

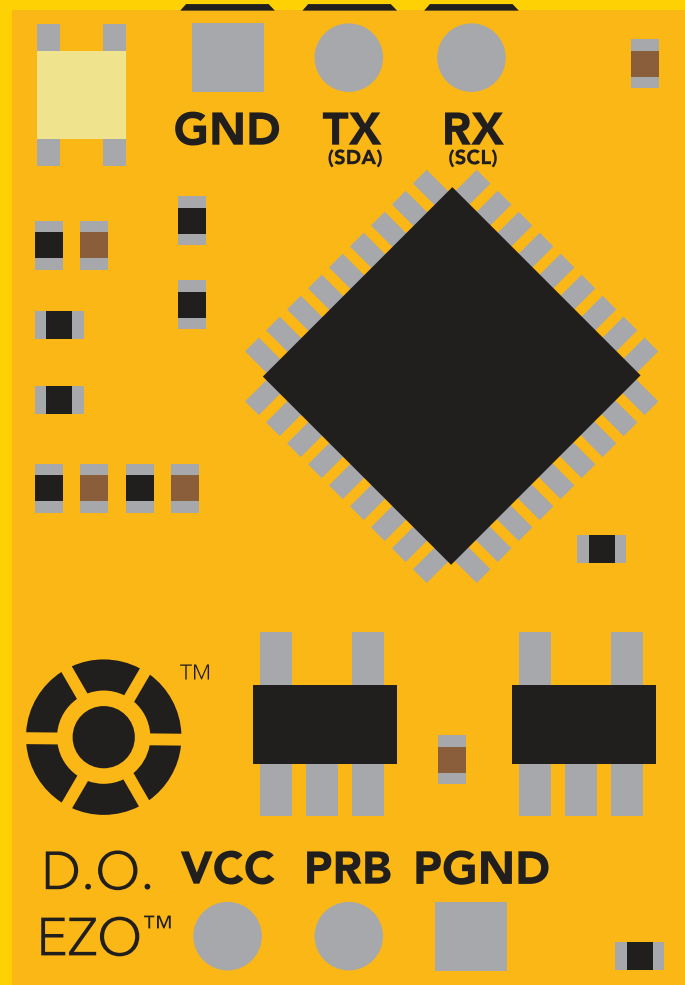
# EZO-DO<sup>TM</sup>

**Embedded Dissolved Oxygen Circuit**

**ISO 5814 Compliant**

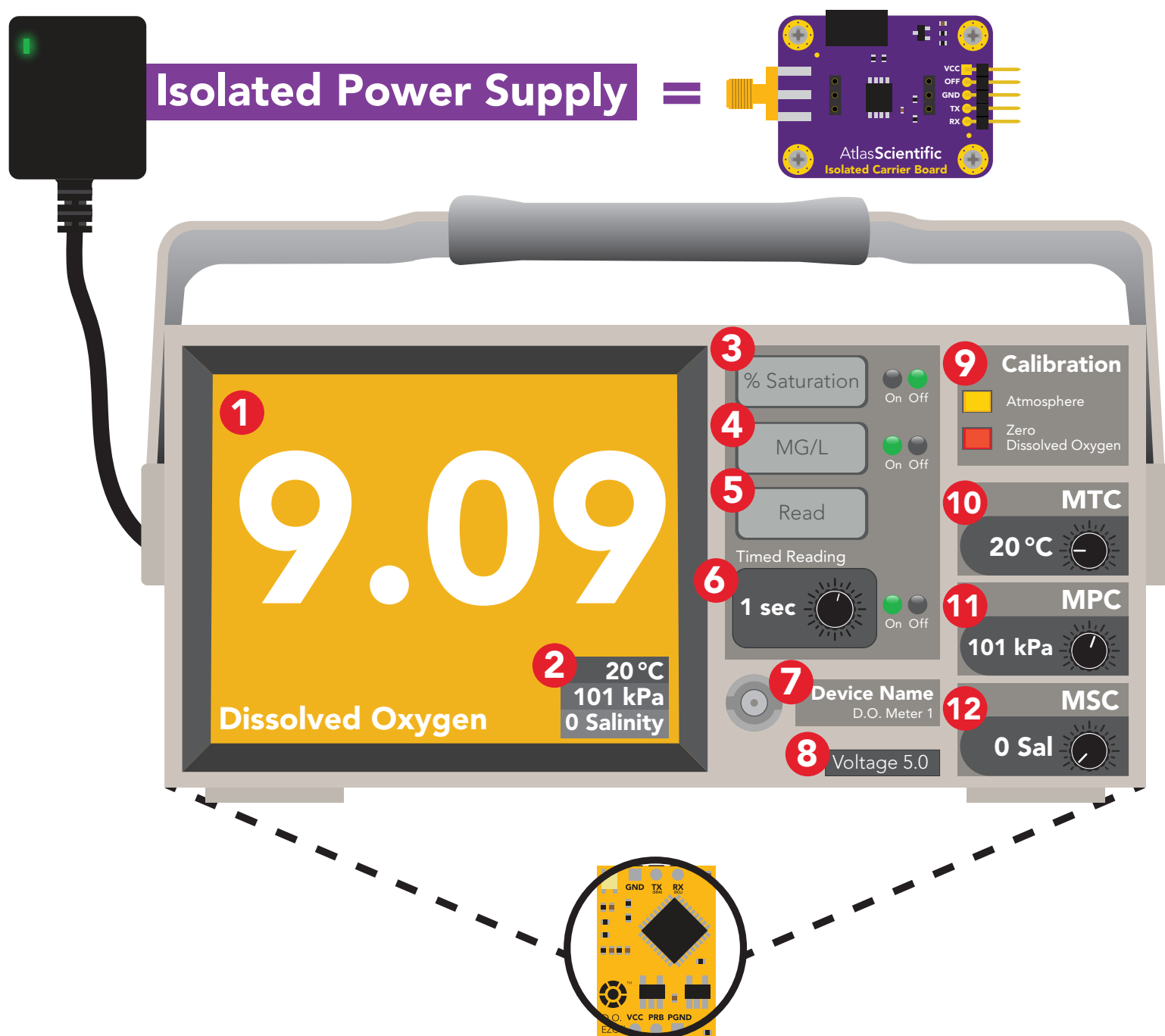
(determination of dissolved oxygen)

Reads	<b>Dissolved Oxygen</b>
Range	<b>0.00 – 100 mg/L 0 – 350% saturation</b>
Accuracy	<b>+/- 0.05 mg/L</b>
D.O. reading time	<b>600ms</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**

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



- 1** Two decimal D.O. reading
- 2** Temperature, pressure, and salinity compensation value
- 3** Percent saturation
- 4** Milligrams per liter
- 5** Immediate reading
- 6** Timed readings

- 7** Set device name
- 8** Voltage usage
- 9** Multi point calibration
- 10** Manual temperature compensation
- 11** Manual pressure compensation
- 12** Manual salinity compensation

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

# ✓ Available data protocols

## UART

Default

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

# ✗ Unavailable data protocols

## SPI

## Analog

## RS-485

## Mod Bus

## 4–20mA

# STOP



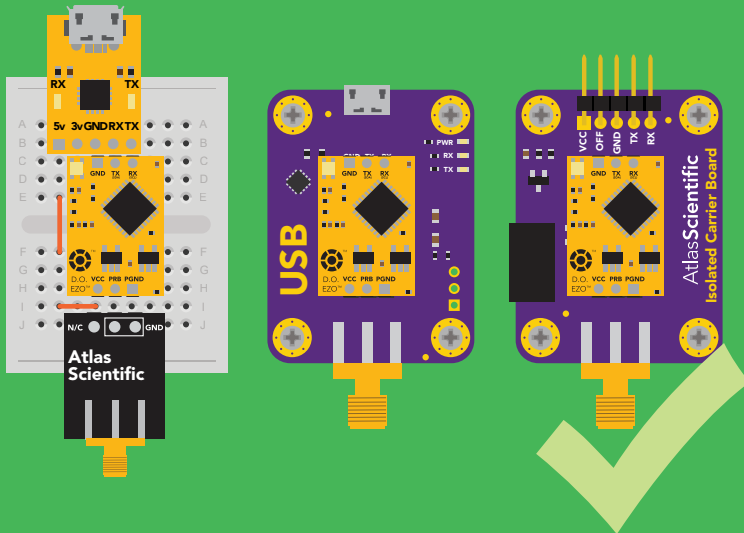
**SOLDERING THIS DEVICE VOIDS YOUR WARRANTY.**

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

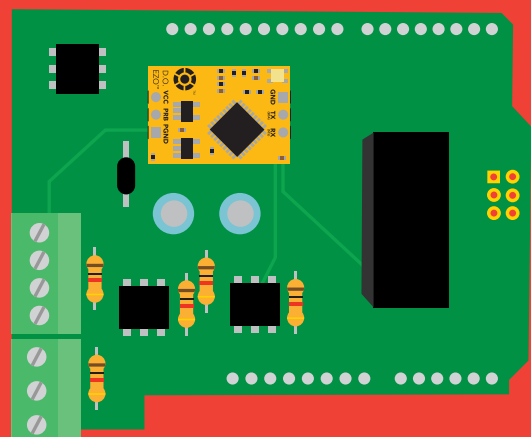
**Can you make a warranty claim after soldering? No.**

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

**Get this device working using one of these methods first.**



**Do not embed before you have experience with this sensor.**



# Table of contents

Available data protocols	3	Correct wiring	9
Circuit dimensions	6	Default state	10
Power consumption	6	Circuit footprint	73
Absolute max ratings	6	Datasheet change log	74
Electrical isolation	7	Warranty	79
Calibration theory	63		
Understanding D.O. measurements	68		
Hyper saturation with pure oxygen	70		

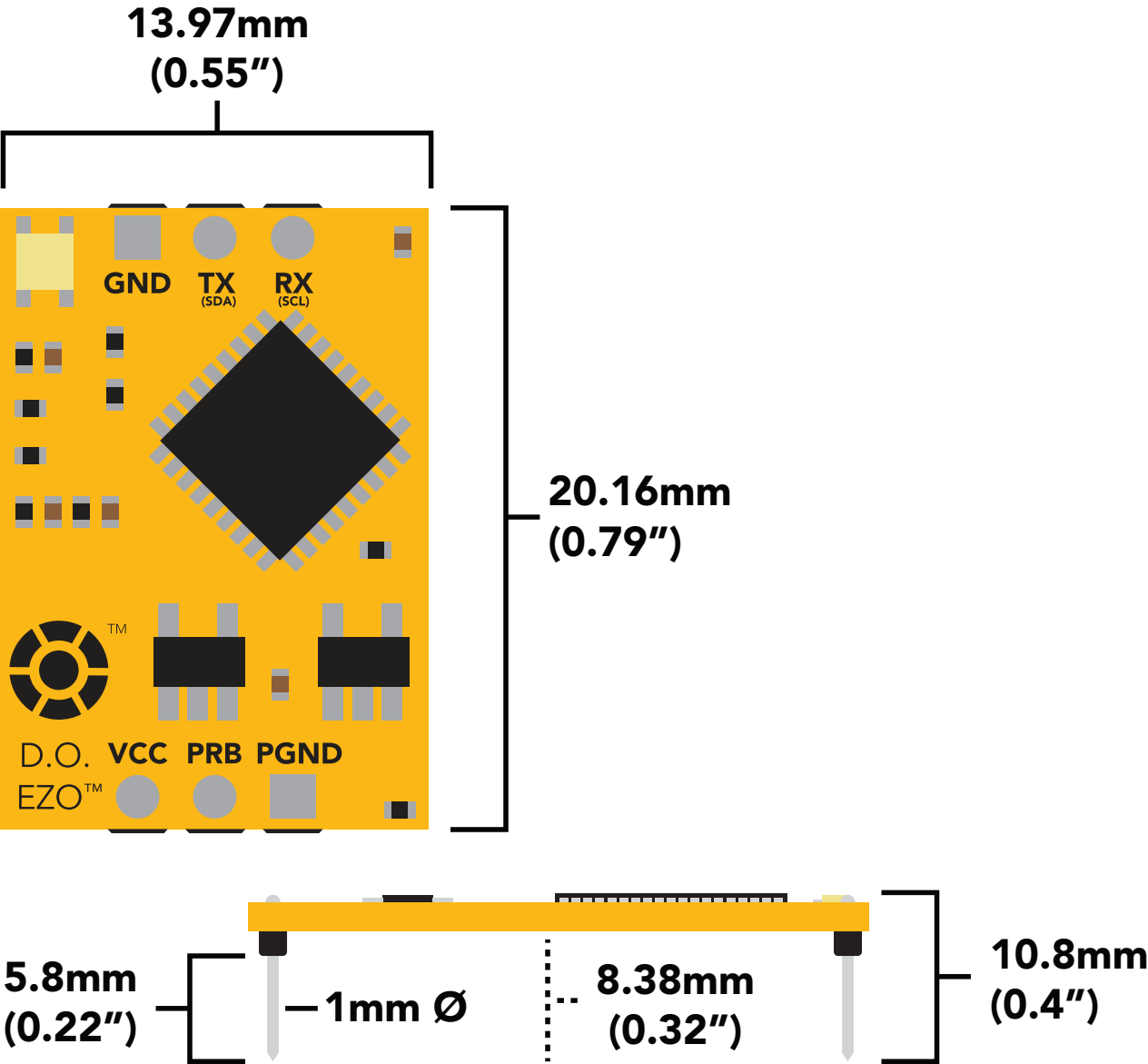
## UART

UART mode	11
LED color definition	12
Receiving data from device	13
Sending commands to device	14
<b>UART quick command page</b>	<b>15</b>
LED control	16
Find	17
Continuous reading mode	18
Single reading mode	19
Calibration	20
Export calibration	21
Import calibration	22
Temperature compensation	23
Salinity compensation	24
Atmospheric	
pressure compensation	25
Enable/disable parameters	26
Naming device	27
Device information	28
Response codes	29
Reading device status	30
Sleep mode/low power	31
Change baud rate	32
Protocol lock	33
Factory reset	34
Change to I <sup>2</sup> C mode	35
Manual switching to I <sup>2</sup> C	36

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

I <sup>2</sup> C mode	38
Sending commands	39
Requesting data	40
Response codes	41
LED color definition	42
<b>I<sup>2</sup>C quick command page</b>	<b>43</b>
LED control	44
Find	45
Taking reading	46
Calibration	47
Export calibration	48
Import calibration	49
Temperature compensation	50
Salinity compensation	51
Atmospheric	
pressure compensation	52
Enable/disable parameters	53
Naming device	54
Device information	55
Reading device status	56
Sleep mode/low power	57
Protocol lock	58
I <sup>2</sup> C address change	59
Factory reset	60
Change to UART mode	61
Manual switching to UART	62

# EZO™ circuit dimensions



## Power consumption

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

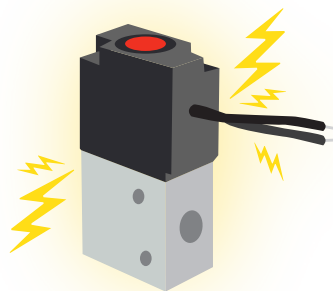
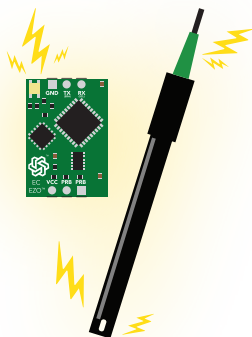
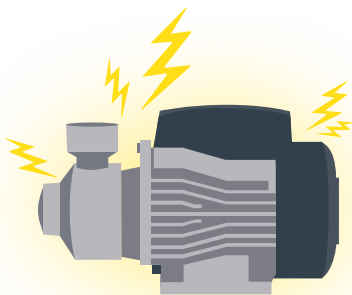
## Absolute max ratings

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

# Electrical isolation

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

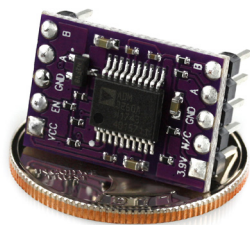
When electrical noise is interfering with the Dissolved Oxygen readings it is common to see rapidly fluctuating readings or readings that are consistently off. To verify that electrical noise is causing inaccurate readings, place the Dissolved Oxygen probe in a cup of water by itself. The readings should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



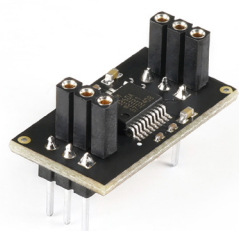
## Advice:

When reading D.O. along with other sensors, electrical isolation is strongly recommended. **Never build a commercial product without electrical isolation.**

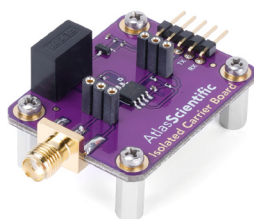
Atlas Scientific offers several different electrical isolation products that can be used in your design. Select the electrical isolation product that works best for your design.



