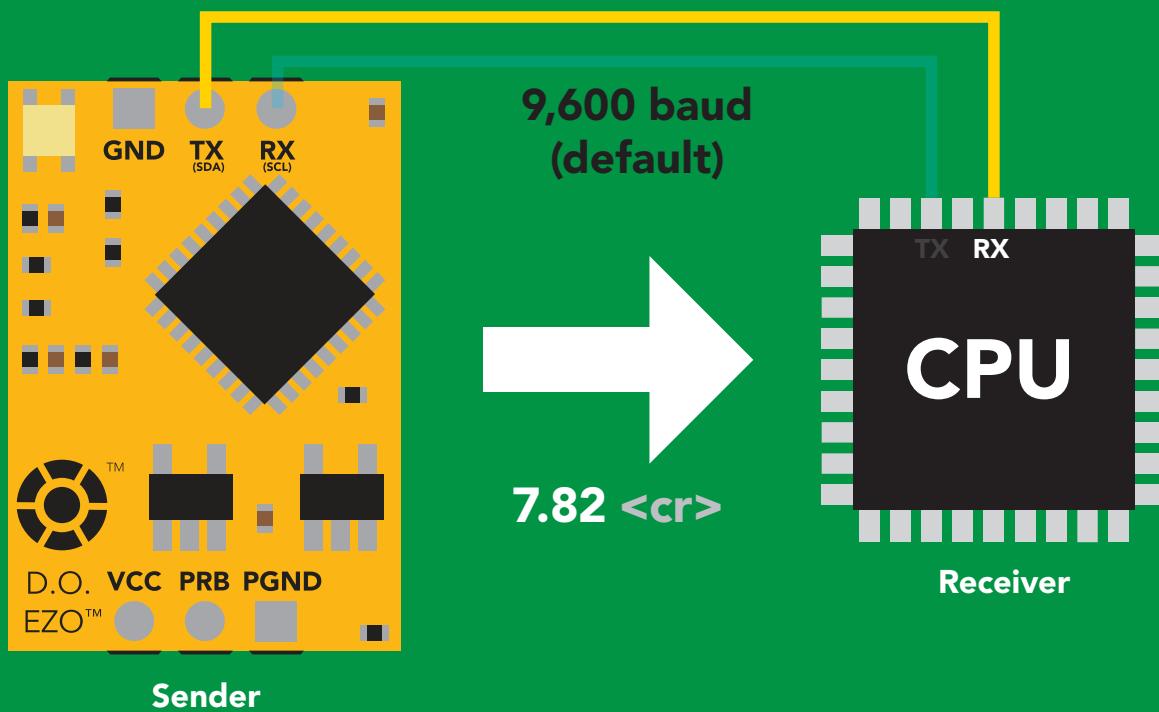
## Data format

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

Data type	<b>floating point</b>
Decimal places	mg/L = 2 % sat = 1
Smallest string	<b>4 characters</b>
Largest string	<b>16 characters</b>

# Receiving data from device

2 parts



## Advanced

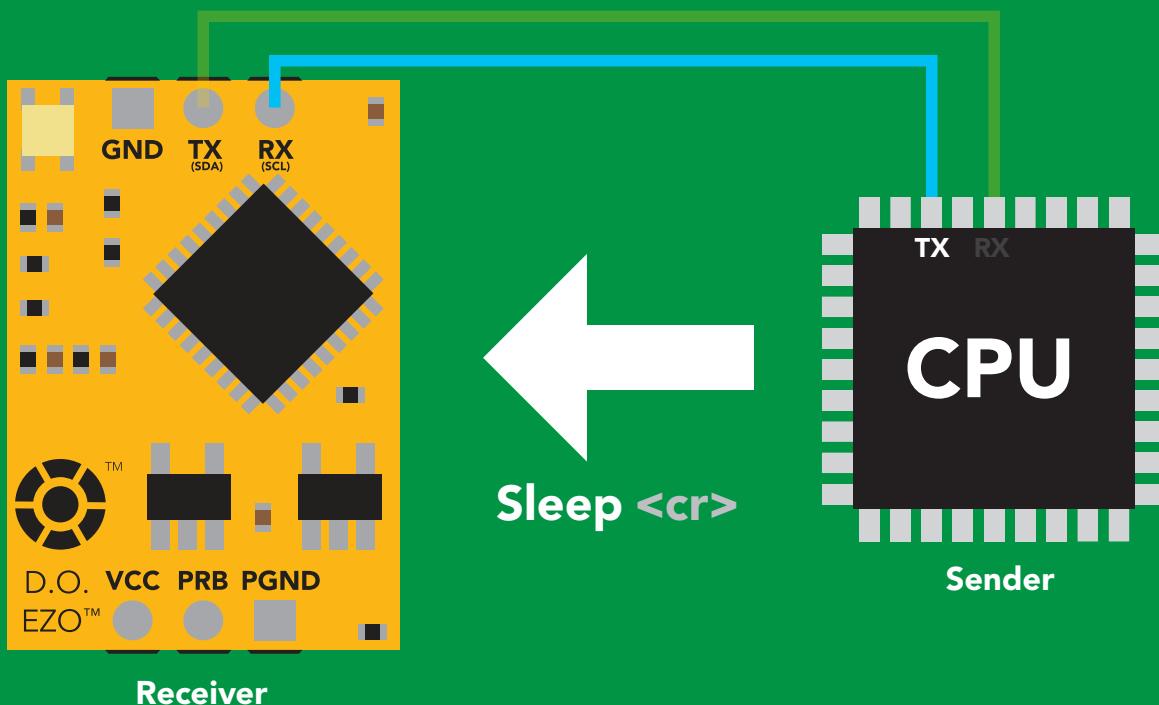
ASCII: 7 . 8 2 <cr>

Hex: 37 2E 38 32 0D

Dec: 55 46 56 50 13

# Sending commands to device

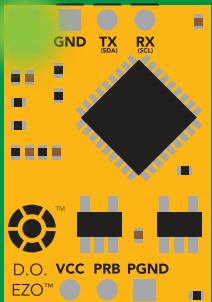
2 parts



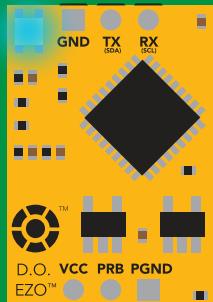
## Advanced

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

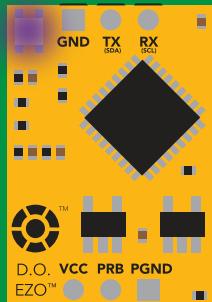
# LED color definition



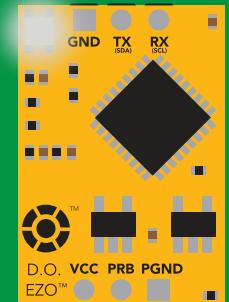
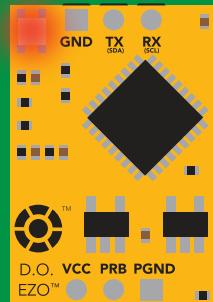
**Green**  
UART standby



**Cyan**  
Taking reading



**Purple**  
Changing baud rate  
**Red**  
Command not understood



**White**  
Find

<b>5V</b>	LED ON <b>+0.4 mA</b>
<b>3.3V</b>	<b>+0.2 mA</b>

# UART mode

## command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function	Default state
Baud	change baud rate	pg. 37 9,600
C	enable/disable continuous reading	pg. 23 enabled
Cal	performs calibration	pg. 25 n/a
Export	export calibration	pg. 26 n/a
Factory	enable factory reset	pg. 39 n/a
Find	finds device with blinking white LED	pg. 22 n/a
i	device information	pg. 33 n/a
I2C	change to I <sup>2</sup> C mode	pg. 40 not set
Import	import calibration	pg. 27 n/a
L	enable/disable LED	pg. 21 enabled
Name	set/show name of device	pg. 32 not set
O	enable/disable parameters	pg. 31 mg/L
P	pressure compensation	pg. 30 101.3 kPa
Plock	enable/disable protocol lock	pg. 38 disabled
R	returns a single reading	pg. 24 n/a
S	salinity compensation	pg. 29 n/a
Sleep	enter sleep mode/low power	pg. 36 n/a
Status	retrieve status information	pg. 35 n/a
T	temperature compensation	pg. 28 20°C
*OK	enable/disable response codes	pg. 34 enable

# LED control

## Command syntax

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

L,0 <cr> LED off

L,? <cr> LED state on/off?

### Example

L,1 <cr>

\*OK <cr>

L,0 <cr>

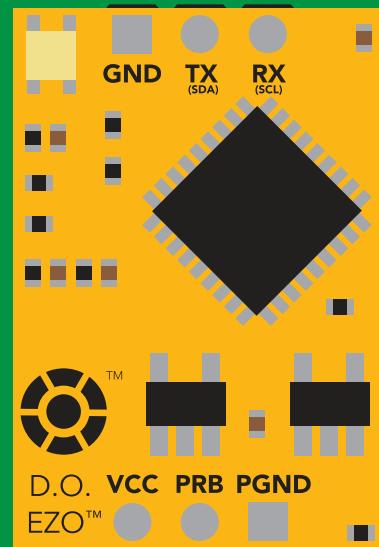
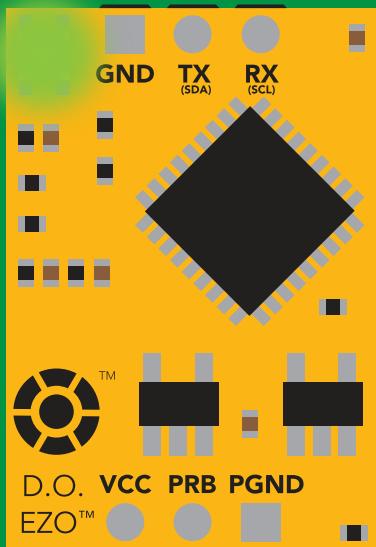
\*OK <cr>