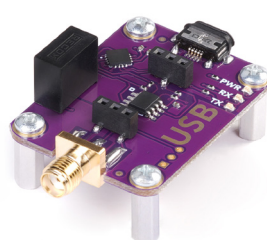
Basic EZO™  
Inline Voltage Isolator



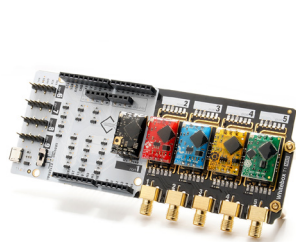
Vertical Isolator



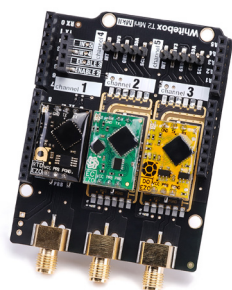
Electrically Isolated  
EZO™ Carrier Board



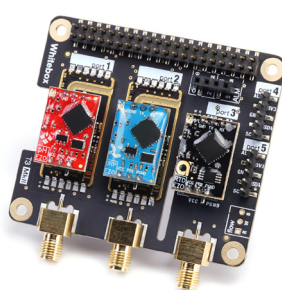
Gen 2 Electrically Isolated  
USB EZO™ Carrier Board



Whitebox T1



Whitebox T3



Whitebox T3



Electrically Isolated EZO™  
Carrier Board (old style)

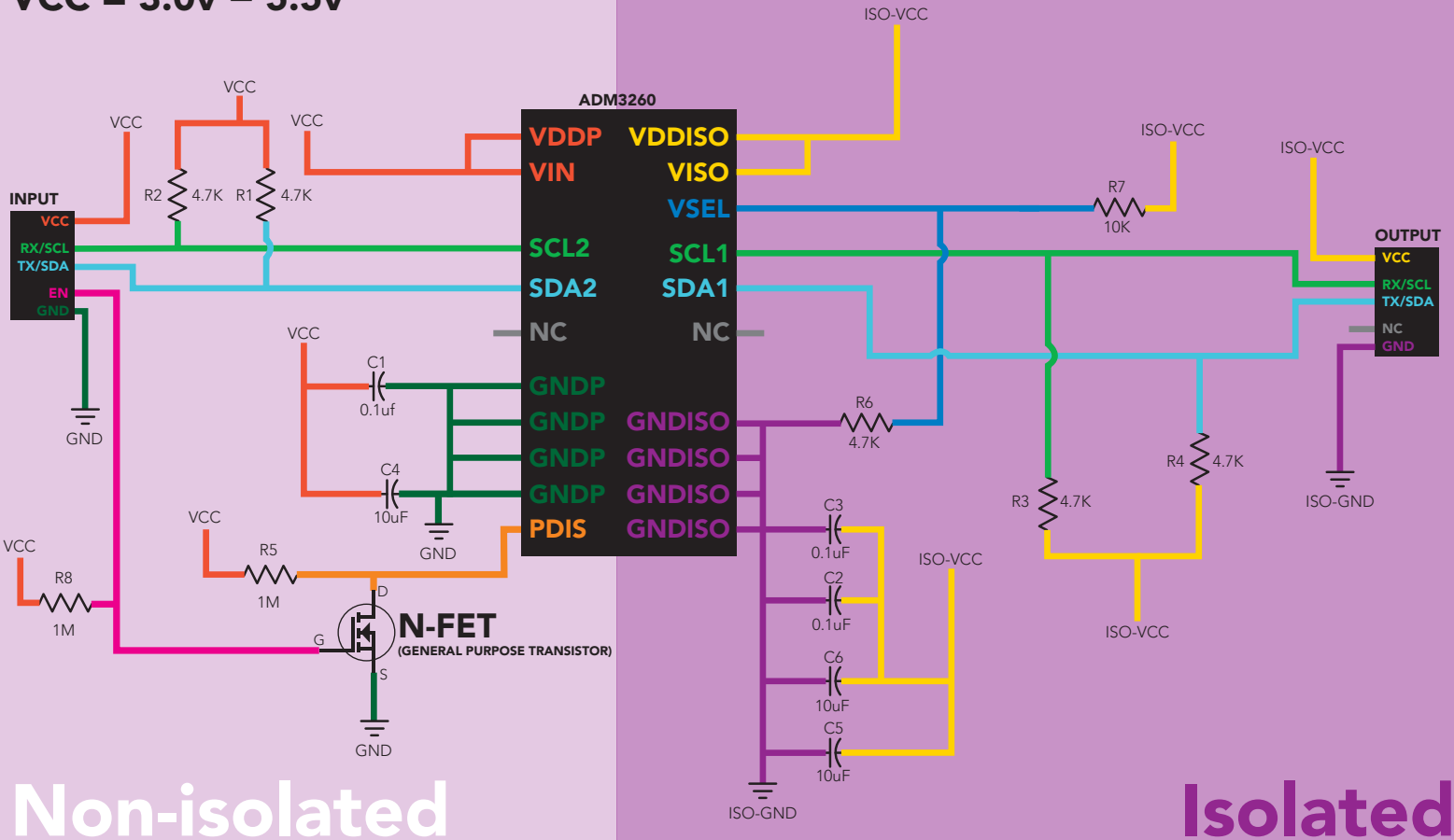
For various reasons, you may need to build your own electrical isolator. Because electrical isolation is so important, we have published our isolation schematic for anyone to use.

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

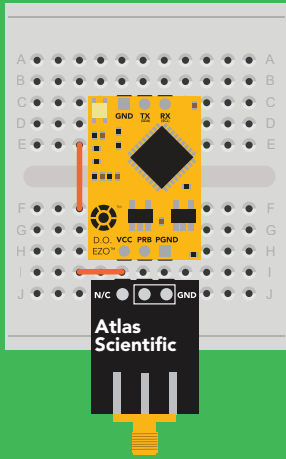
The two data channels have a 4.7kΩ pull-up resistor on both the isolated and non-isolated lines (R1, R2, R3, and R4). The output voltage is set using a voltage divider (R6 and R7). This produces a voltage of 3.9V regardless of your input voltage.

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

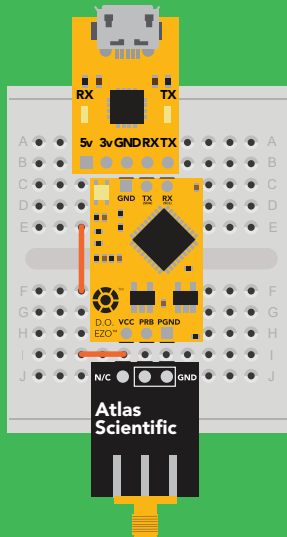
**VCC = 3.0v – 5.5v**



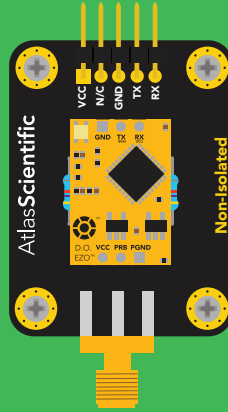
# ✓ Correct wiring



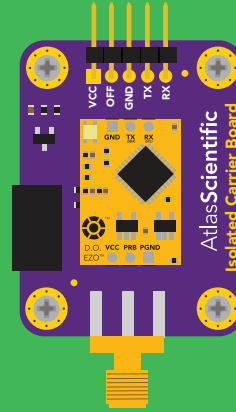
Bread board



Bread board via USB



Non-Isolated  
EZO™ Carrier Board



Electrically Isolated  
EZO™ Carrier Board



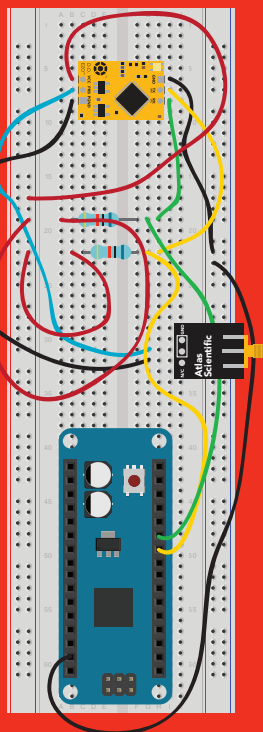
USB  
carrier board

# ✗ Incorrect wiring

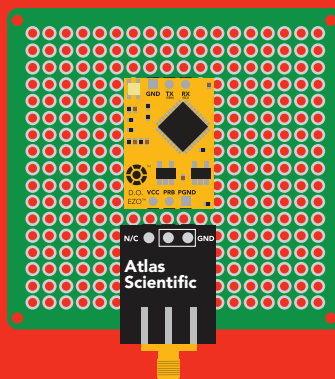
Extended leads



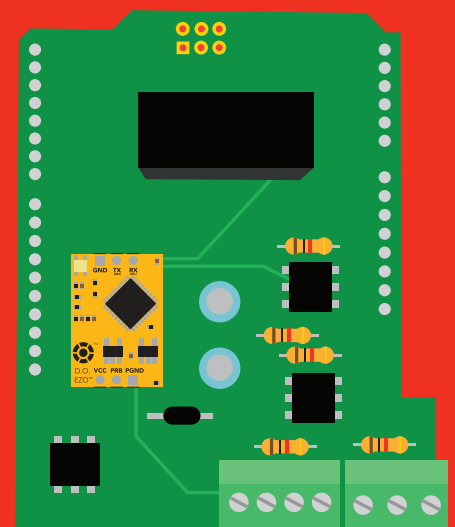
Sloppy setup



Perfboards or Protoboards



\*Embedded into your device



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

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

# Default state

## UART mode

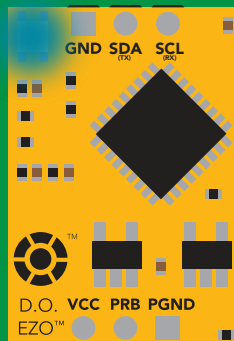
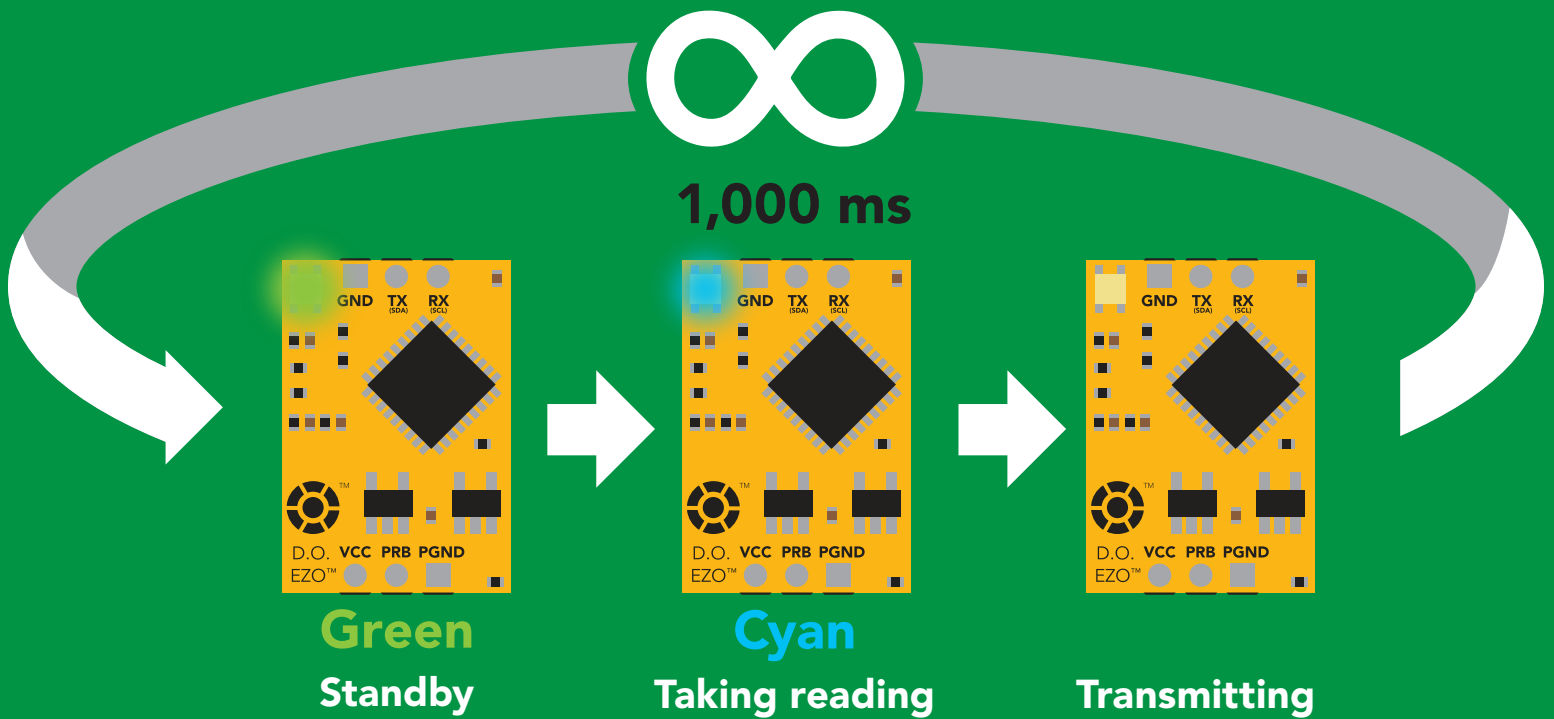
**Baud** 9,600

**Readings** continuous

**Units** mg/L

**Speed** 1 reading per second

**LED** on



**Solid Blue LED**  
in I<sup>2</sup>C mode  
Not UART ready

# UART mode

8 data bits      no parity  
1 stop bit      no flow control

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

**RX**  
Data in



**TX**  
Data out



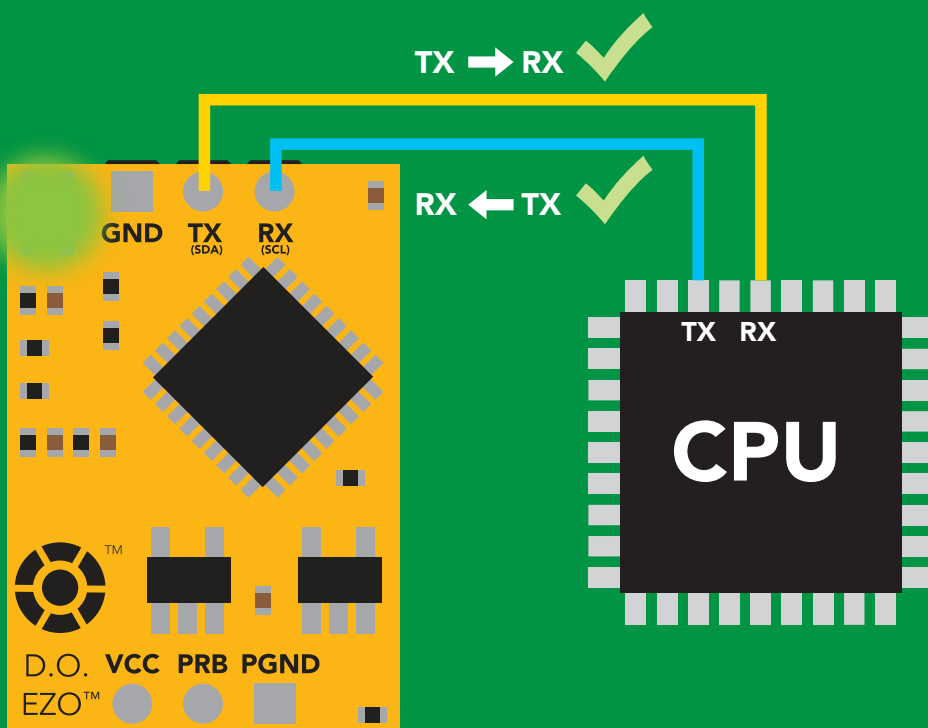
**Vcc** 3.3V – 5.5V

0V



Vcc

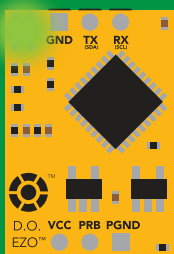
0V



## Data format

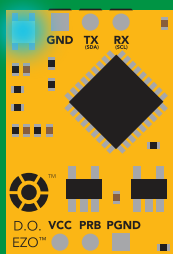
<b>Reading</b>	<b>D.O.</b>	<b>Data type</b>	<b>floating point</b>
<b>Order</b>	<b>mg/L &amp; (% sat)</b> <small>when enabled</small>	<b>Decimal places</b>	<b>mg/L = 2</b> <b>% sat = 1</b>
<b>Encoding</b>	<b>ASCII</b>	<b>Smallest string</b>	<b>4 characters</b>
<b>Format</b>	<b>string</b> <small>(CSV string when % sat is enabled)</small>	<b>Largest string</b>	<b>40 characters</b>
<b>Terminator</b>	<b>carriage return</b>		