L,? <cr>

?L,1 <cr> or ?L,0 <cr>

\*OK <cr>

### Response



L,1

L,0

# Find

## Command syntax

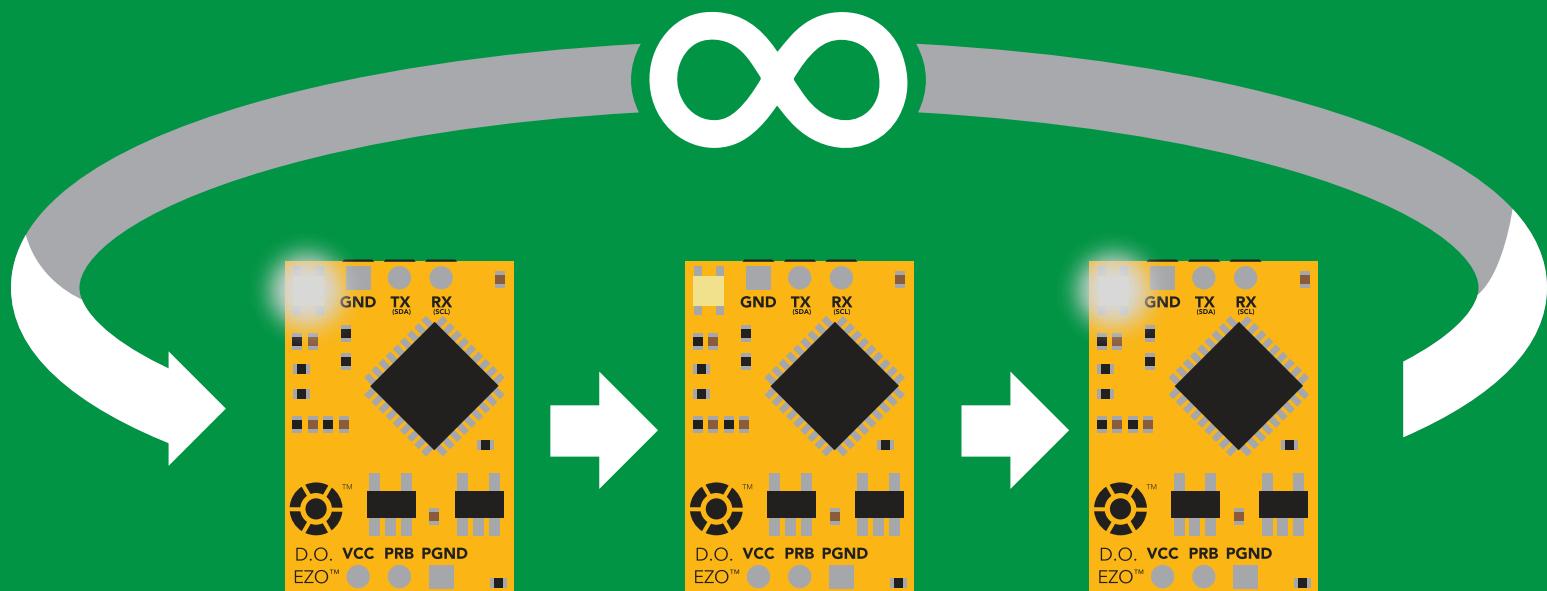
This command will disable continuous mode  
Send any character or command to terminate find.

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

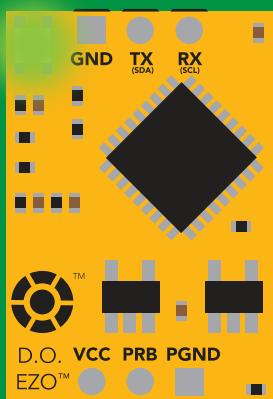
R <cr> takes single reading

### Example      Response

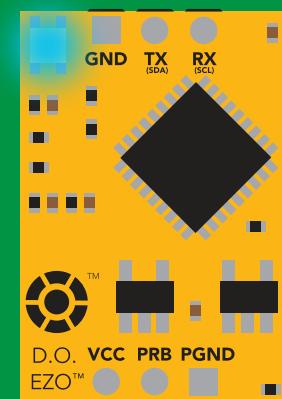
R <cr>

7.82 <cr>

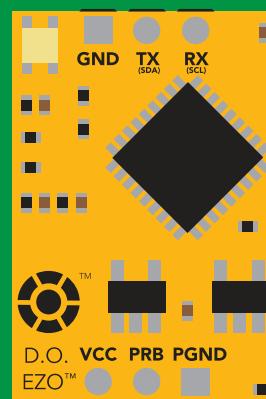
\*OK <cr>



Green  
Standby



Cyan  
Taking reading



Transmitting



# Calibration

## Command syntax

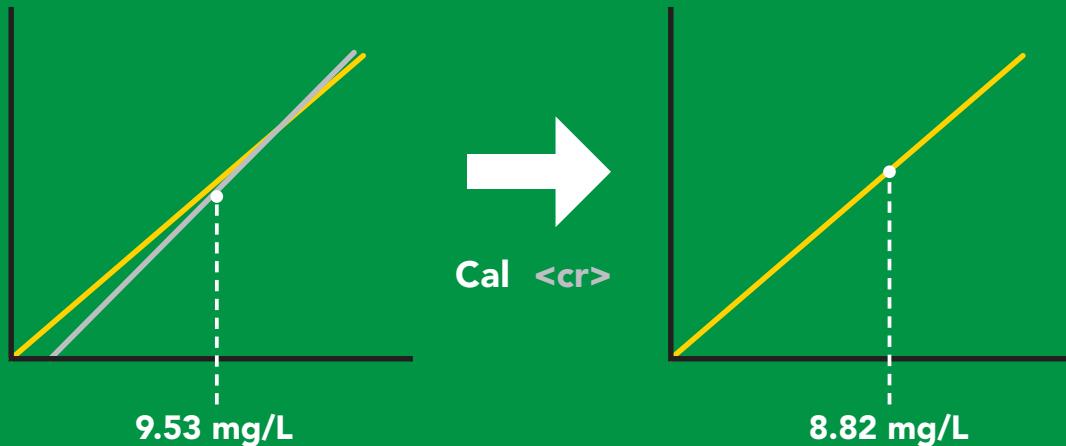
The EZO™ Dissolved Oxygen circuit uses single and/or two point calibration

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> or ?Cal,2 <cr> *OK <cr>

single point  
two point



# 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

Export,? <cr>

## Response

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>

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>

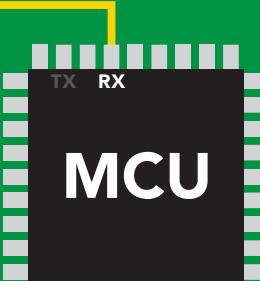
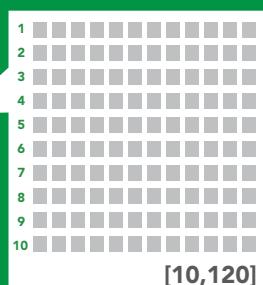
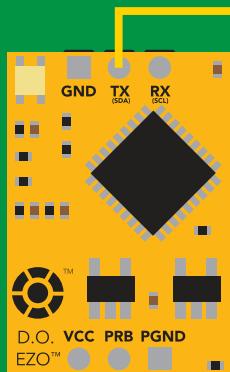
6F 6C 20 67 75 79 <cr> (10 of 10)