# LED color definition



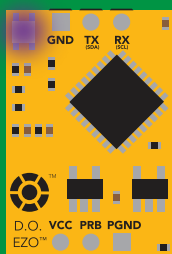
**Green**

UART standby



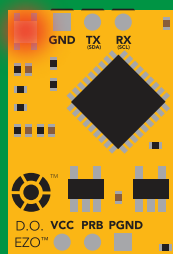
**Cyan**

Taking reading



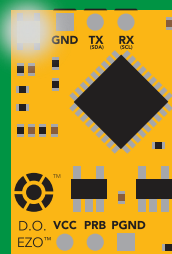
**Purple**

Changing  
baud rate



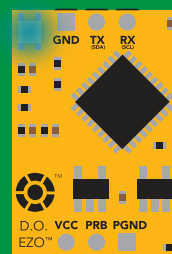
**Red**

Command  
not understood



**White**

Find



**Blue**

I2C standby

**5V**

LED ON  
**+0.4 mA**

**3.3V**

**+0.2 mA**

## Settings that are retained if power is cut

- Baud rate
- Calibration
- Continuous mode
- Device name
- Enable/disable parameters
- Enable/disable response codes
- Hardware switch to 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

# Receiving data from device

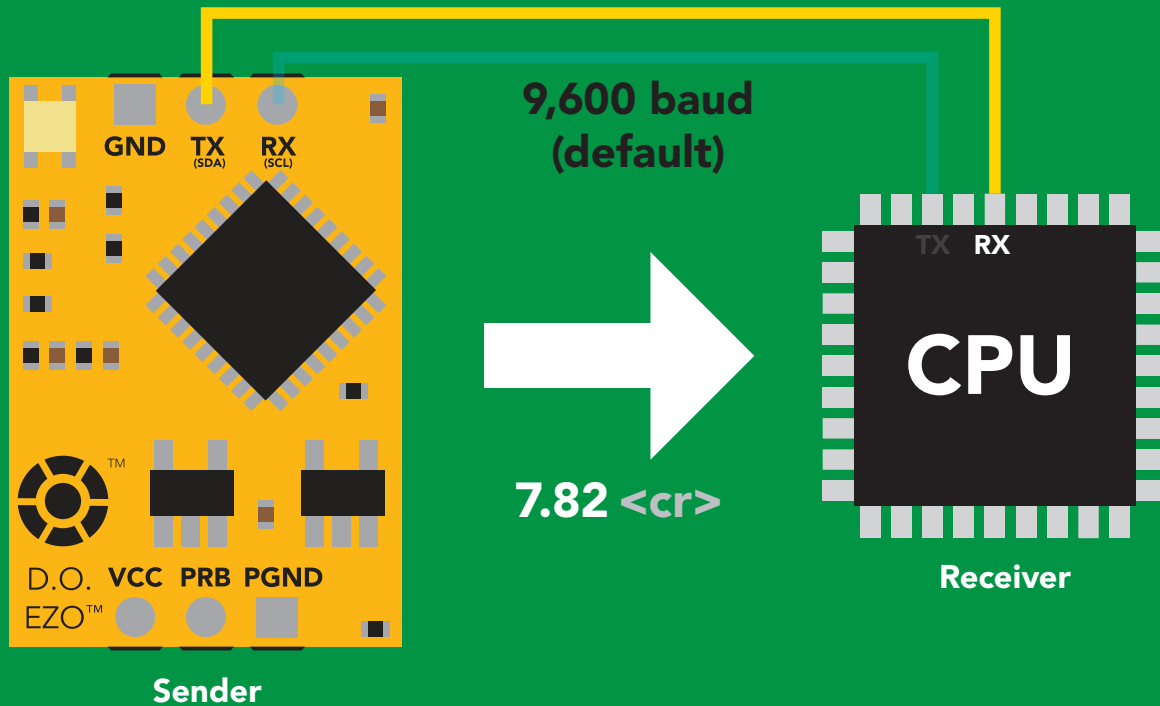
2 parts

ASCII data string

Command

Carriage return <cr>

Terminator



## Advanced

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

# Sending commands to device

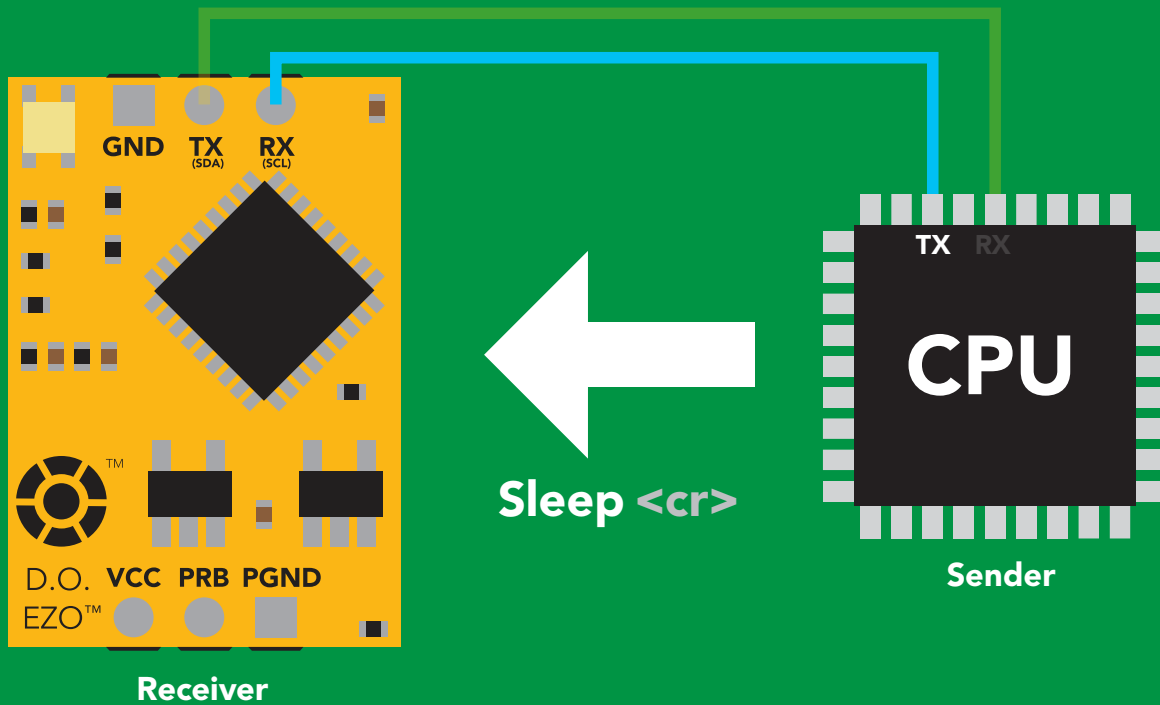
2 parts

**Command (not case sensitive)**

ASCII data string

**Carriage return <cr>**

Terminator



## Advanced

ASCII: **S** **I** **e** **e** **p** **<cr>**

Hex: **53** **6C** **65** **65** **70** **0D**

Dec: **83** **108** **101** **101** **112** **13**

# UART mode

## command quick reference

All commands are ASCII strings or single ASCII characters.

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

# LED control

## Command syntax

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

L,0 <cr> LED off

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

## Example

## Response

L,1 <cr>

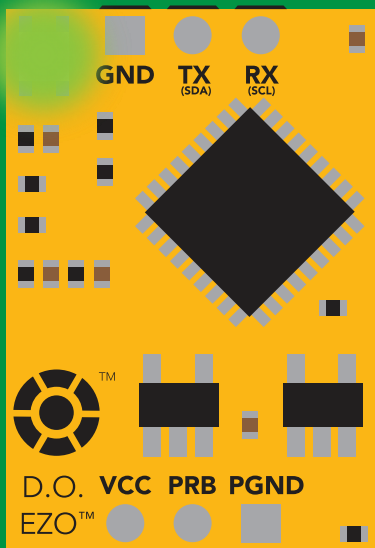
\*OK <cr>

L,0 <cr>

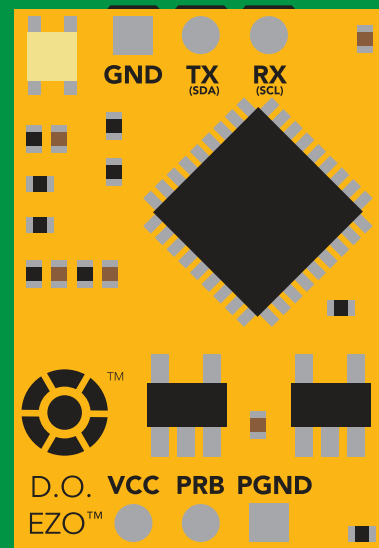
\*OK <cr>

L,? <cr>

?L,1 <cr> **or** ?L,0 <cr>  
\*OK <cr>



L,1



L,0

# Find

## Command syntax

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

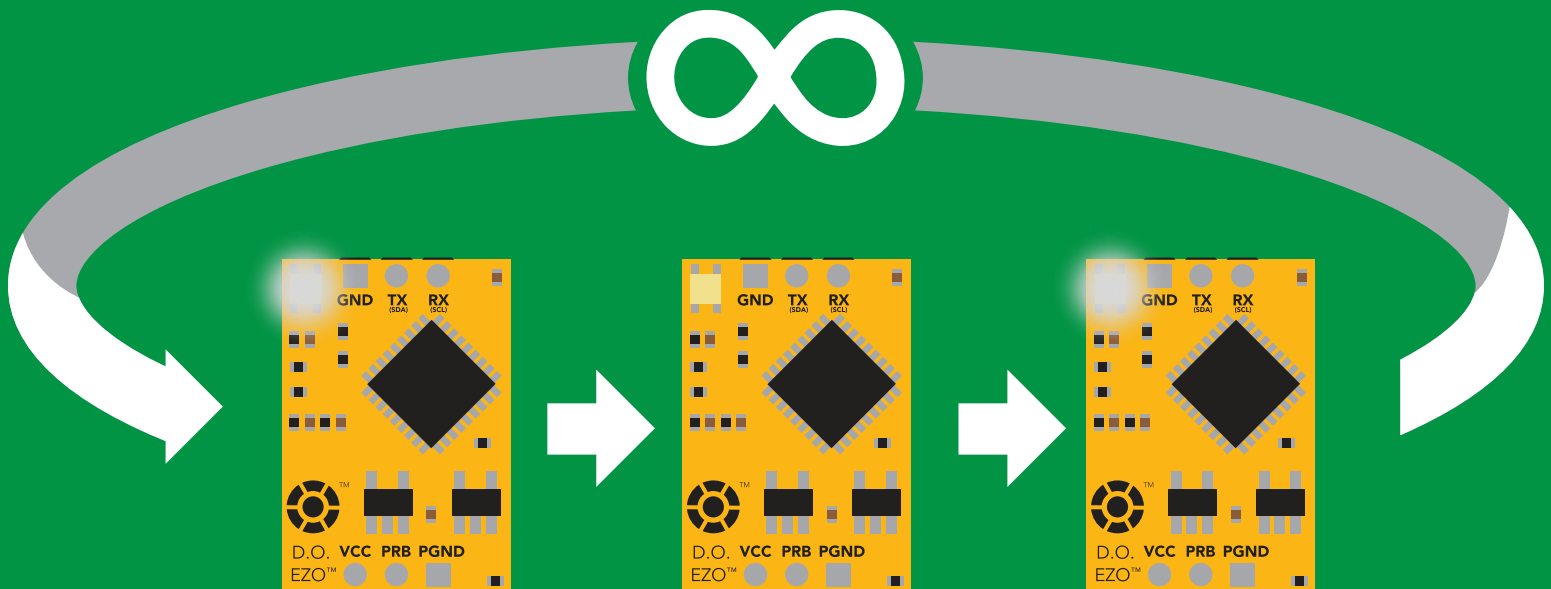
**Find** <cr> LED rapidly blinks white, used to help find device

## Example

**Find** <cr>

## Response

**\*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

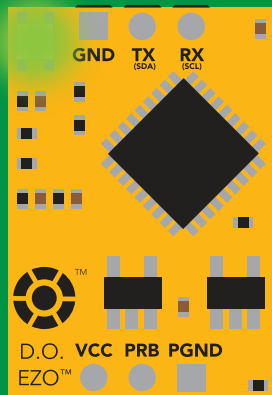
### Example

**R** <cr>

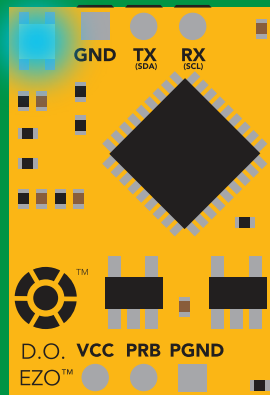
### Response

**7.82** <cr>

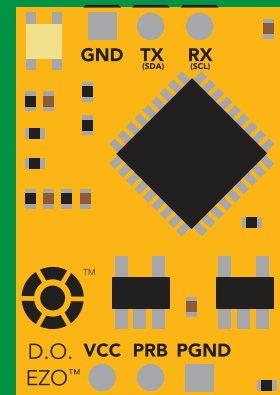
**\*OK** <cr>



**Green**  
Standby



**Cyan**  
Taking reading



**Transmitting**



**600 ms**

# 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>

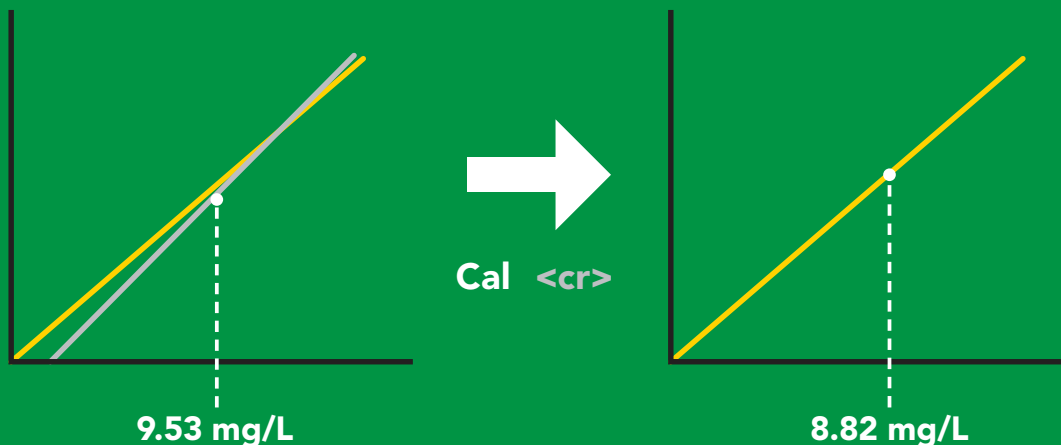
**\*OK** <cr>

**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>

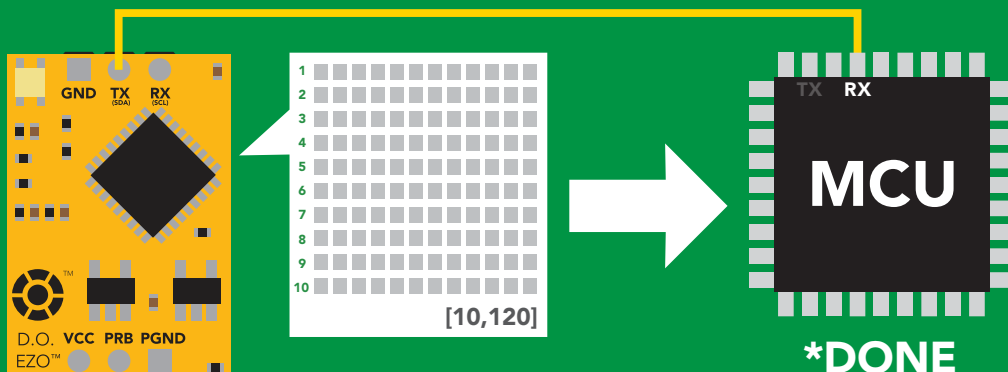
**6F 6C 20 67 75 79** <cr> **(10 of 10)**

**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

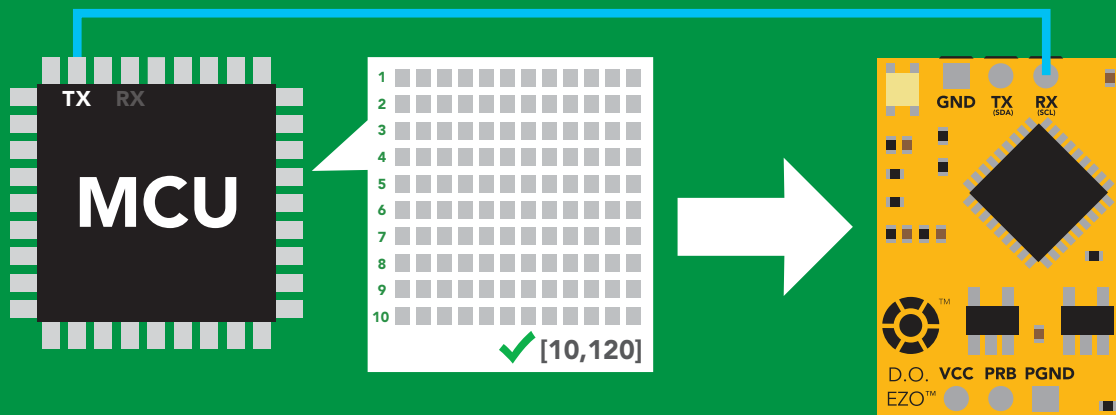
\*OK <cr>

\*OK <cr>

⋮

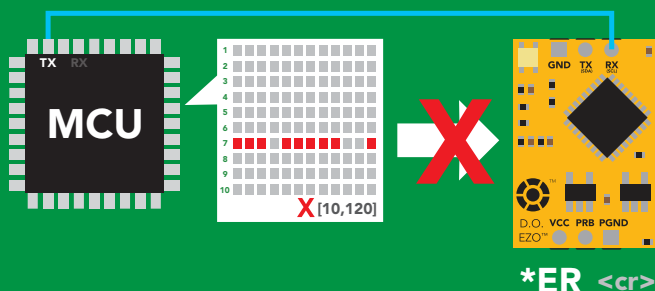
\*OK <cr>

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

## Response

**T,19.5** <cr>

**\*OK** <cr>

**RT,19.5** <cr>

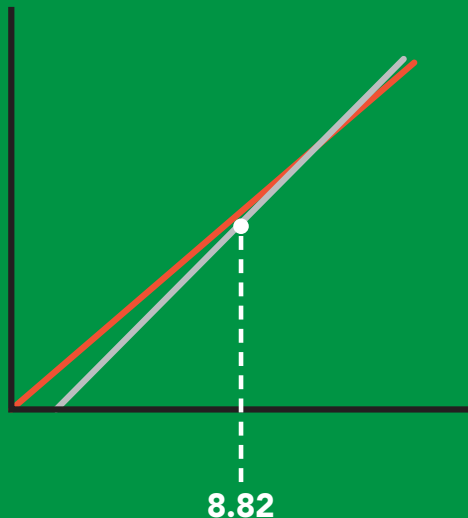
**\*OK** <cr>

**8.91** <cr>

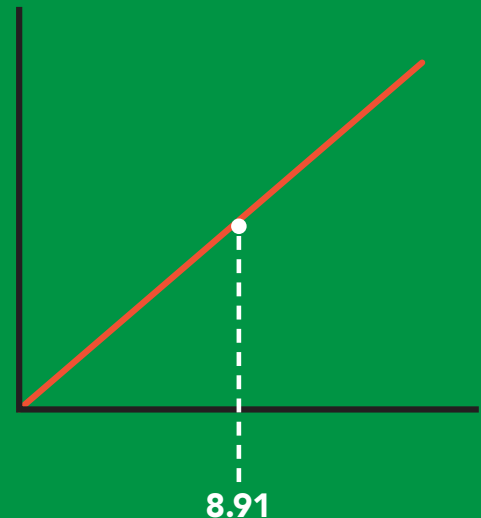
**T,?** <cr>

**?T,19.5** <cr>

**\*OK** <cr>



→  
**T,19.5** <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

## Response

**S,50000** <cr>

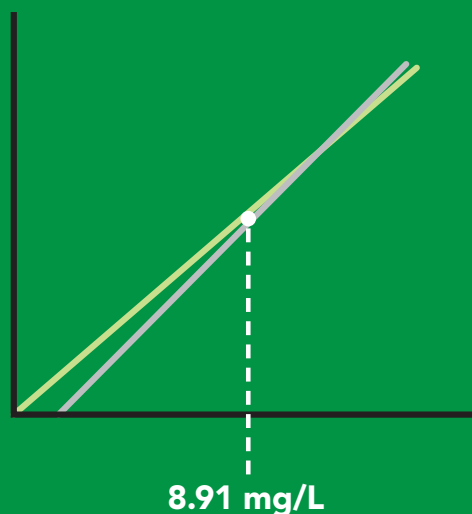
**\*OK** <cr>

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

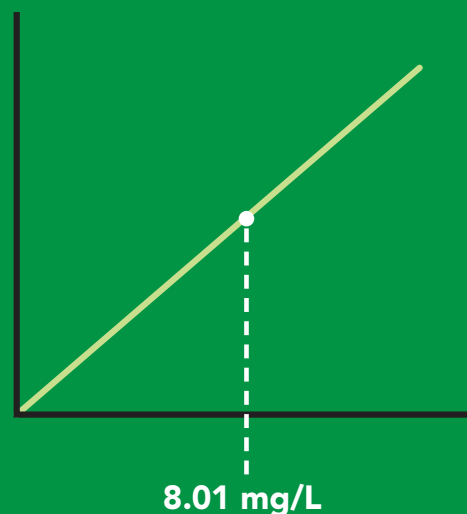
**\*OK** <cr>

**S,?** <cr>

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



**S,50000** <cr>



# Atmospheric pressure compensation

## Command syntax

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

**P,?** <cr> compensated pressure value?

### Example

### Response

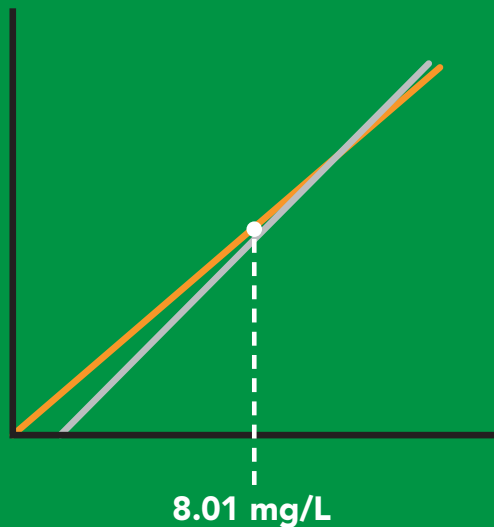
**P,90.25** <cr>

**\*OK** <cr>

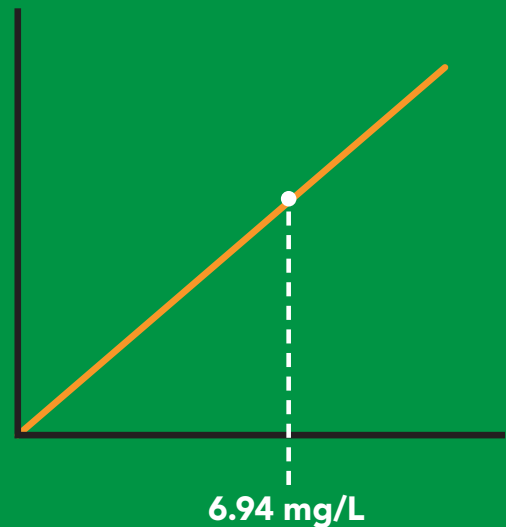
**P,?** <cr>

**?,P,90.25** <cr>

**\*OK** <cr>



→  
**P,90.25** <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>

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

O,? <cr>

## Response

\*OK <cr> enable / disable mg/L

\*OK <cr> enable / disable percent saturation

?,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

Do not use spaces in the name

Name,n <cr> set name

Name, <cr> clears name

Name,? <cr> show name

n =

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

Up to 16 ASCII characters

## Example

## Response

Name, <cr>

\*OK <cr> name has been cleared

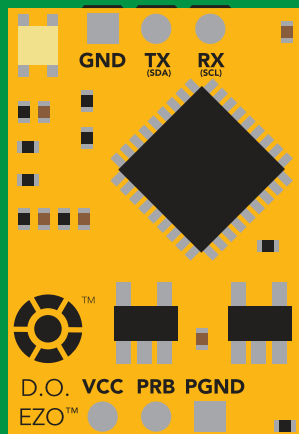
Name,zzt <cr>

\*OK <cr>

Name,? <cr>

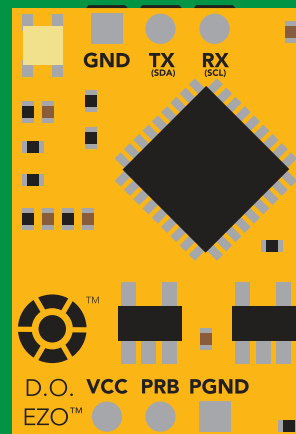
?Name,zzt <cr>  
\*OK <cr>

Name,zzt



\*OK <cr>

Name,?



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

# Device information

## Command syntax

```
i <cr> device information
```

### Example

```
i <cr>
```

### Response

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

## Response breakdown

?i,	D.O.,	1.98
	↑	↑
	Device	Firmware

# Response codes

## Command syntax

**\*OK,1** <cr> enable response **default**

**\*OK,0** <cr> disable response

**\*OK,?** <cr> response on/off?

## Example

## Response

**R** <cr>

**7.82** <cr>  
**\*OK** <cr>

**\*OK,0** <cr>

no response, **\*OK** disabled

**R** <cr>

**7.82** <cr> **\*OK** disabled

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

**?\*OK,1** <cr> or **?\*OK,0** <cr>

## Other response codes

**\*ER** unknown command  
**\*OV** over volt ( $VCC \geq 5.5V$ )  
**\*UV** under volt ( $VCC \leq 3.1V$ )  
**\*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

Status <cr>

### Response

?Status,P,5.038 <cr>  
\*OK <cr>

## Response breakdown

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

### Restart codes

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

# Sleep mode/low power

## Command syntax

Send any character or command to awaken device.

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

## Example

## Response

**Sleep** <cr>

**\*OK** <cr>

**\*SL** <cr>

**Any command**

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

**5V**

STANDBY

**13.1 mA**

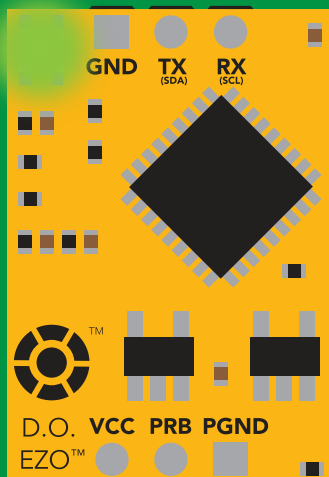
SLEEP

**0.66 mA**

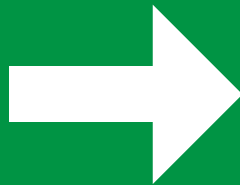
**3.3V**

12 mA

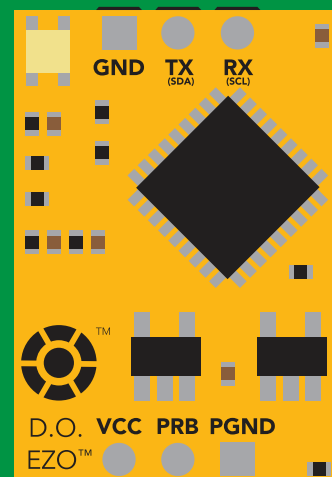
0.3 mA



**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>

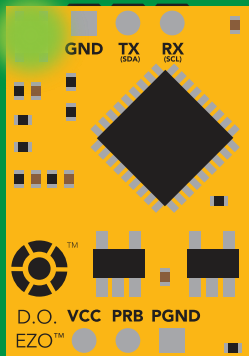
Baud,? <cr>

?Baud,38400 <cr>

\*OK <cr>

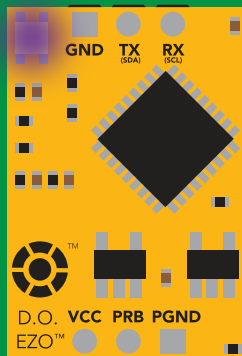
n =

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



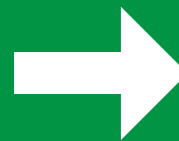
Standby

Baud,38400 <cr>

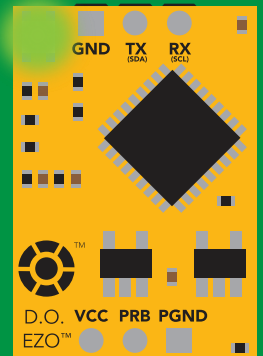


Changing  
baud rate

\*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

## Response

Plock,1 <cr>

\*OK <cr>

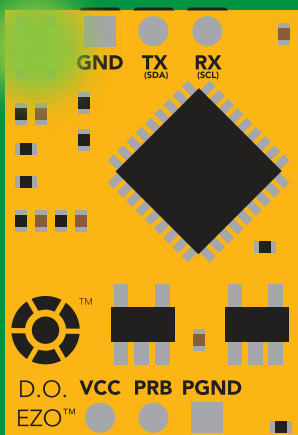
Plock,0 <cr>

\*OK <cr>

Plock,? <cr>

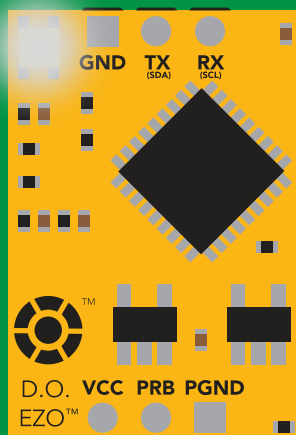
?Plock,1 <cr> **or** ?Plock,0 <cr>

Plock,1



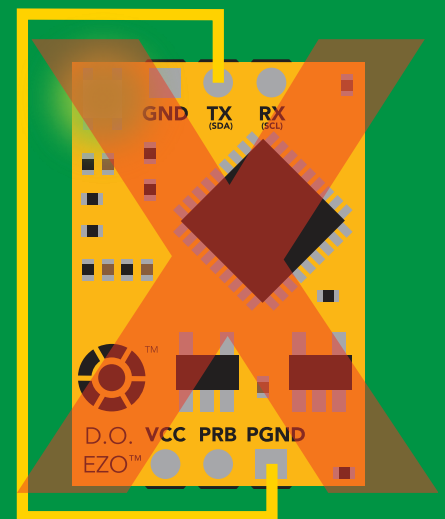
\*OK <cr>

I2C,100



cannot change to 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

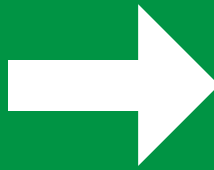
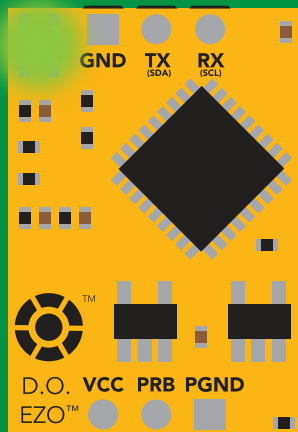
### Example