Export <cr>

\*DONE

Disabling \*OK simplifies this process

Export <cr>



# 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

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

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

⋮

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

## Response

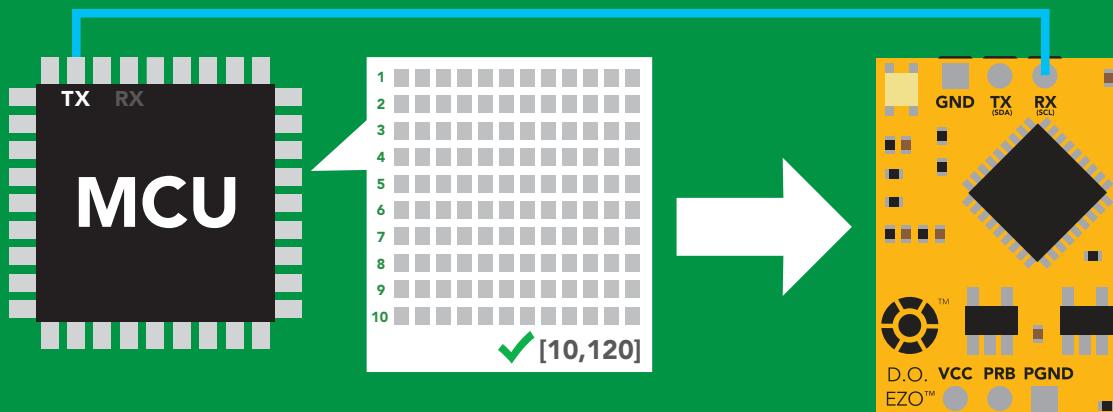
\*OK <cr>

\*OK <cr>

⋮

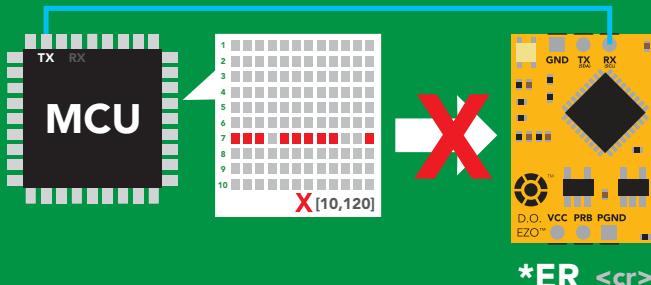
\*OK <cr>

**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

T,19.5 <cr>

## Response

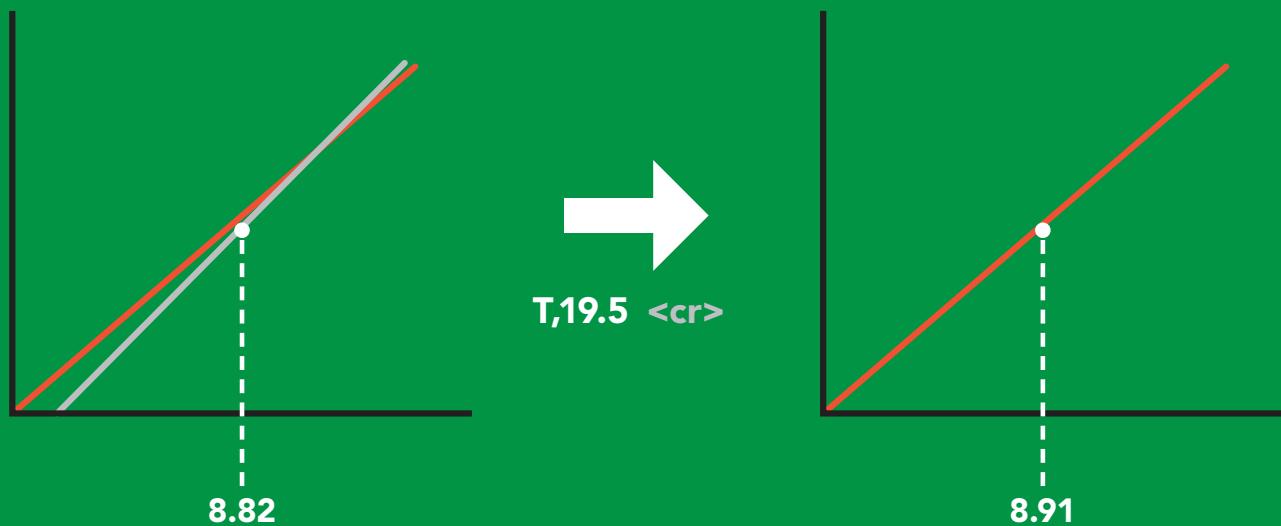
\*OK <cr>

RT,19.5 <cr>

\*OK <cr>  
8.91 <cr>

T,? <cr>

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



# Salinity compensation

## Command syntax

Default value = 0  $\mu\text{s}$

If the conductivity of your water is less than 2,500 $\mu\text{S}$  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

**S,50000 <cr>**

## Response

\*OK <cr>

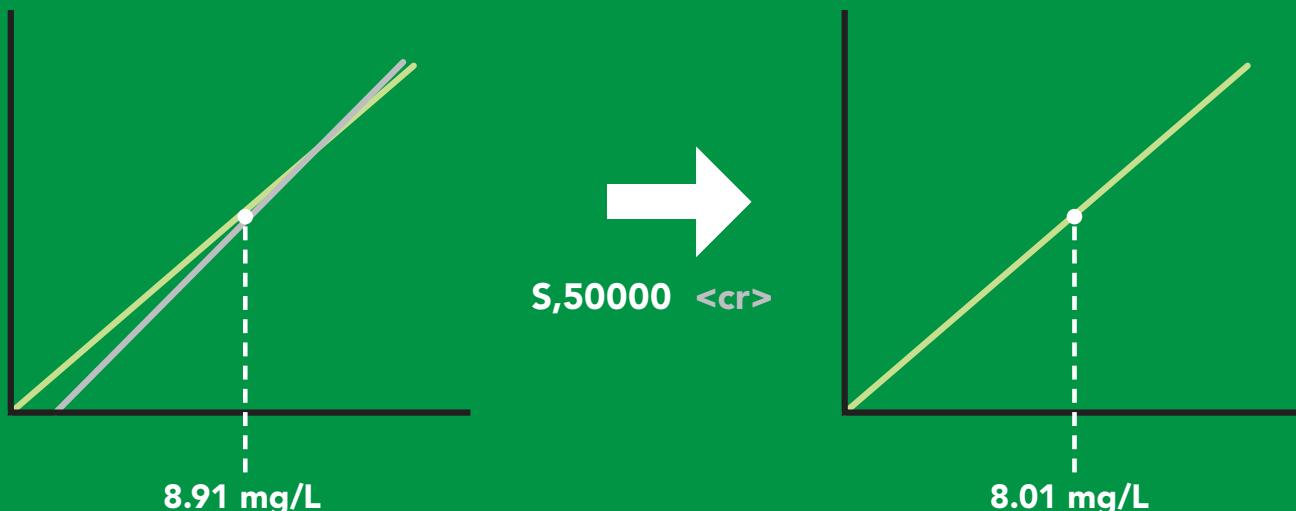
**S,37.5,ppt <cr>**

**S,? <cr>**

\*OK <cr>

?S,50000, $\mu\text{S}$  <cr> or ?S,37.5,ppt <cr>

\*OK <cr>



# Pressure compensation

## Command syntax

Default value = 101.3 kPa  
This parameter can be omitted if the water is less than 10 meters deep

P,n <cr> n = any value in kPa

P,? <cr> compensated pressure value?

## Example

P,90.25 <cr>

## Response

\*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

O,mg,1 / O,mg,0 <cr>

### Response

\*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".

# 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

Name,zzt <cr>

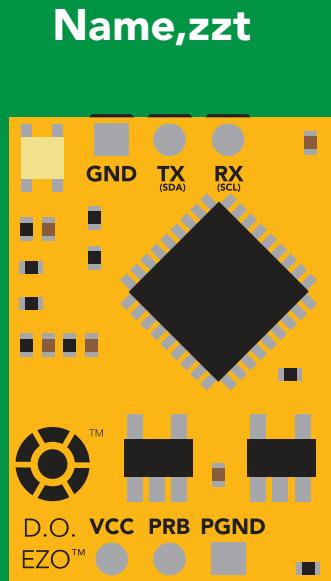
\*OK <cr>

Name,? <cr>

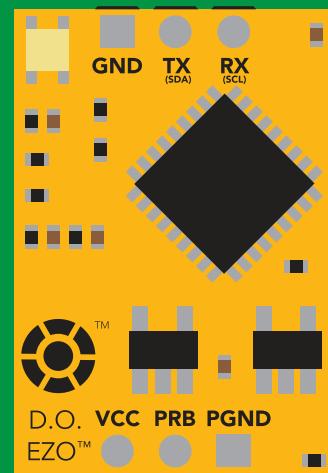
?Name,zzt <cr>

\*OK <cr>

## Response



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>	<b>no response, *OK disabled</b>
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

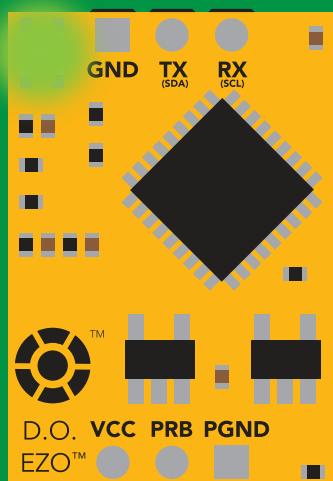
**Sleep <cr>**

**\*OK <cr>**  
**\*SL <cr>**

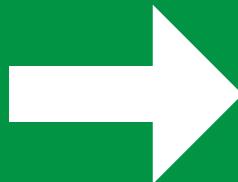
### Any command

**\*WA <cr>** wakes up device

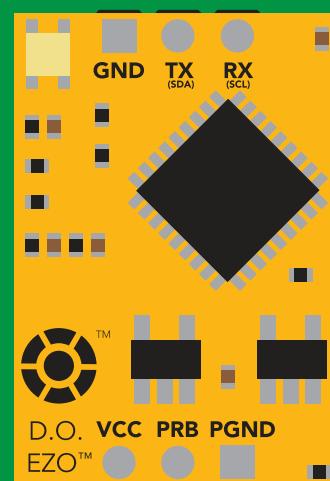
	STANDBY	SLEEP
<b>5V</b>	<b>13.1 mA</b>	<b>0.66 mA</b>
<b>3.3V</b>	<b>12 mA</b>	<b>0.3 mA</b>



**Standby**  
**13.1 mA**



**Sleep <cr>**



**Sleep**  
**0.66 mA**

# Change baud rate

## Command syntax

Baud,n <cr> change baud rate

### Example

Baud,38400 <cr>

### Response

\*OK <cr>

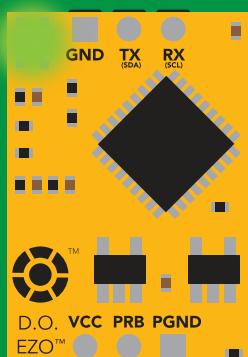
### Example

Baud,? <cr>

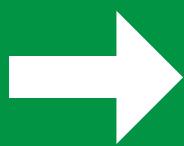
?Baud,38400 <cr>

\*OK <cr>

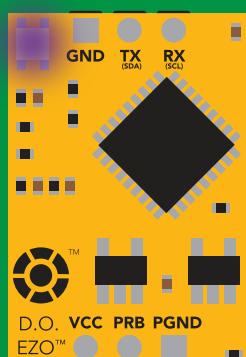
n = [ 300  
1200  
2400  
**9600 default**  
19200  
38400  
57600  
115200 ]



Standby



Baud,38400 <cr>

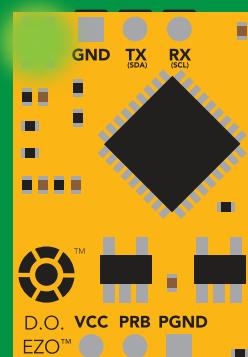


Changing  
baud rate

\*OK <cr>



(reboot)



Standby

# 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

Plock,1 <cr>

\*OK <cr>

Plock,0 <cr>

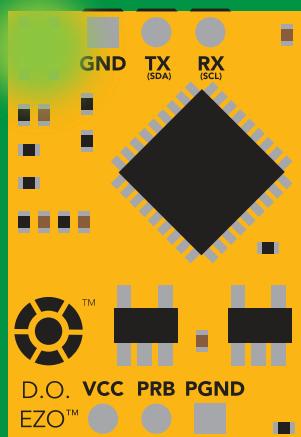
\*OK <cr>

Plock,? <cr>

?Plock,1 <cr> or ?Plock,0 <cr>

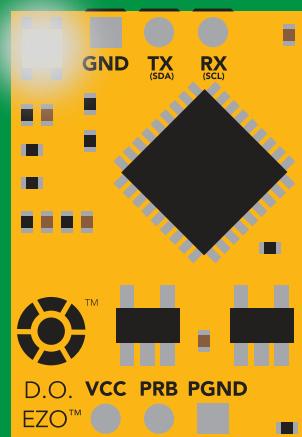
## Response

Plock,1



\*OK <cr>

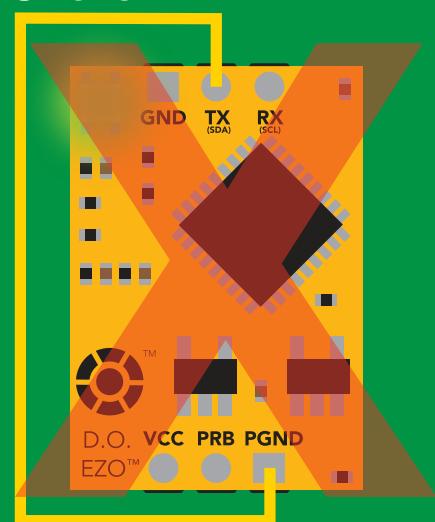
I2C,100



cannot change to I<sup>2</sup>C

\*ER <cr>

Short



cannot change to I<sup>2</sup>C

# Factory reset

## Command syntax

Clears calibration  
LED on  
"\*OK" enabled

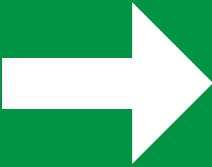
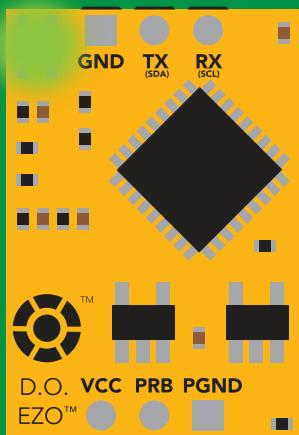
Factory <cr> enable factory reset

## Example Response

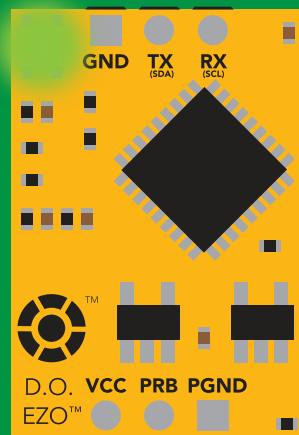
Factory <cr>

\*OK <cr>

Factory <cr>



(reboot)



\*OK <cr>

\*RS <cr>

\*RE <cr>

Baud rate will not change

# Change to I<sup>2</sup>C mode

## Command syntax

Default I<sup>2</sup>C address 97 (0x61)

I<sup>2</sup>C,n <cr> sets I<sup>2</sup>C address and reboots into I<sup>2</sup>C mode

n = any number 1 – 127

## Example      Response

I<sup>2</sup>C,100 <cr>

\*OK (reboot in I<sup>2</sup>C mode)

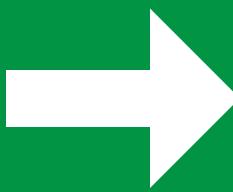
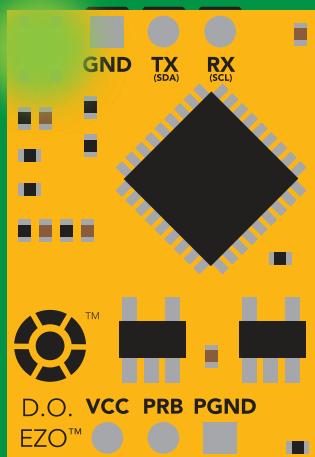
## Wrong example

I<sup>2</sup>C,139 <cr> n ≠ 127

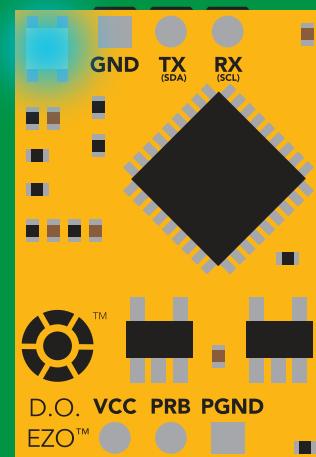
## Response

\*ER <cr>

I<sup>2</sup>C,100



(reboot)



Green  
\*OK <cr>

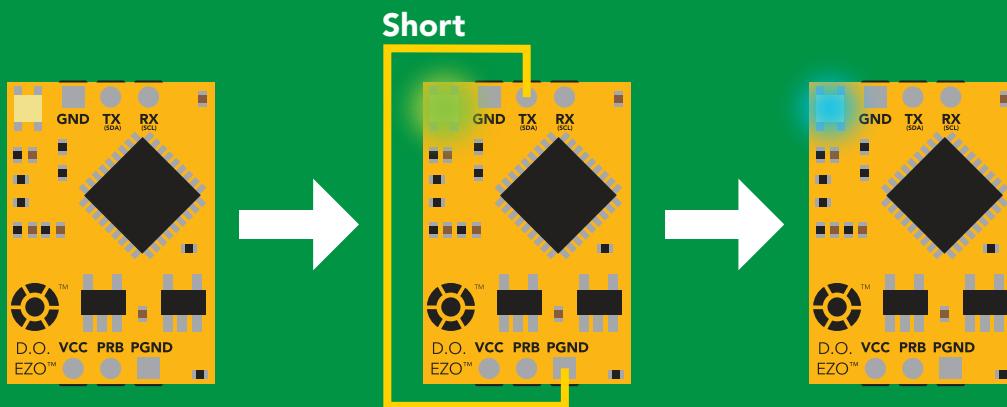
Blue  
now in I<sup>2</sup>C mode

# Manual switching to I<sup>2</sup>C

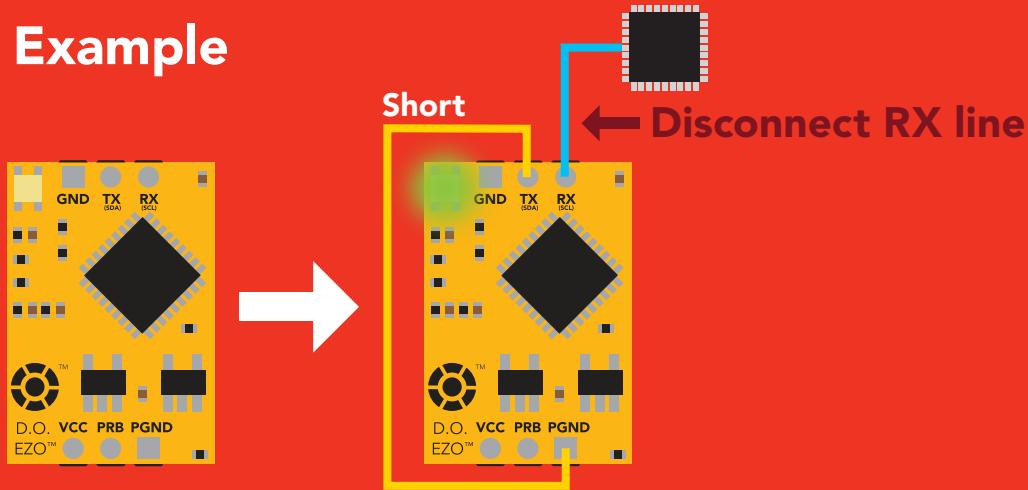
- Make sure Plock is set to 0
- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to PGND
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from Green to Blue
- Disconnect ground (power off)
- Reconnect all data and power

Manually switching to I<sup>2</sup>C will set the I<sup>2</sup>C address to 97 (0x61)

## Example



## Wrong Example



# I<sup>2</sup>C mode

The I<sup>2</sup>C 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<sup>2</sup>C mode [click here](#)

## Settings that are retained if power is cut

- Calibration
- Change I<sup>2</sup>C 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

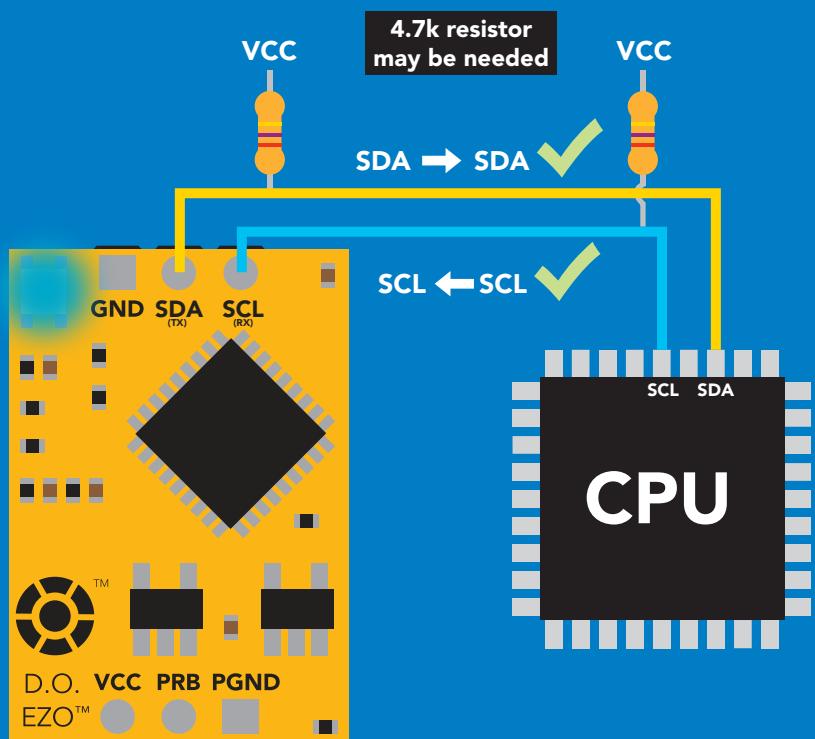
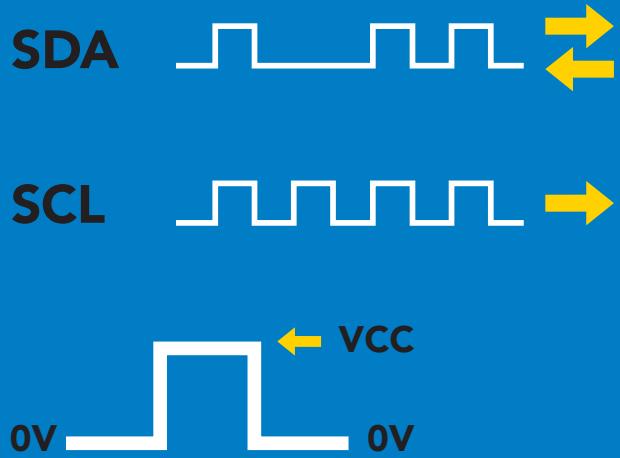
- Find
- Pressure compensation
- Salinity compensation
- Sleep mode
- Temperature compensation