### Response

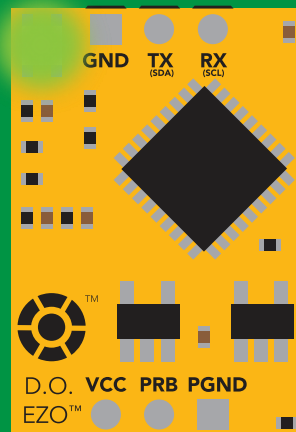
Factory <cr>

\*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)

I2C,n <cr> sets I<sup>2</sup>C address and reboots into I<sup>2</sup>C mode

n = any number 1 – 127

### Example

I2C,100 <cr>

### Response

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

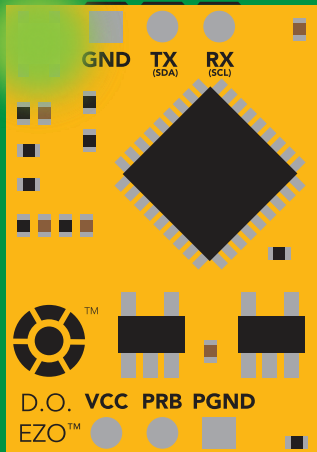
### Wrong example

I2C,139 <cr> n ≠ 127

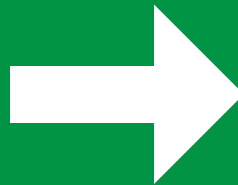
### Response

\*ER <cr>

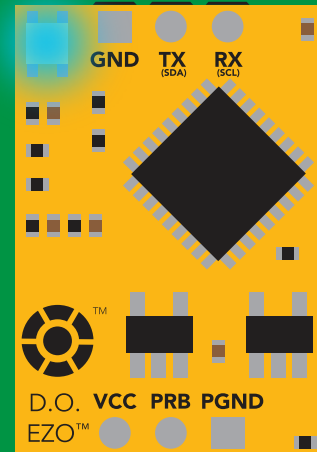
I2C,100



Green  
\*OK <cr>



(reboot)



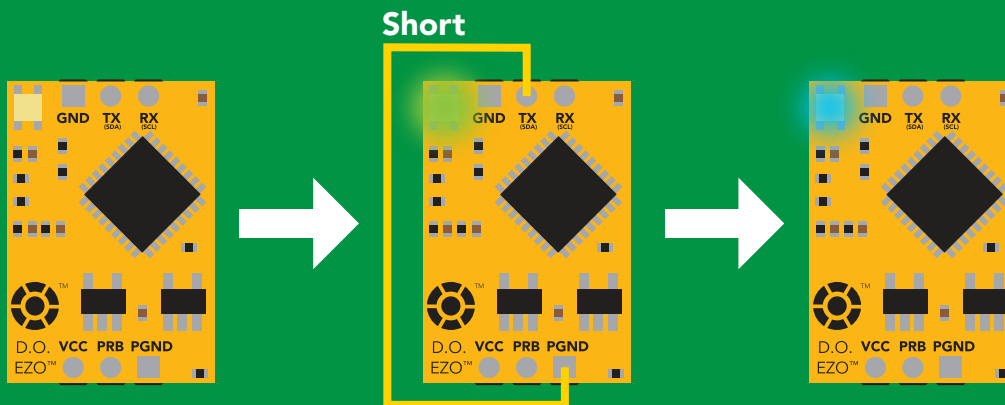
Blue  
now in I<sup>2</sup>C mode

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

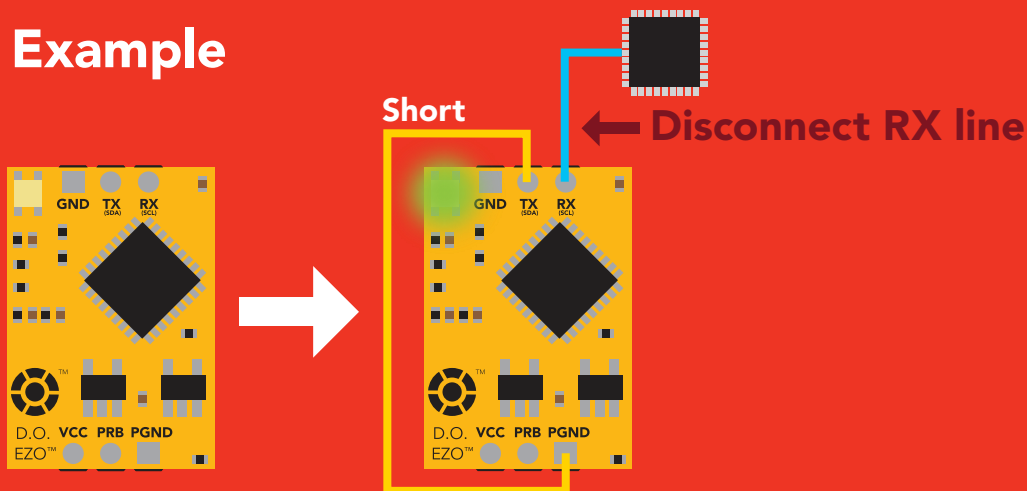
- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to PGND
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from **Green** to **Blue**
- Disconnect ground (power off)
- Reconnect all data and power

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**

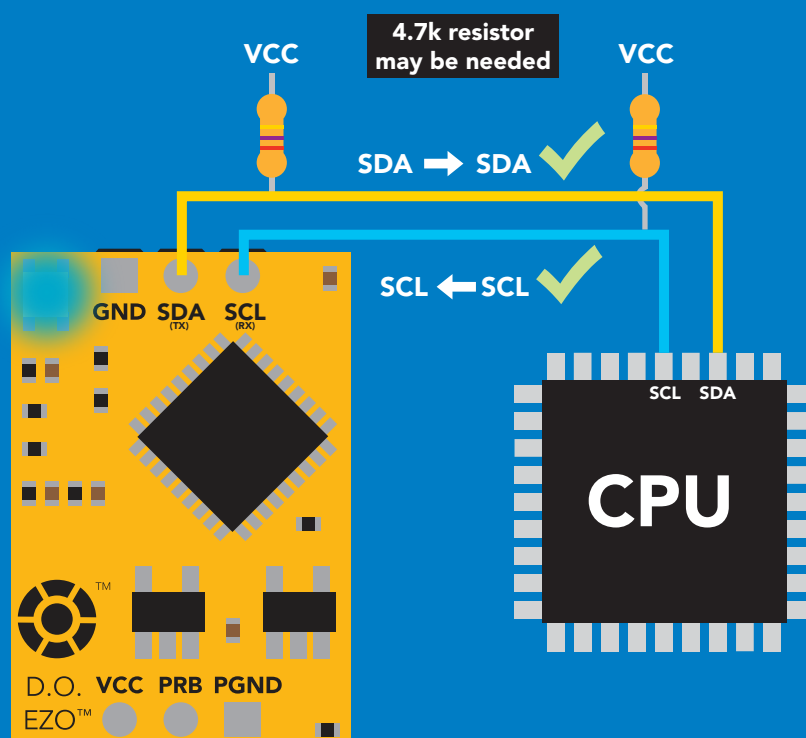
**V<sub>cc</sub>** 3.3V – 5.5V

**Clock speed** 100 – 400 kHz

**SDA** 

**SCL** 





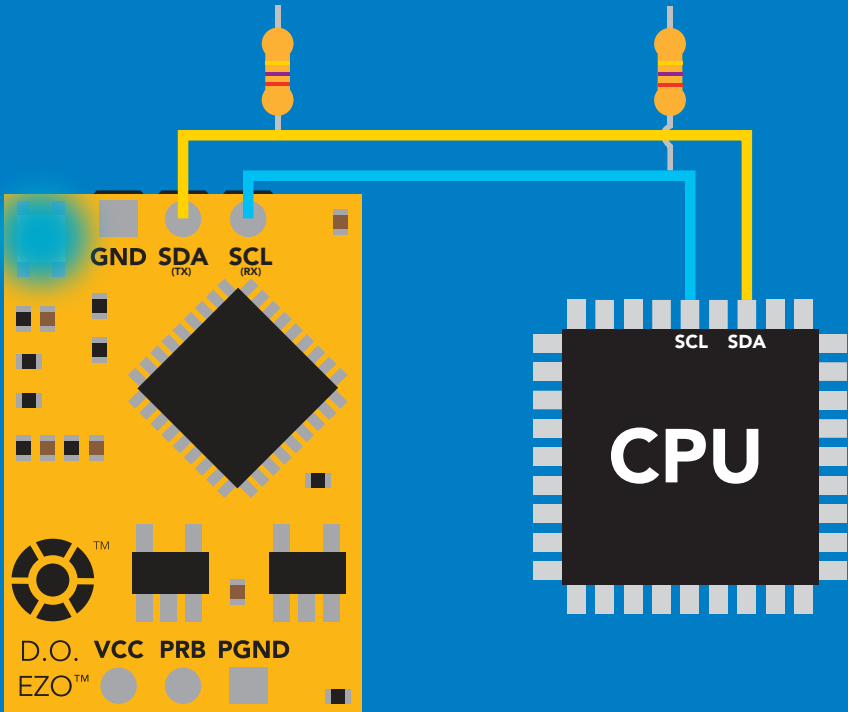
## Data format

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

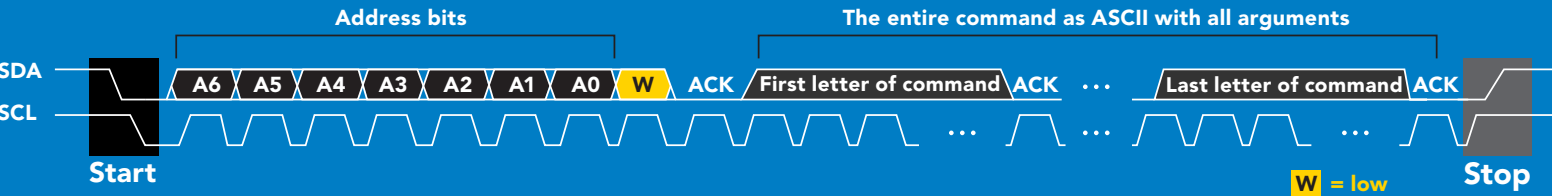
# Sending commands to device



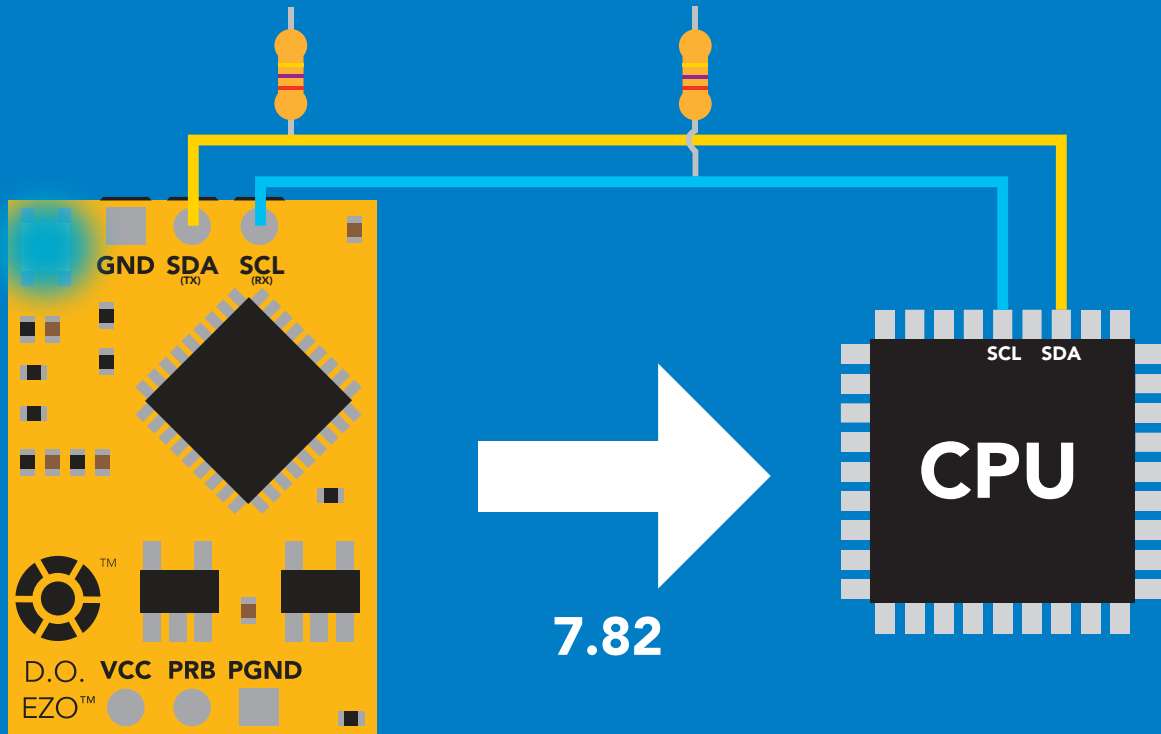
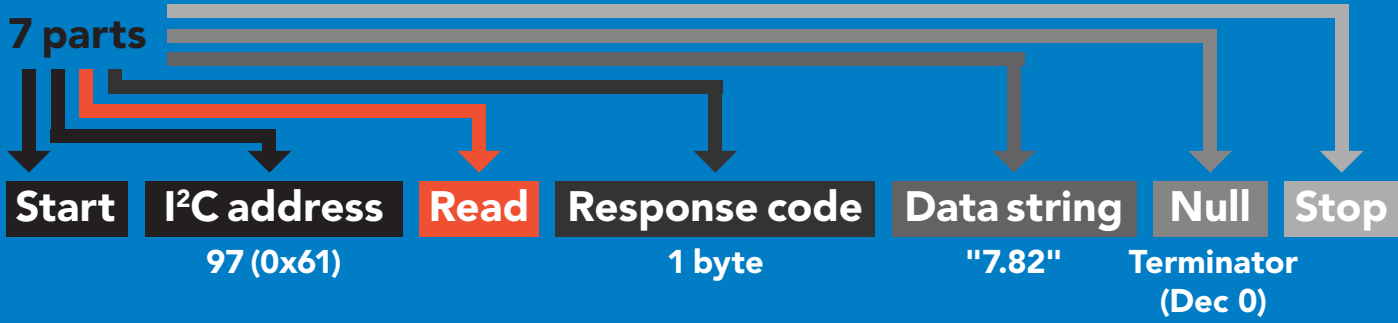
## Example



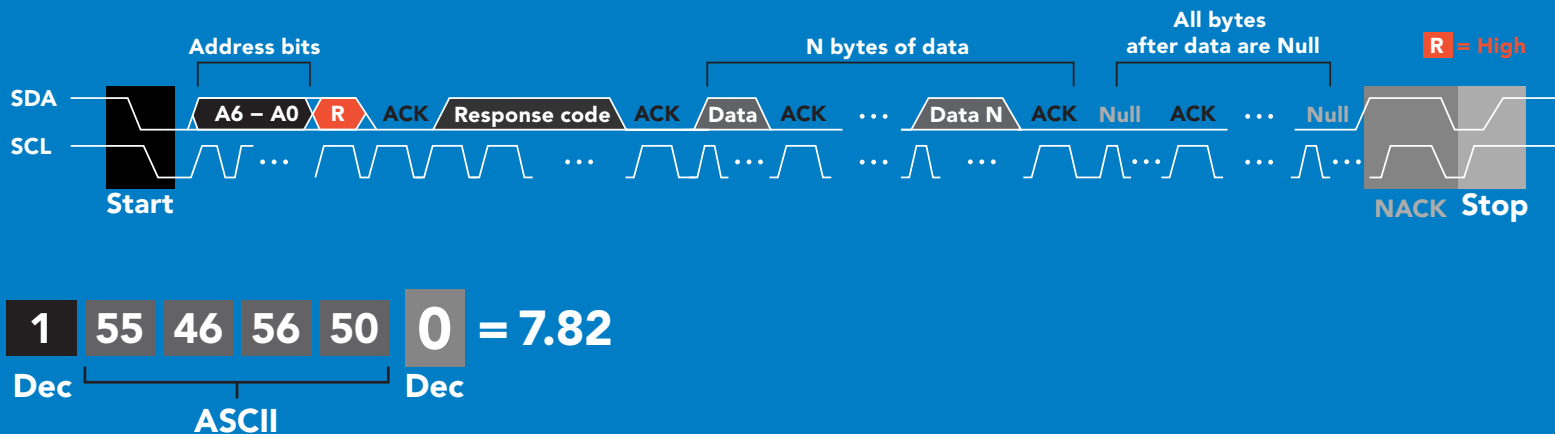
## Advanced



# Requesting data from device



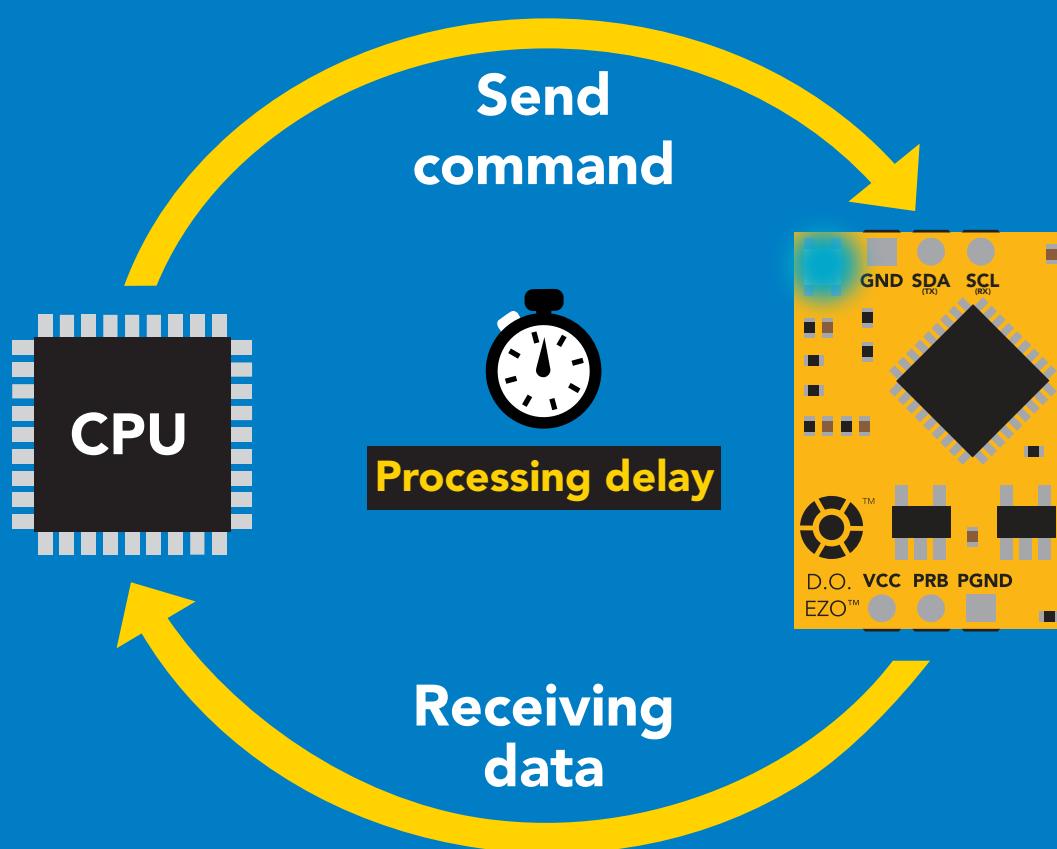
# Advanced



# Response codes

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

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



## Example

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

**delay(300);**



**Processing delay**

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

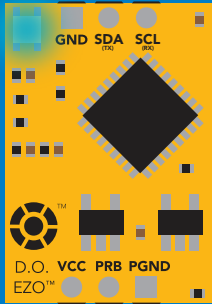
The response code will always be 254, if you do not wait for the processing delay.

### Response codes

Single byte, not string

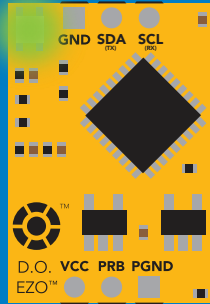
<b>255</b>	<b>no data to send</b>
<b>254</b>	<b>still processing, not ready</b>
<b>2</b>	<b>syntax error</b>
<b>1</b>	<b>successful request</b>

# LED color definition



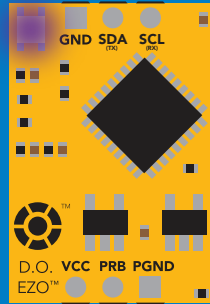
**Blue**

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



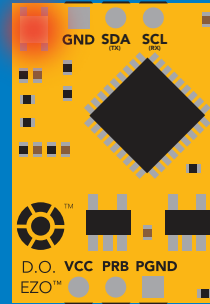
**Green**

**Taking reading**



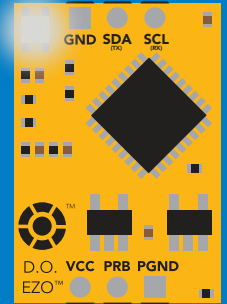
**Purple**

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



**Red**

**Command  
not understood**



**White**

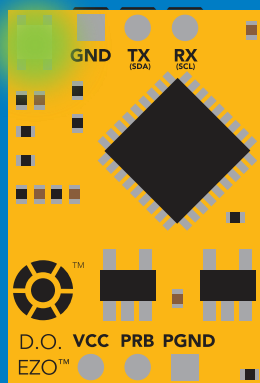
**Find**

**5V**

LED ON  
**+0.4 mA**

**3.3V**

**+0.2 mA**



**Solid Green LED**

**in UART mode  
Not I<sup>2</sup>C ready**

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

# LED control

## Command syntax

300ms  processing delay

L,1 LED on **default**

L,0 LED off

L,? LED state on/off?

## Example

## Response

L,1

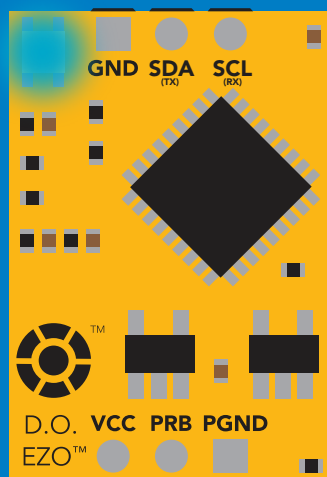
 **Wait 300ms** **1** **0**  
Dec Null

L,0

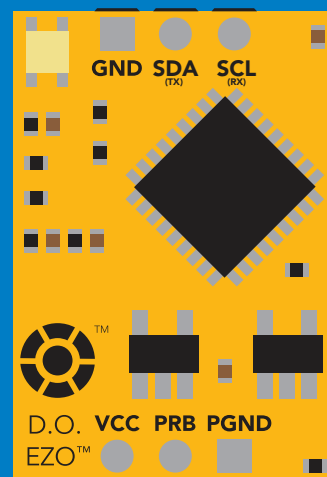
 **Wait 300ms** **1** **0**  
Dec Null

L,?

 **Wait 300ms** **1** **?L,1** **0** or **1** **?L,0** **0**  
Dec ASCII Null 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.

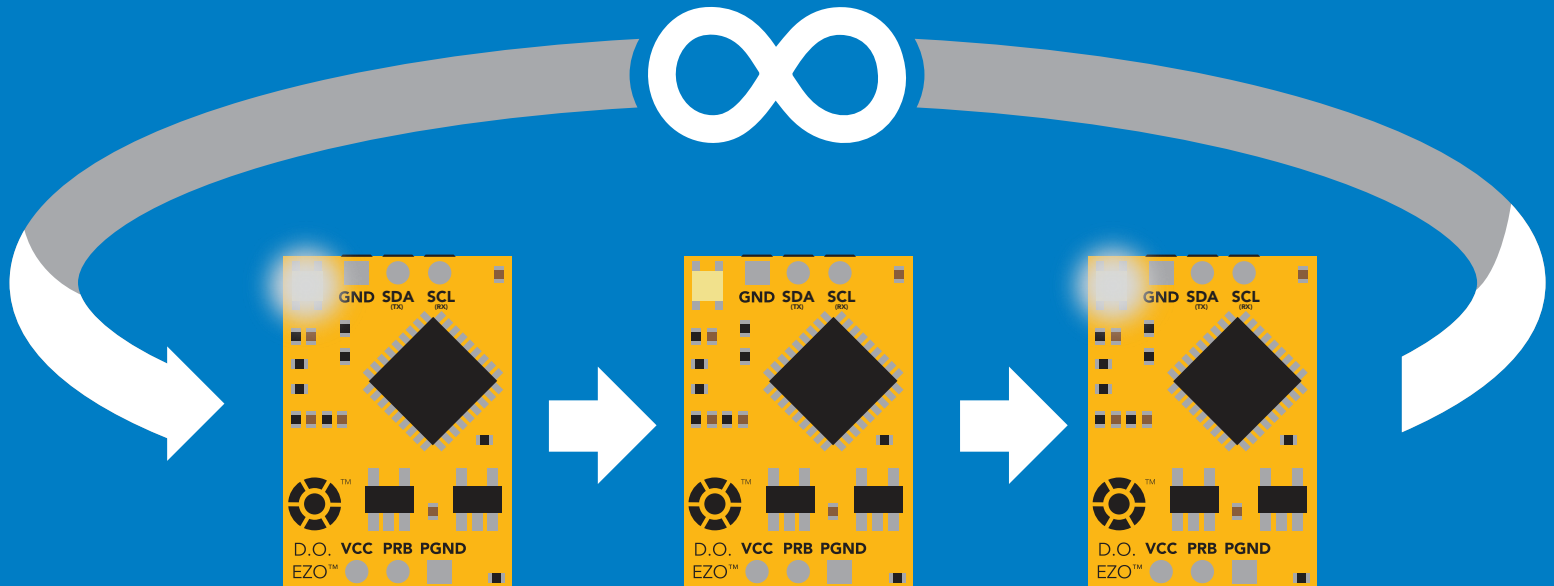
**Find** LED rapidly blinks white, used to help find device

## Example

## Response

**Find**

 **Wait 300ms**  
**1** **0**  
Dec Null



# Taking reading

## Command syntax

600ms  processing delay

R return 1 reading

## Example

R

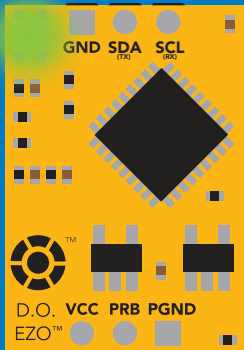
## Response



1  
Dec

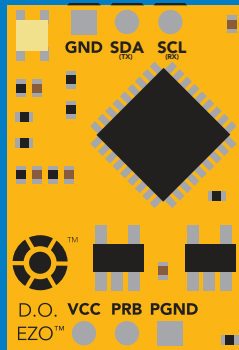
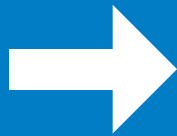
7.82  
ASCII

0  
Null

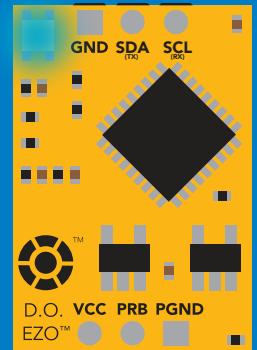
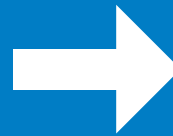


Green

Taking reading



Transmitting



Blue

Standby

# 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	0
Dec	Null

Cal,0

 Wait 1300ms 


1	0
Dec	Null

Cal,clear

 Wait 300ms 

1	0
Dec	Null

Cal,?

 Wait 300ms 

1	?Cal,0	0
Dec	ASCII	Null

 or 

1	?Cal,1	0
Dec	ASCII	Null

  
or 

1	?Cal,2	0
Dec	ASCII	Null

# Export calibration

300ms  processing delay

## Command syntax

Export: Use this command to download calibration settings

Export,? calibration string info

Export export calibration string from calibrated device

## Example

## Response

Export,?

 **Wait 300ms**    **1**    **10,120**    **0**  
Dec    ASCII    Null

### Response breakdown

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

Export strings can be up to 12 characters long

Export

 **Wait 300ms**    **1**    **59 6F 75 20 61 72**    **0**    **(1 of 10)**  
Dec    ASCII    Null

Export

 **Wait 300ms**    **1**    **65 20 61 20 63 6F**    **0**    **(2 of 10)**  
Dec    ASCII    Null

(7 more)

⋮

Export

 **Wait 300ms**    **1**    **6F 6C 20 67 75 79**    **0**    **(10 of 10)**  
Dec    ASCII    Null

Export

 **Wait 300ms**    **1**    **\*DONE**    **0**  
Dec    ASCII    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

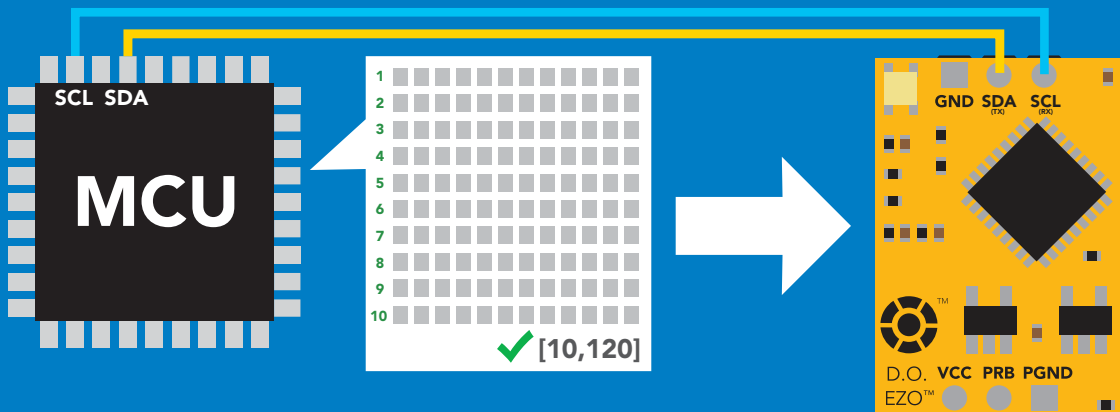
 **1** **0**  
Wait 300ms    Dec    Null

 **1** **0**  
Wait 300ms    Dec    Null

⋮

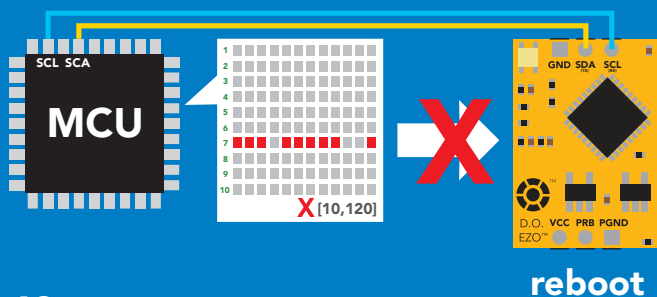
 **1** **0**  
Wait 300ms    Dec    Null

Import,n



**1** **\*Pending** **0**  
Dec    ASCII    Null

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

## Response

**T,19.5**

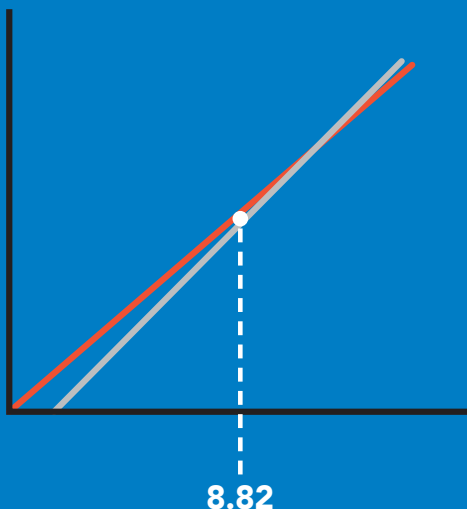
 **Wait 300ms**    **1**    **0**  
Dec    Null

**RT,19.5**

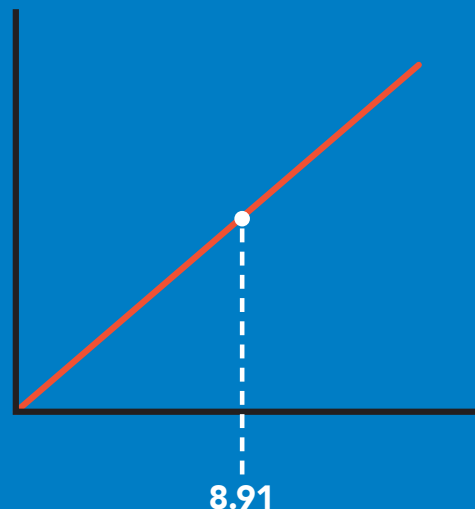
 **Wait 900ms**    **1**    **8.91**    **0**  
Dec    ASCII    Null

**T,?**

 **Wait 300ms**    **1**    **?T,19.5**    **0**  
Dec    ASCII    Null



→  
**T,19.5**



# Salinity compensation

## Command syntax

300ms  processing delay

S,n      n = any value in microsiemens

**default**

S,n,ppt      n = any value in ppt

S,?      compensated salinity value?