# I<sup>2</sup>C mode

I<sup>2</sup>C address (0x01 – 0x7F)  
**97 (0x61) default**

V<sub>cc</sub> 3.3V – 5.5V

Clock speed 100 – 400 kHz



## Data format

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

Data type floating point  
mg/L = 2  
% sat = 1  
Decimal places  
Smallest string 4 characters  
Largest string 16 characters

# Sending commands to device

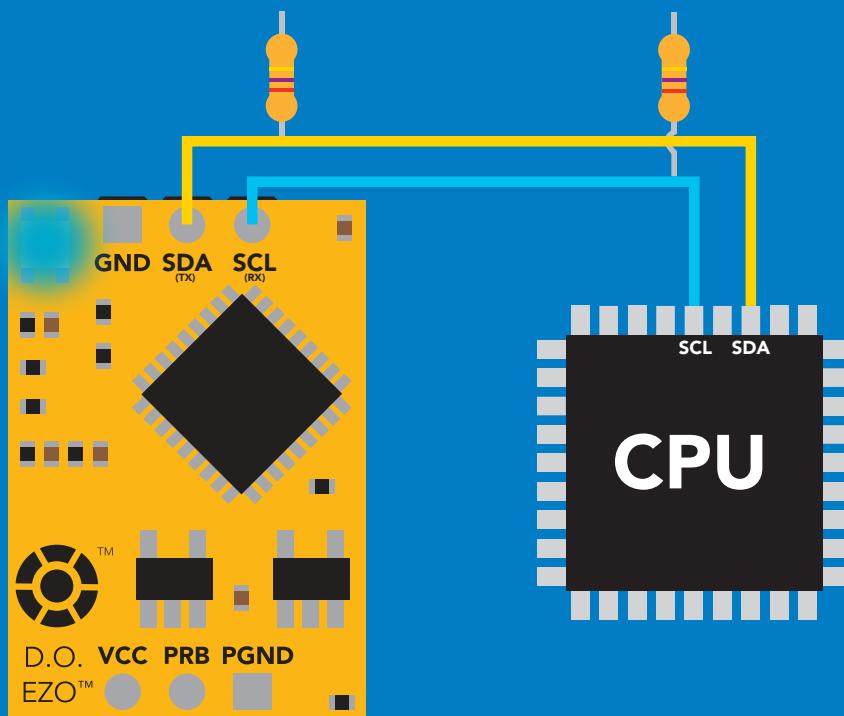
5 parts



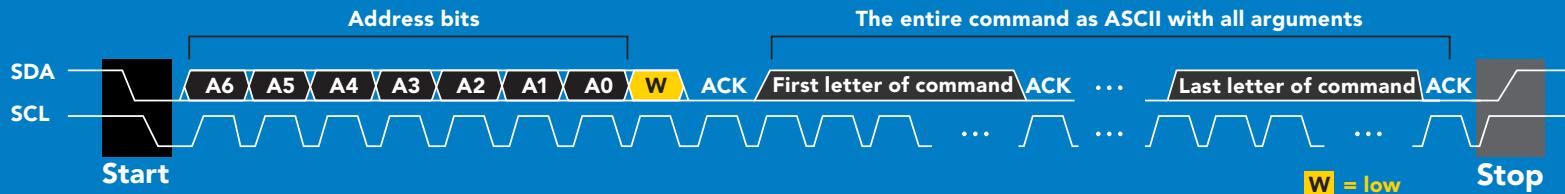
## Example

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

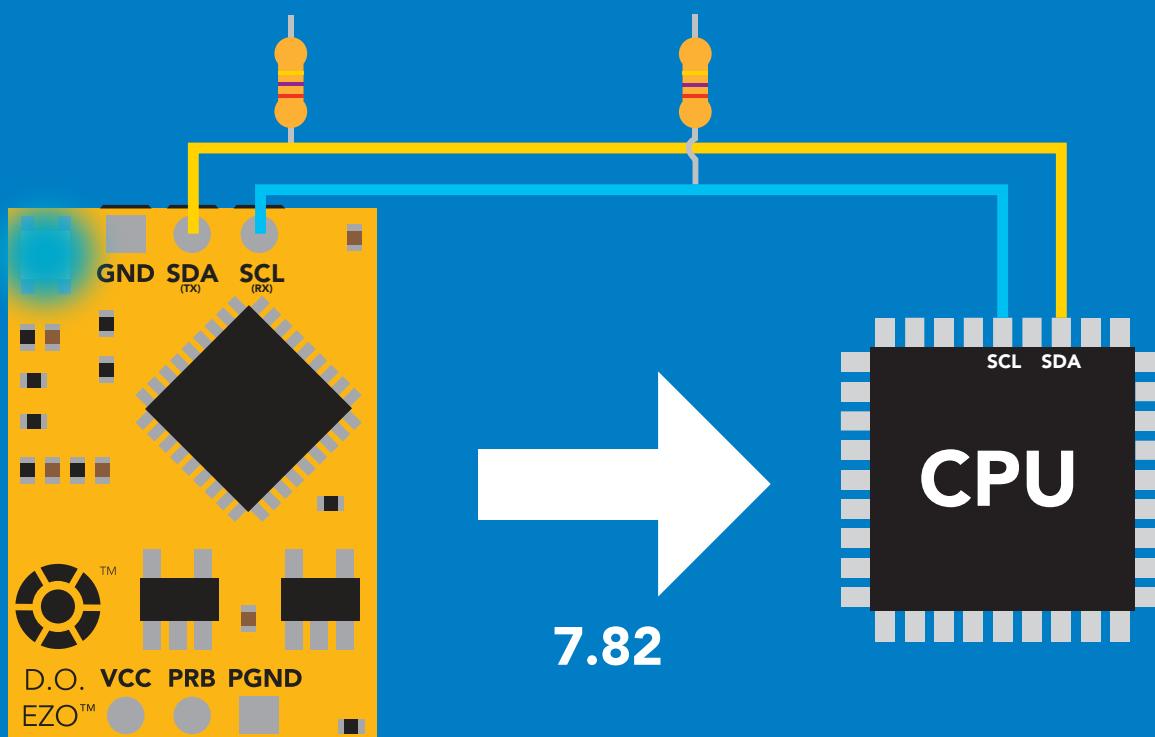
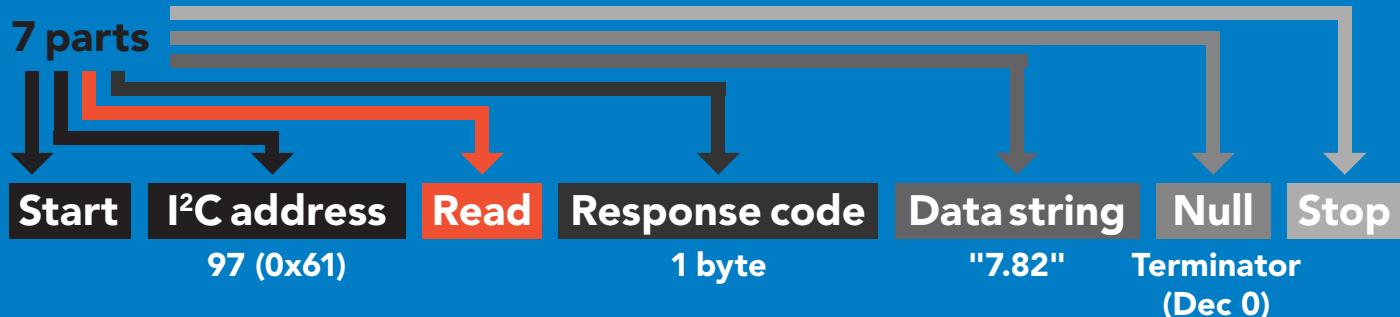
I<sup>2</sup>C address      Command



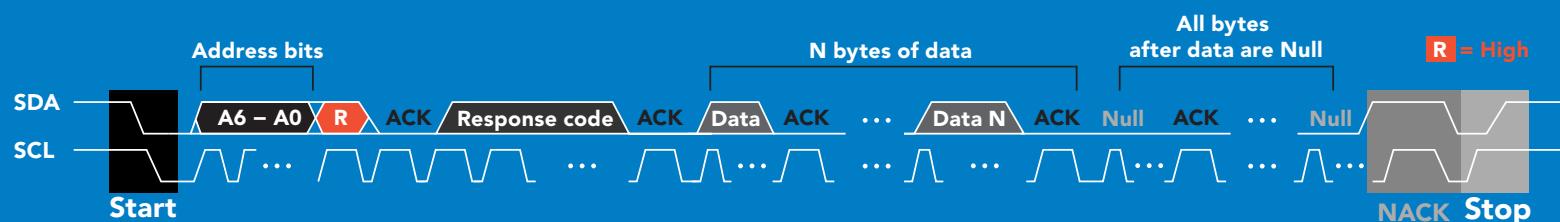
## Advanced



# Requesting data from device



## Advanced



1 55 46 56 50 0 = 7.82

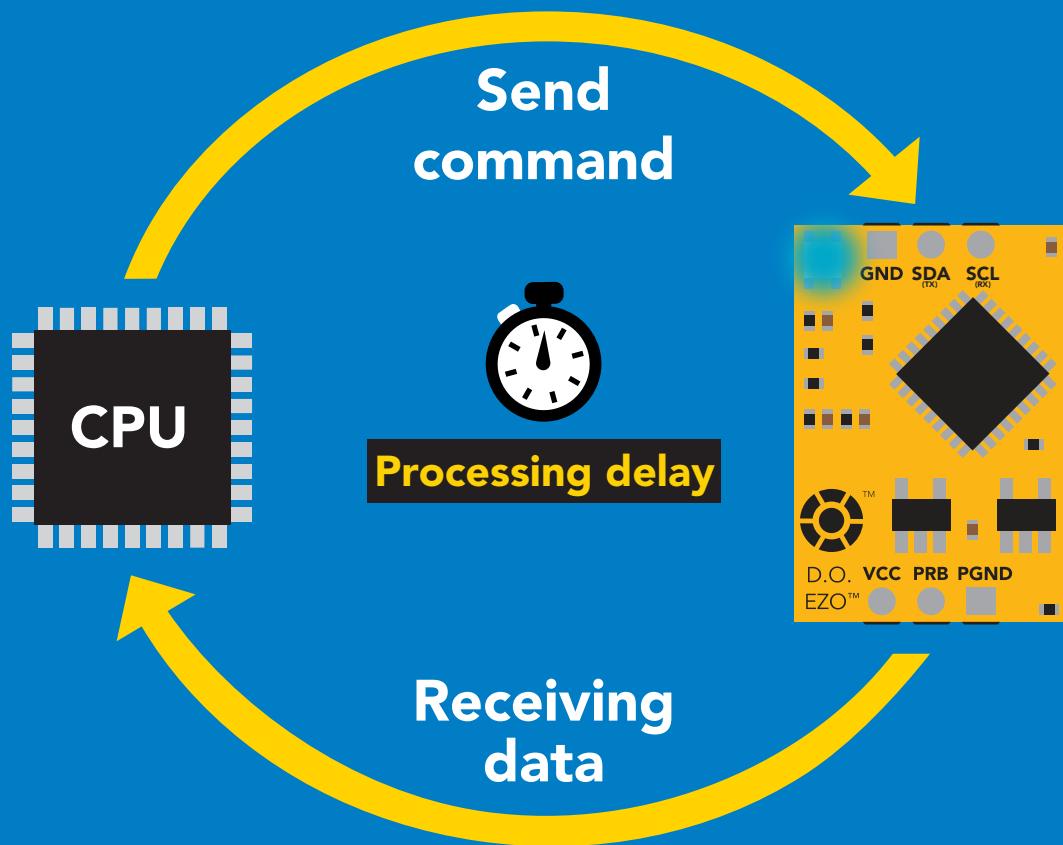
Dec                      Dec

ASCII

# 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;
```

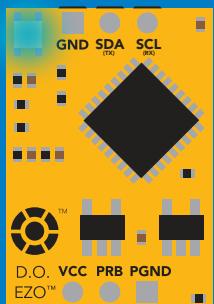
The response code will always be 254, if you do not wait for the processing delay.

### Response codes

Single byte, not string

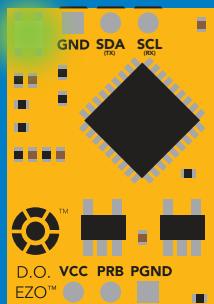
255	no data to send
254	still processing, not ready
2	syntax error
1	successful request

# LED color definition



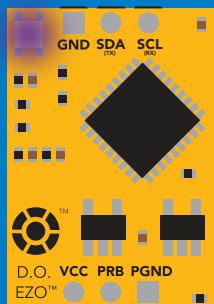
**Blue**

I<sup>2</sup>C standby



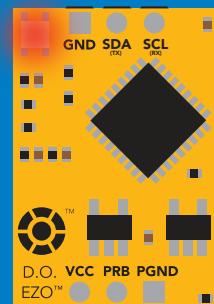
**Green**

Taking reading



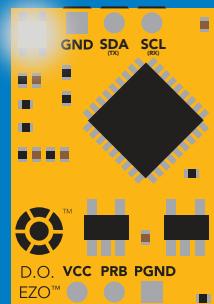
**Purple**

Changing I<sup>2</sup>C address



**Red**

Command not understood



**White**

Find

<b>5V</b>	LED ON <b>+0.4 mA</b>
<b>3.3V</b>	<b>+0.2 mA</b>

# I<sup>2</sup>C mode

## command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function	
Baud	change back to UART mode	pg. 65
Cal	performs calibration	pg. 52
Export	export calibration	pg. 53
Factory	enable factory reset	pg. 64
Find	finds device with blinking white LED	pg. 50
i	device information	pg. 59
I2C	change I <sup>2</sup> C address	pg. 63
Import	import calibration	pg. 54
L	enable/disable LED	pg. 49
O	removing parameters	pg. 58
P	pressure compensation	pg. 57
Plock	enable/disable protocol lock	pg. 62
R	returns a single reading	pg. 51
S	salinity compensation	pg. 56
Sleep	enter sleep mode/low power	pg. 61
Status	retrieve status information	pg. 60
T	temperature compensation	pg. 55

# LED control

## Command syntax

300ms  processing delay

L,1 LED on **default**

L,0 LED off

L,? LED state on/off?

## Example

L,1

  
Wait 300ms

**1**  
Dec      **0**  
Null

L,0

  
Wait 300ms

**1**  
Dec      **0**  
Null

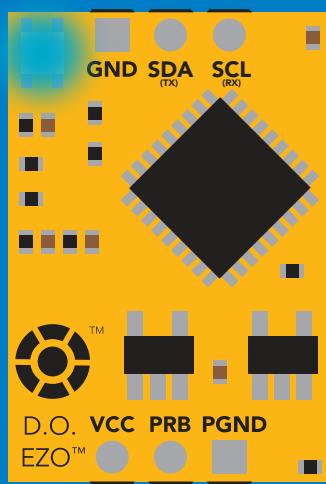
L,?

  
Wait 300ms

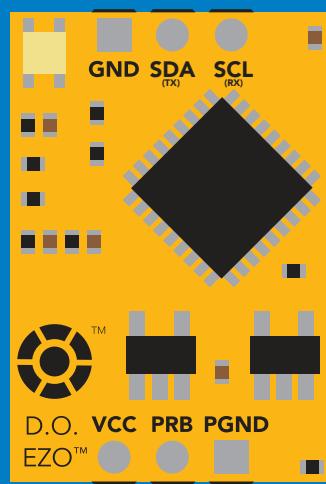
**1**   **?L,1**   **0**  
Dec   ASCII   Null

or

**1**   **?L,0**   **0**  
Dec   ASCII   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

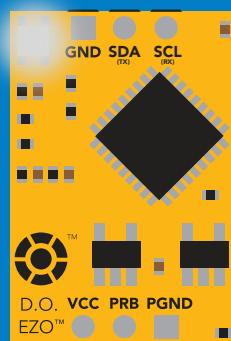
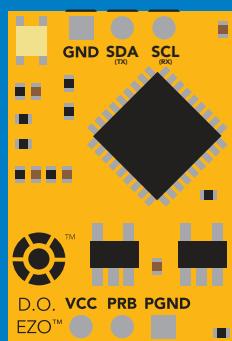
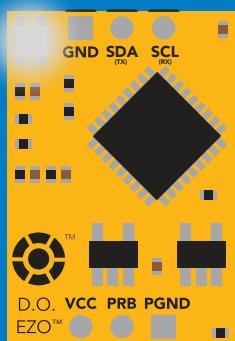
## Example

## Response

Find

  
Wait 300ms

1 Dec Null



# Taking reading

## Command syntax

600ms  processing delay

R return 1 reading

Example

Response

R



Wait 600ms

1

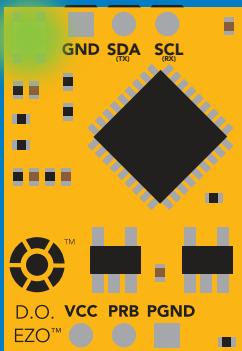
7.82

0

Dec

ASCII

Null



Green

Taking reading



Wait 600ms

# 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 0 or 1 Dec ?Cal,1 0  
or 1 Dec ?Cal,2 0

# 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

**Export,?**

## Response



1 Dec 10,120 ASCII 0 Null

### Response breakdown

10, 120

# of strings to export # of bytes to export

Export strings can be up to 12 characters long

**Export**



1 Dec 59 6F 75 20 61 72 0 Null

(1 of 10)