## Example

## Response

S,50000



Wait 300ms

1

Dec

0

Null

S,37.5,ppt



Wait 300ms

1

Dec

0

Null

S,?



Wait 300ms

1

Dec

?S,50000,μS

ASCII

0

Null

or

1

Dec

?S,37.5,ppt

ASCII

0

Null

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

# Atmospheric pressure compensation

## Command syntax

300ms  processing delay

P,n    n = any value in kPa

P,?    compensated pressure value?

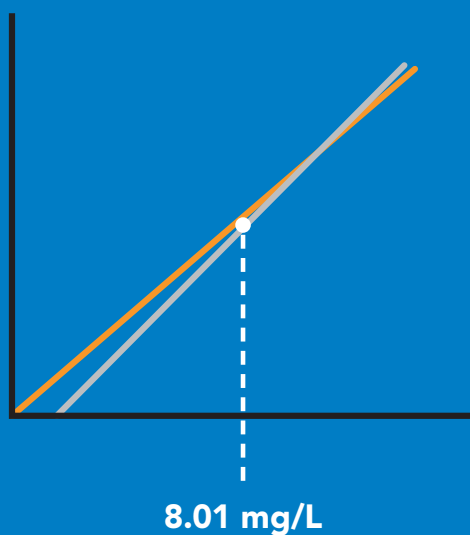
## Example

P,90.25

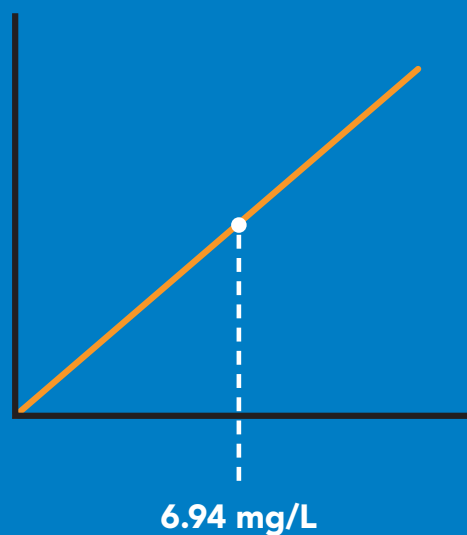
 Wait 300ms    1 Dec    0 Null

P,?

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



P,90.25



# 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

**? , O , % , mg**  
ASCII

**0**  
Null

if both are enabled

### Parameters

mg      mg/L  
%      percent saturation

### Followed by 1 or 0

1      enabled  
0      disabled

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

# Naming device

300ms  processing delay

## Command syntax

Do not use spaces in the name

Name,n set name

n =

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

Name, clears name

Up to 16 ASCII characters

Name,? show name

## Example

## Response

Name,



Wait 300ms

1

Dec

0

Null

name has been cleared

Name,zzt



Wait 300ms

1

Dec

0

Null

Name,?



Wait 300ms

1

Dec

?Name,zzt

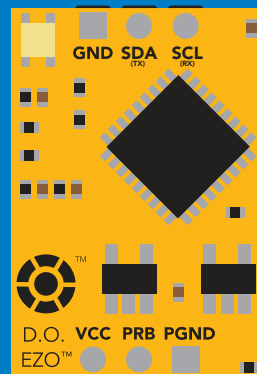
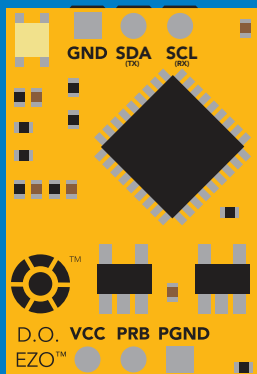
ASCII

0

Null

Name,zzt

Name,?



1

0

1

?Name,zzt

0

# Device information

## Command syntax

300ms  processing delay

i device information

## Example

i

## Response



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



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**

**5V**

STANDBY

**13.1 mA**

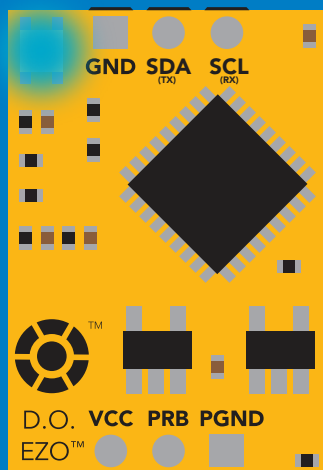
SLEEP

**0.66 mA**

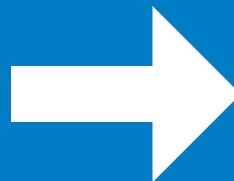
**3.3V**

**12 mA**

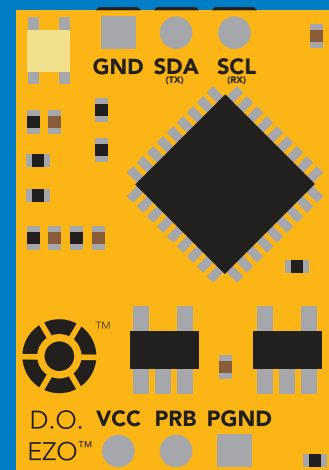
**0.3 mA**



**Standby**



**Sleep**



**Sleep**

# Protocol lock

## Command syntax

300ms  processing delay

Plock,1 enable Plock

Plock,0 disable Plock

Plock,? Plock on/off?

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

default

## Example

## Response

Plock,1

 Wait 300ms  
1 0  
Dec Null

Plock,0

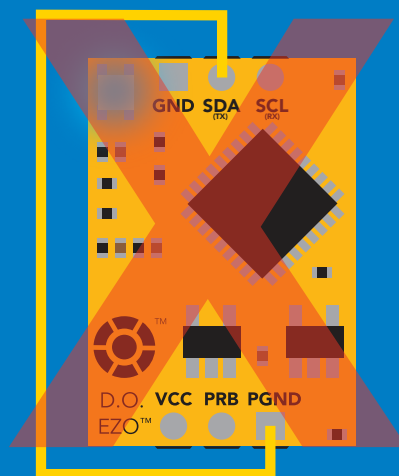
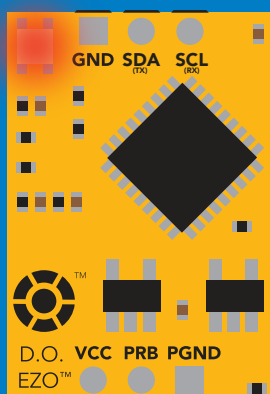
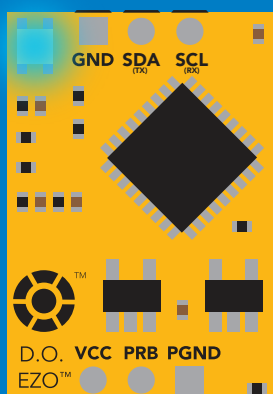
 Wait 300ms  
1 0  
Dec Null

Plock,?

 Wait 300ms  
1 ?Plock,1 0  
Dec ASCII 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

I2C,n sets I<sup>2</sup>C address and reboots into I<sup>2</sup>C mode

## Example

I2C,100

## Response

device reboot  
(no response given)

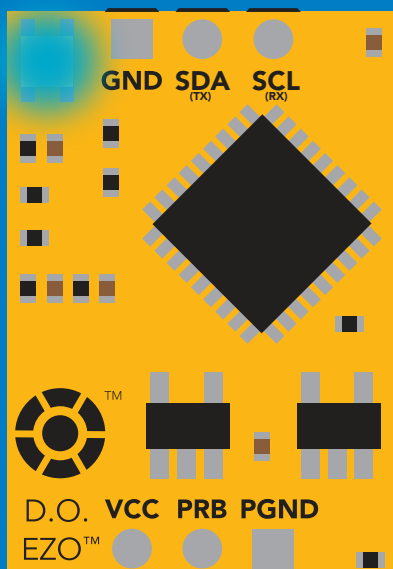
## Warning!

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.

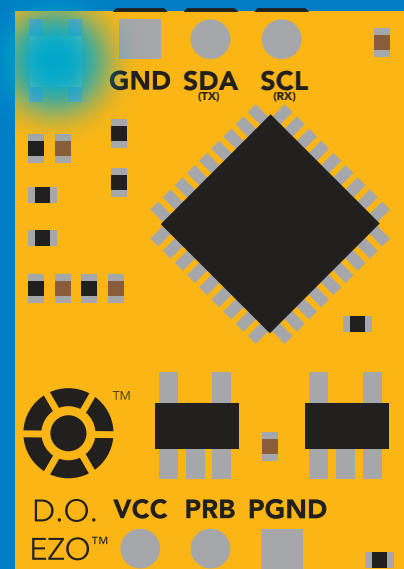
Default I<sup>2</sup>C address is 97 (0x61).

n = any number 1 – 127

I2C,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

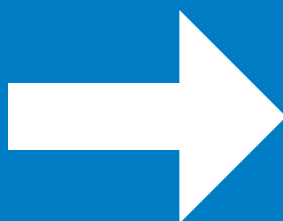
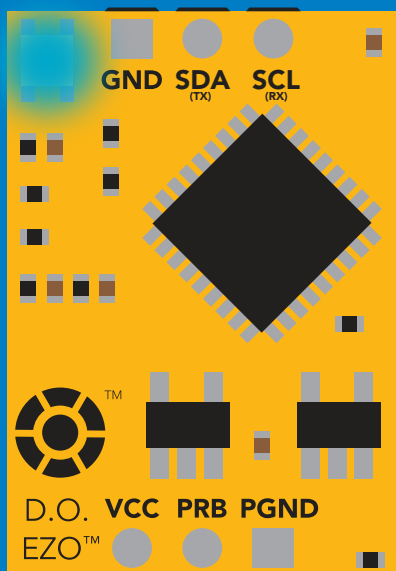
## Response

**Factory**

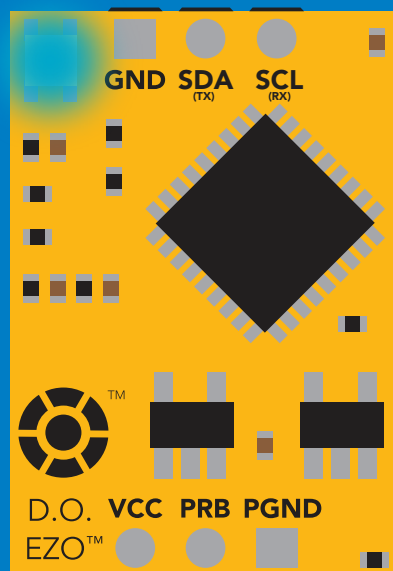
**device reboot**  
(no response given)

Clears calibration  
LED on  
Response codes enabled

## Factory



(reboot)



# Change to UART mode

## Command syntax

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

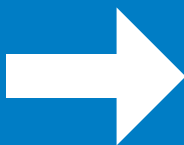
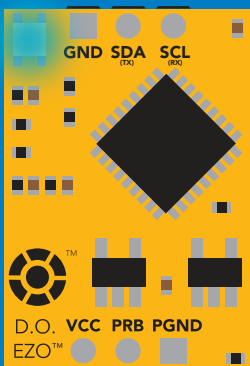
### Example

Baud,9600

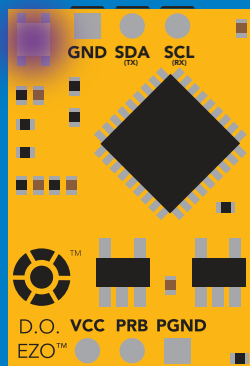
### Response

reboot in UART mode  
(no response given)

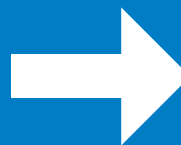
n =  $\left\{ \begin{array}{l} 300 \\ 1200 \\ 2400 \\ 9600 \\ 19200 \\ 38400 \\ 57600 \\ 115200 \end{array} \right.$



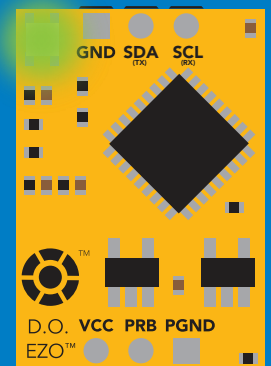
Baud,9600



Changing to  
UART mode



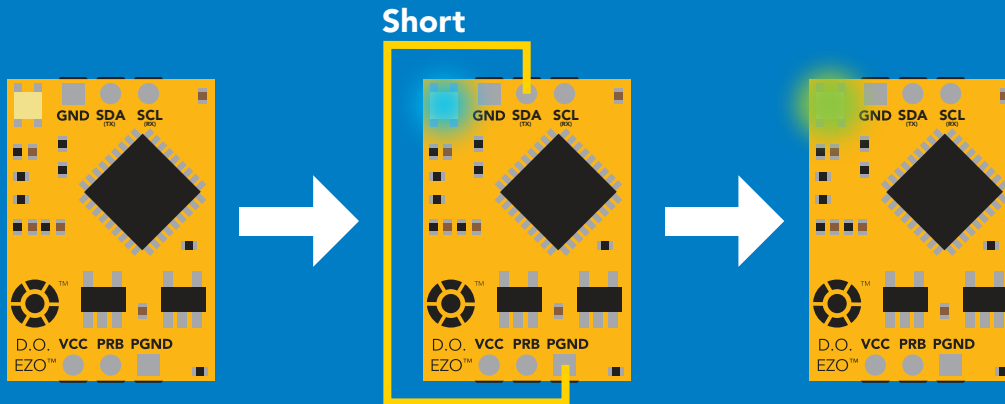
(reboot)



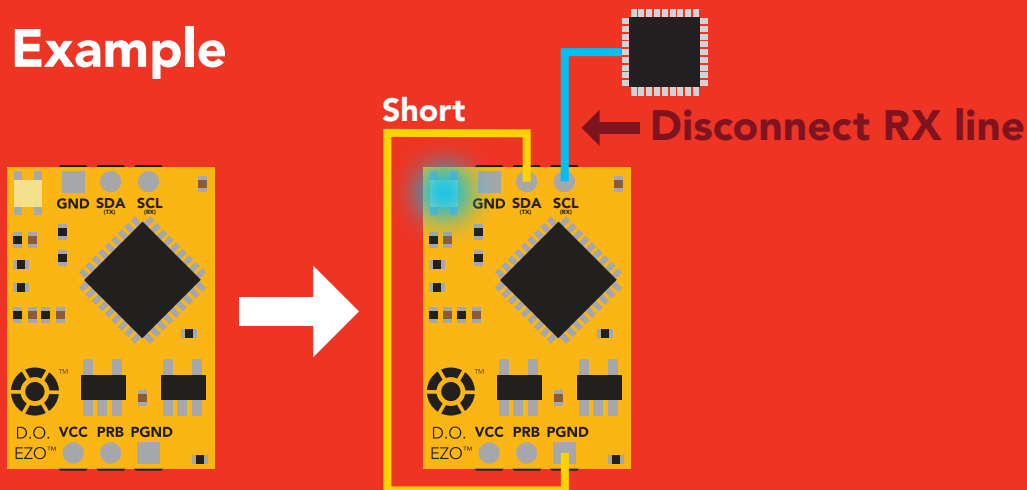
# Manual switching to UART

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

## Example



## Wrong Example



# Calibration theory

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

## Confirm the D.O. probe is working correctly

Take readings in air first.