**Export**



1 Dec 65 20 61 20 63 6F 0 Null

(2 of 10)

**(7 more)**

⋮

**Export**



1 Dec 6F 6C 20 67 75 79 0 Null

(10 of 10)

**Export**



1 Dec \*DONE 0 Null

# 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)

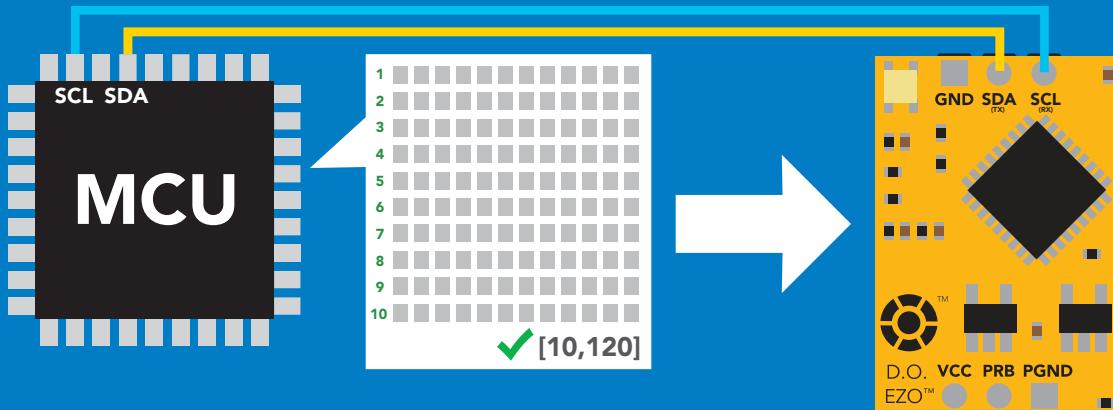
## Response

 1 0 Null  
Wait 300ms

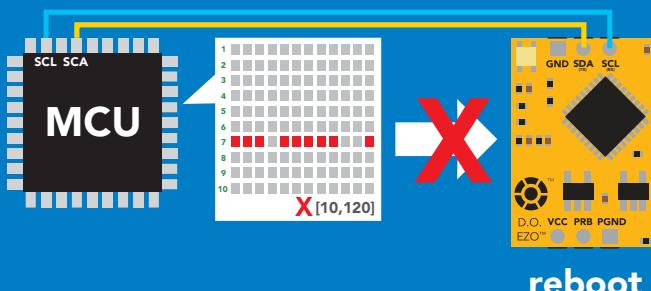
 1 0 Null  
Wait 300ms

⋮  
 1 0 Null  
Wait 300ms

Import,n



system will reboot



\* If one of the imported strings is not correctly entered, the device will not accept the import 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 n = any value; floating point or int    300ms  processing delay
- T,? compensated temperature value?
- RT,n set temperature compensation and take a reading\*

This is a new command  
for firmware V2.13

## Example

T,19.5

## Response

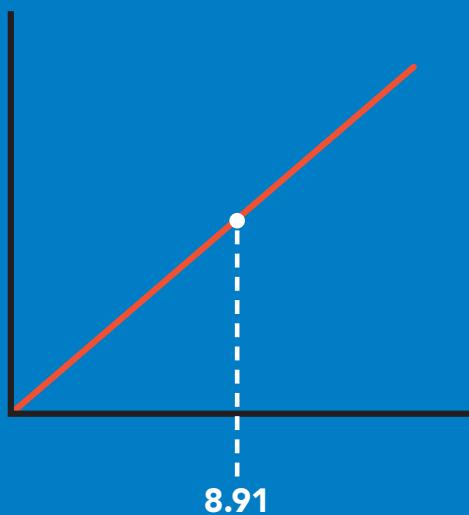
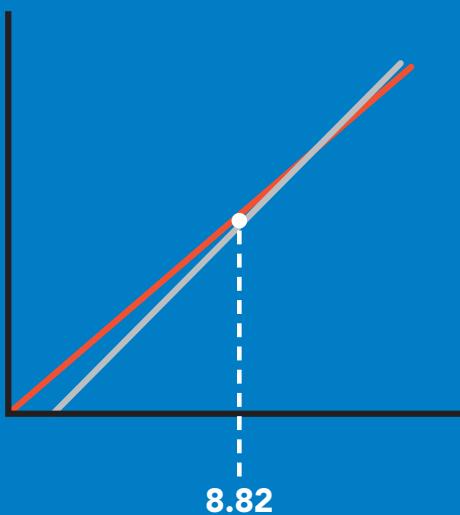
 Wait 300ms    1 Dec 0 Null

RT,19.5

 Wait 900ms    1 Dec 8.91 0 Null

T,?

 Wait 300ms    1 Dec ?T,19.5 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

**S,50000**

## Response

 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** **?S,37.5,ppt** ASCII    **0** Null

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

# Pressure compensation

## Command syntax

P,n   n = any value in kPa

P,?   compensated pressure value?

300ms  processing delay

This parameter can be omitted if the water is less than 10 meters deep

## Example

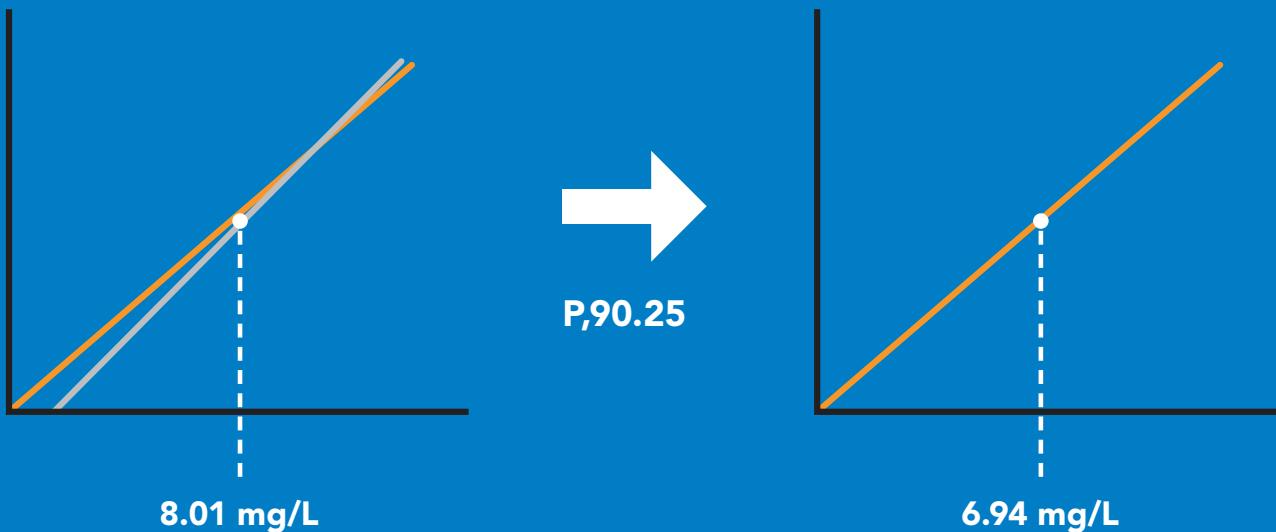
P,90.25

## Response

 Wait 300ms  
1 Dec 0 Null

P,?

 Wait 300ms  
1 Dec ?,P,90.25 0 ASCII 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



1 Dec 0 Null

enable / disable mg/L

O,%,1 / O,%,0



1 Dec 0 Null

enable / disable percent saturation

O,?



1 Dec 0 Null

? ,O,%,mg 0 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".

# 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  
↑      ↑  
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

?Status,P,5.038

Dec

ASCII

0

Null

## Response breakdown

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

### Restart codes

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

# Sleep mode/low power

## Command syntax

**Sleep** enter sleep mode/low power

Send any character or command to awaken device.

### Example

### Response

**Sleep**

**no response**

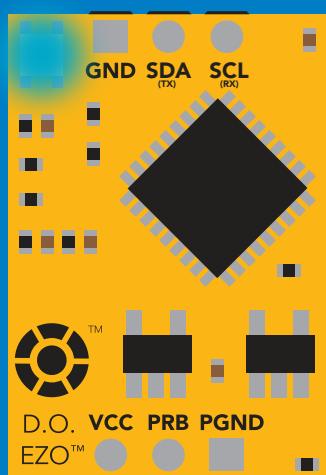
Do not read status byte after issuing sleep command.

**Any command**

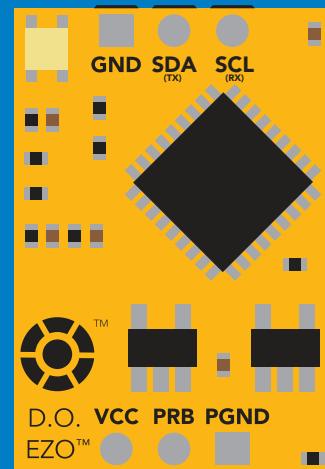
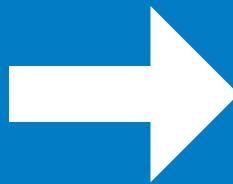
**wakes up device**

5V	STANDBY	SLEEP
	<b>13.1 mA</b>	<b>0.66 mA</b>

3.3V	12 mA	0.3 mA
------	-------	--------



Standby



Sleep

# Protocol lock

## Command syntax

300ms  processing delay

Plock,1 enable Plock

Locks device to I<sup>2</sup>C mode.

Plock,0 disable Plock

default

Plock,? Plock on/off?

## Example

Plock,1

 Wait 300ms

1  
Dec  
Null

Plock,0

 Wait 300ms

1  
Dec  
Null

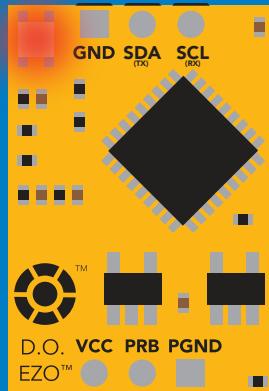
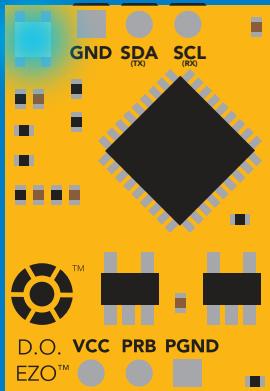
Plock,?

 Wait 300ms

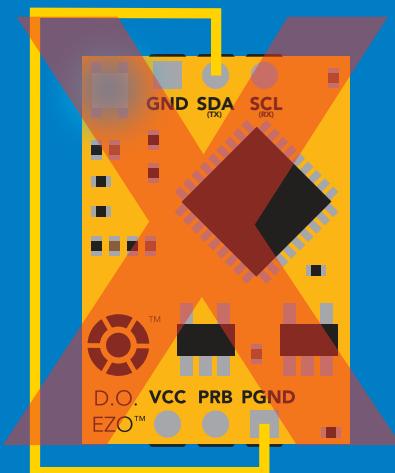
1  
Dec  
?Plock,1  
ASCII  
0  
Null

Plock,1

Baud, 9600



cannot change to UART



cannot change to UART

# I<sup>2</sup>C address change

## Command syntax

300ms  processing delay

I<sup>2</sup>C,n sets I<sup>2</sup>C address and reboots into I<sup>2</sup>C mode

## Example Response

I<sup>2</sup>C,100

device reboot

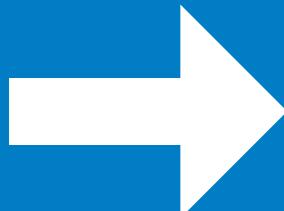
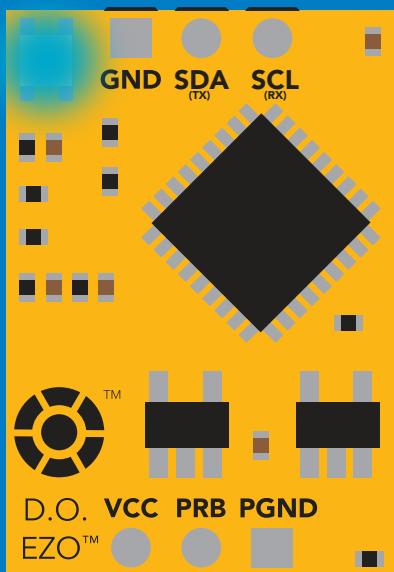
### Warning!

Changing the I<sup>2</sup>C address will prevent communication between the circuit and the CPU until your CPU is updated with the new I<sup>2</sup>C address.

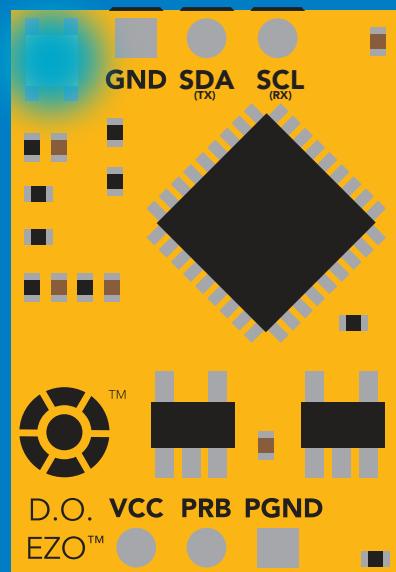
Default I<sup>2</sup>C address is 97 (0x61).

n = any number 1 – 127

I<sup>2</sup>C,100



(reboot)



# Factory reset

## Command syntax

Factory reset will not take the device out of I<sup>2</sup>C mode.

Factory enable factory reset

I<sup>2</sup>C address will not change

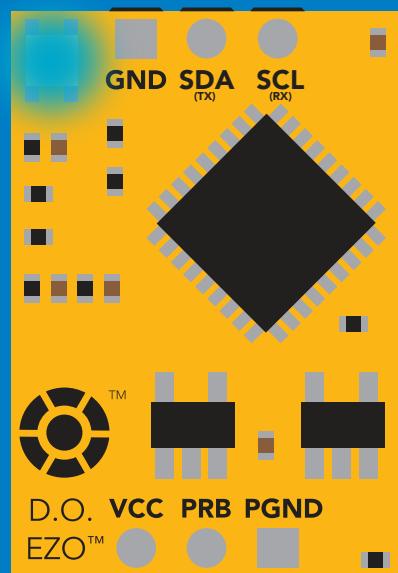
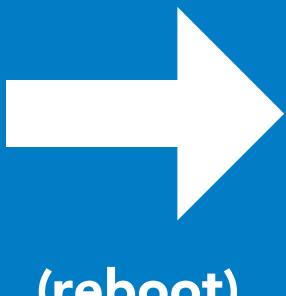
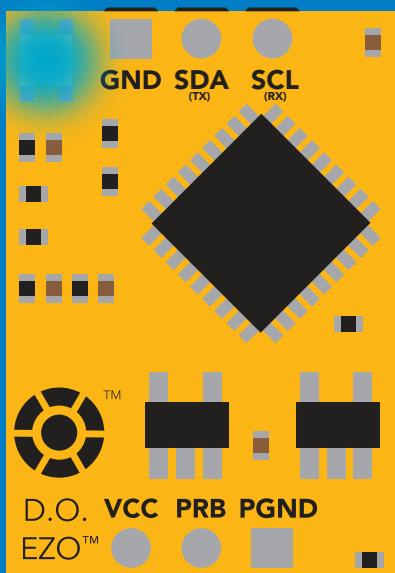
## Example Response

Factory

device reboot

Clears calibration  
LED on  
Response codes enabled

Factory



# Change to UART mode

## Command syntax

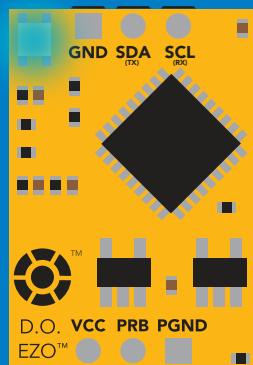
Baud,n switch from I<sup>2</sup>C to UART

### Example      Response

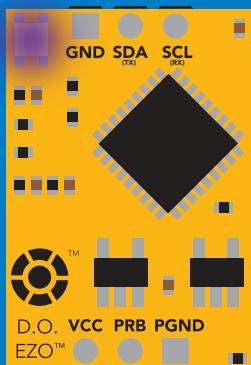
Baud,9600

reboot in UART mode

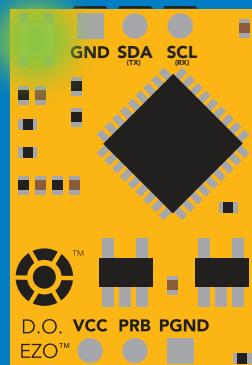
n = [ 300  
1200  
2400  
9600  
19200  
38400  
57600  
115200 ]



→  
Baud,9600



→  
(reboot)

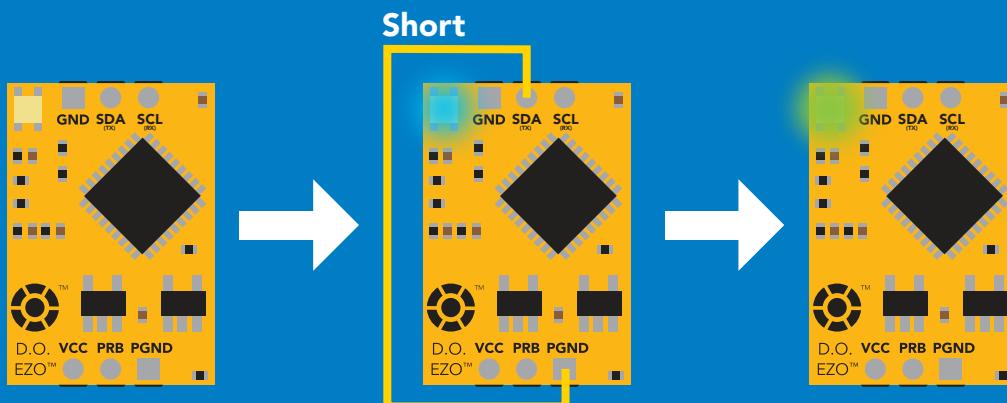


Changing to  
UART mode

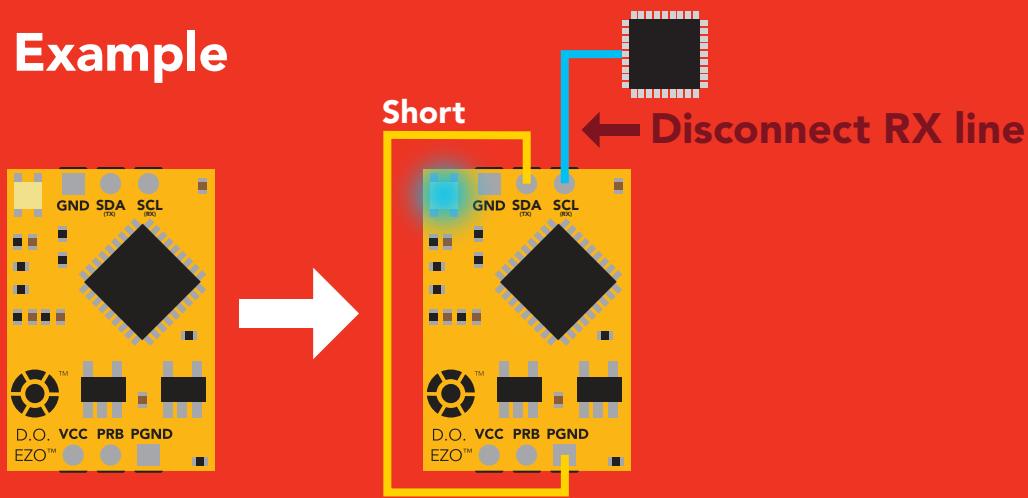
# Manual switching to UART

- 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

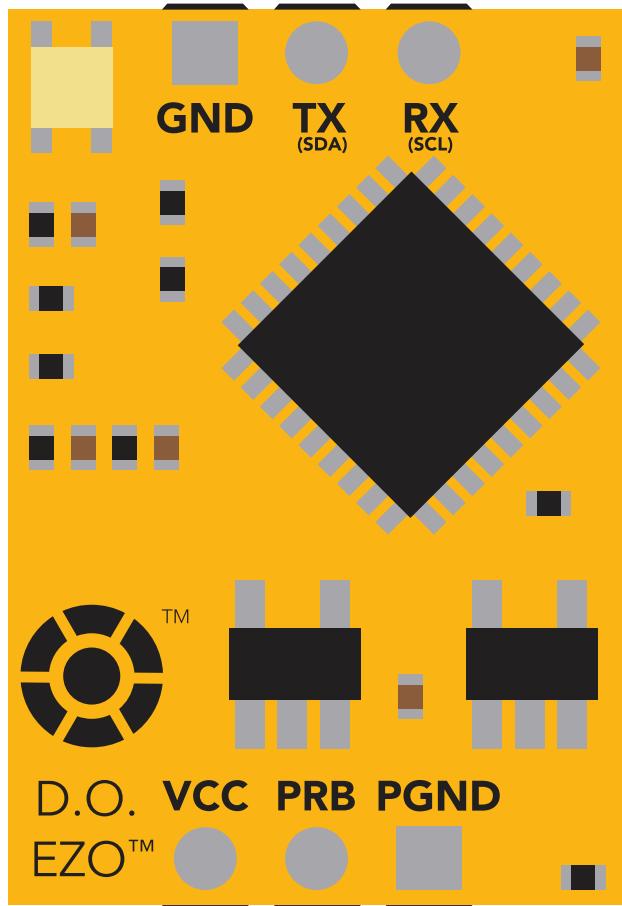
## Example



## Wrong Example



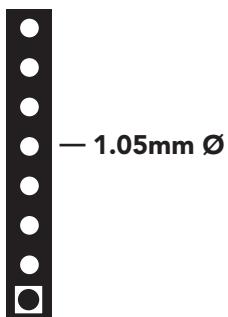
# EZO™ circuit footprint



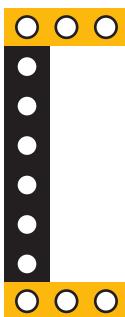
17.78mm  
(0.7")

2.54mm  
(0.1")

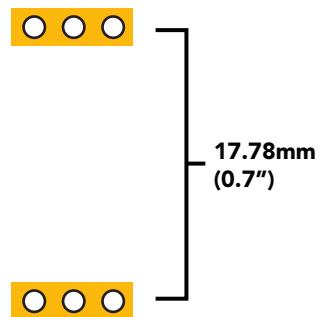
**1** In your CAD software place a 8 position header.



**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 5.4

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<sup>2</sup>C).  
Added firmware information to Firmware update list.

## **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<sup>2</sup>C).

## **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<sup>2</sup>C).

"Export/Import calibration" pages 25 (UART) & 51 (I<sup>2</sup>C).

Added new feature to continuous 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 reflect 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<sup>2</sup>C bug (Dec 1, 2014)

- Fixed I<sup>2</sup>C bug where the circuit may inappropriately respond when other I<sup>2</sup>C 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.

# 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.