Readings > 10



Readings < 5 or > 25

*Refer to probes datasheet  
for instructions.*

## Calibrate first, compensate later

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

### Default compensation values

**Temp** = 20 °C  
**Pressure** = 101 kPa  
**Salinity** = 0

**Temp** = 29 °C  
**Pressure** = 93 kPa  
**Salinity** = 5

*(too many variables)*

### Known calibration value

**9.09 Mg/L**

**???**  
**(6.84 Mg/L)**

# Best practices for calibration

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



## ⚠️ Never do a blind calibration! ⚠️

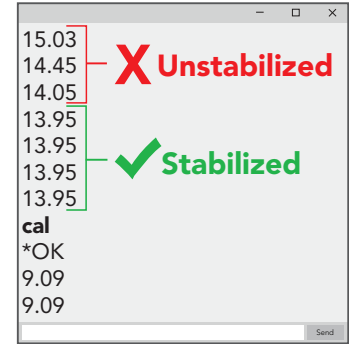
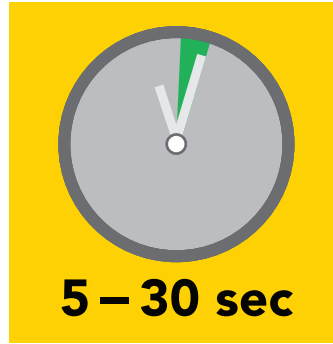
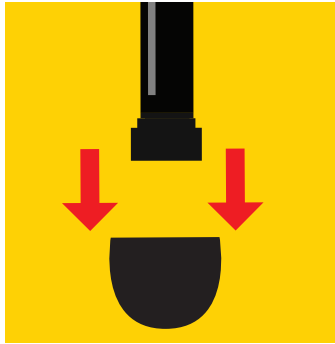
Issuing a calibration command before the readings stabilize will result in drifting readings.



# Calibration order

## High point calibration

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

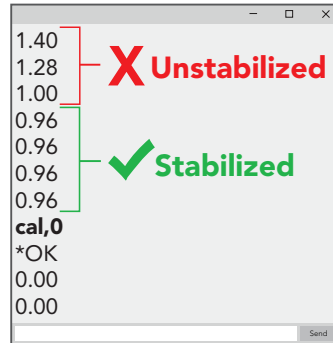
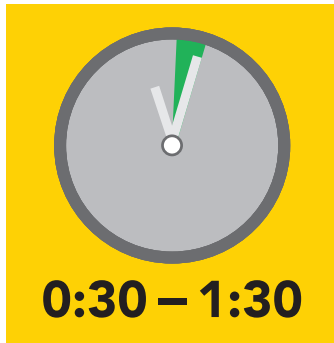


After calibration is complete, you should see readings between **9.09 – 9.1X mg/L**.  
(only if temperature, salinity and pressure compensation are at default values)



## Low point calibration

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



# Advanced calibration

## Probe temperature calibration

### Probe temperature calibration $\neq$ Temperature compensation.

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

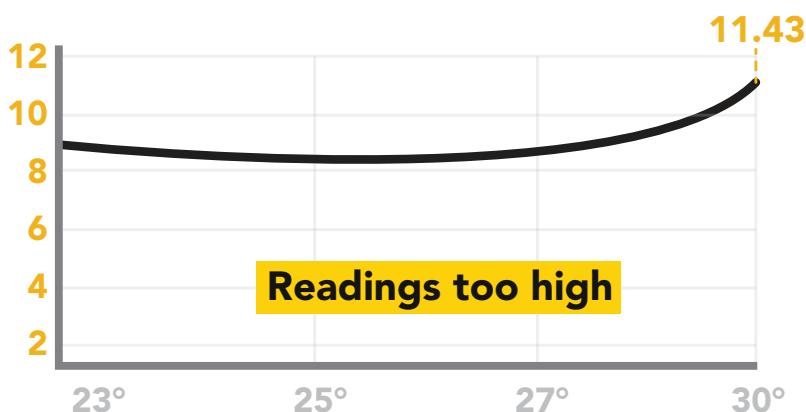
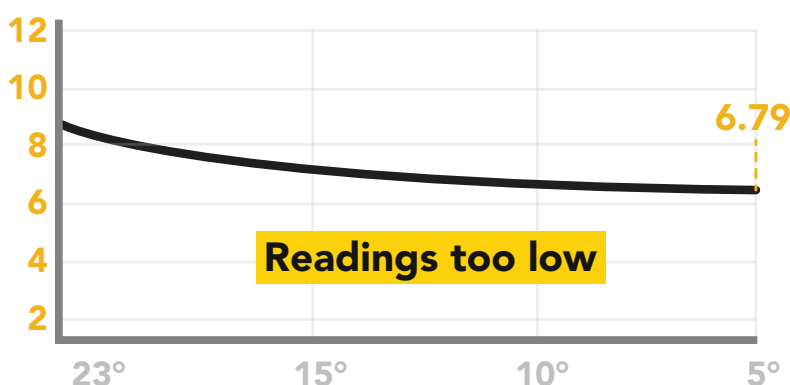
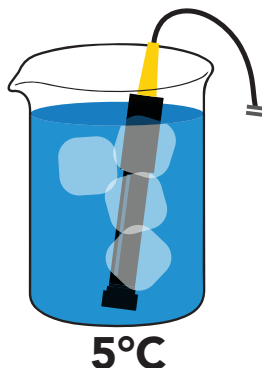
#### Calibrated probe

Air temperature

**$23^\circ\text{C}$**

Air Reading

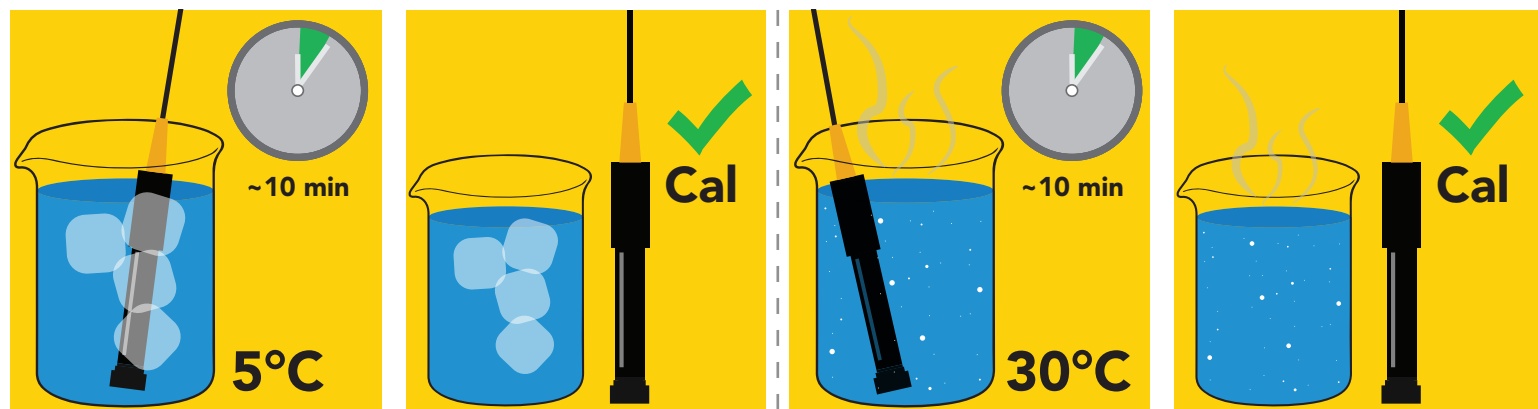
**$9.10\text{ mg/L}$**



# Advanced calibration

## What to do:

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

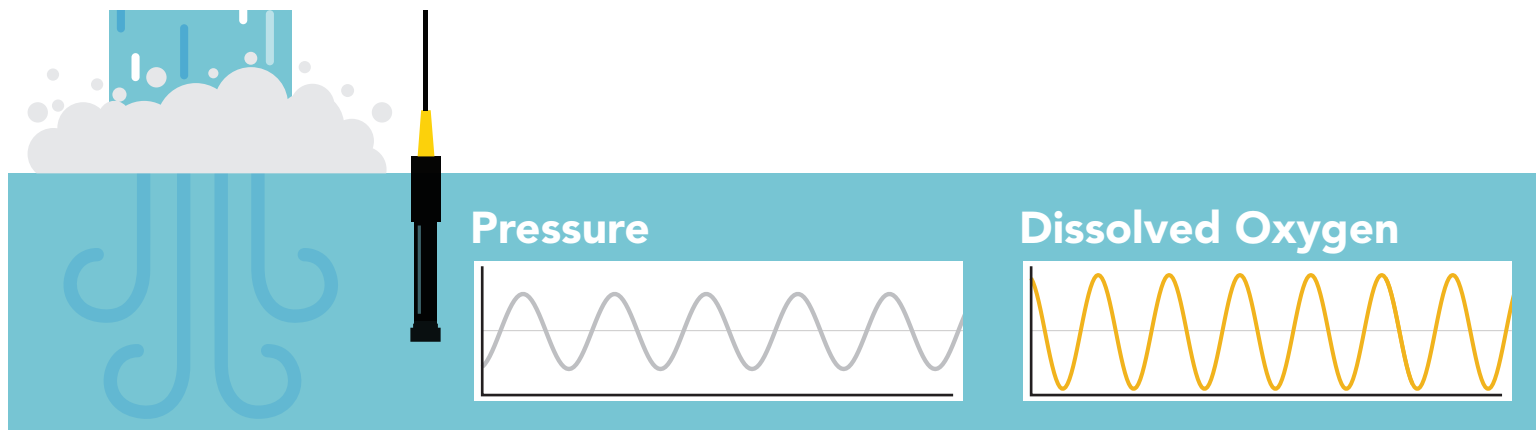


Let the probe acclimate to its operating temperature and then recalibrate. Once the probe has been calibrated at its intended operating temperature, using temperature compensation will give accurate readings.

# Understanding D.O. measurements

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

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



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

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

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

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

**Temperature**

**Salinity**

**Atmospheric Pressure**

**Temperature**

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

**Pure water at 1°C can hold 14.2 mg/L**

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

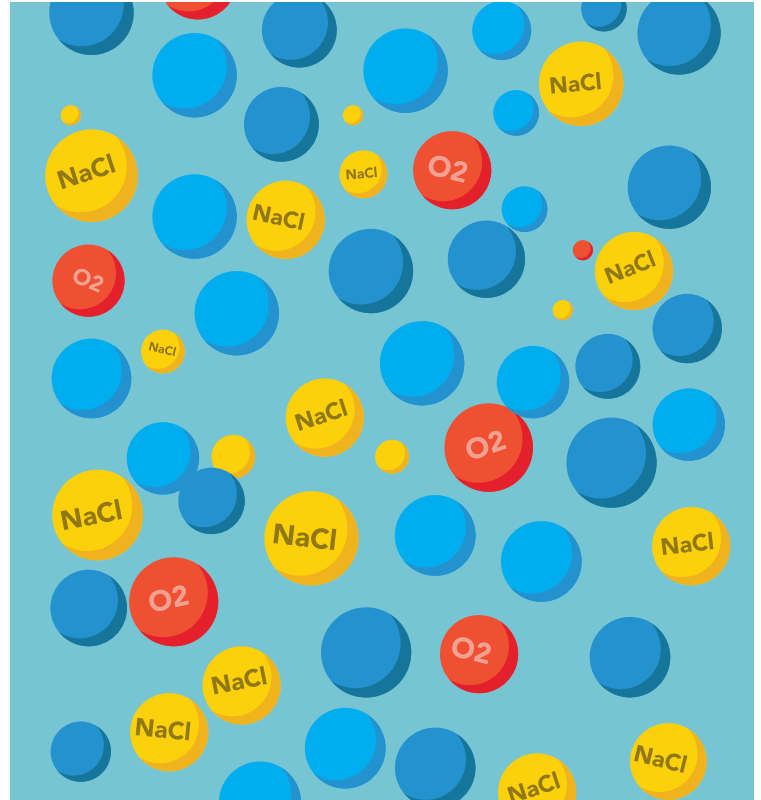
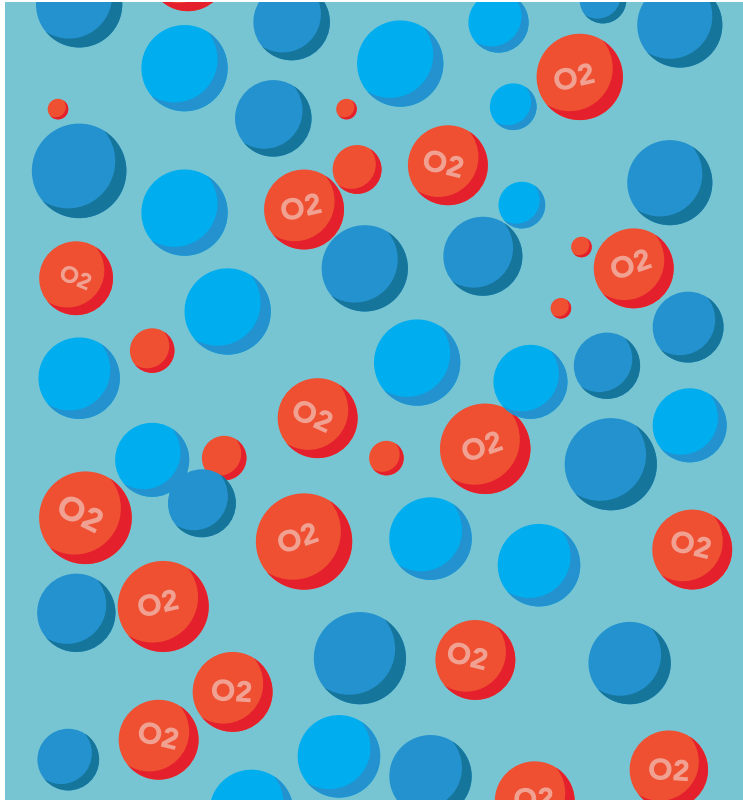
# Understanding D.O. measurements

## Salinity

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

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

**Pure water at 1°C can hold 14.2 mg/L**



## Atmospheric Pressure

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

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

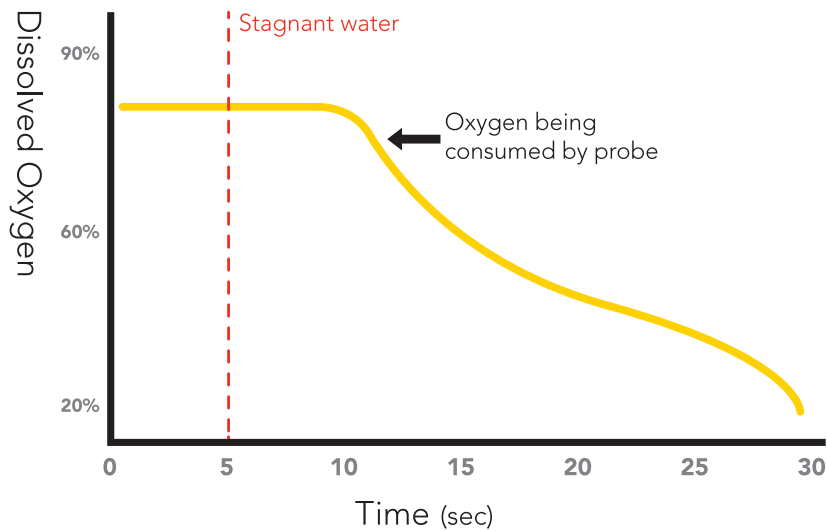
As altitude increases, oxygen concentration decreases, and because D.O. readings are expressed in Mg/L, the oxygen concentration must be known.

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

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

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

# Flow Dependence



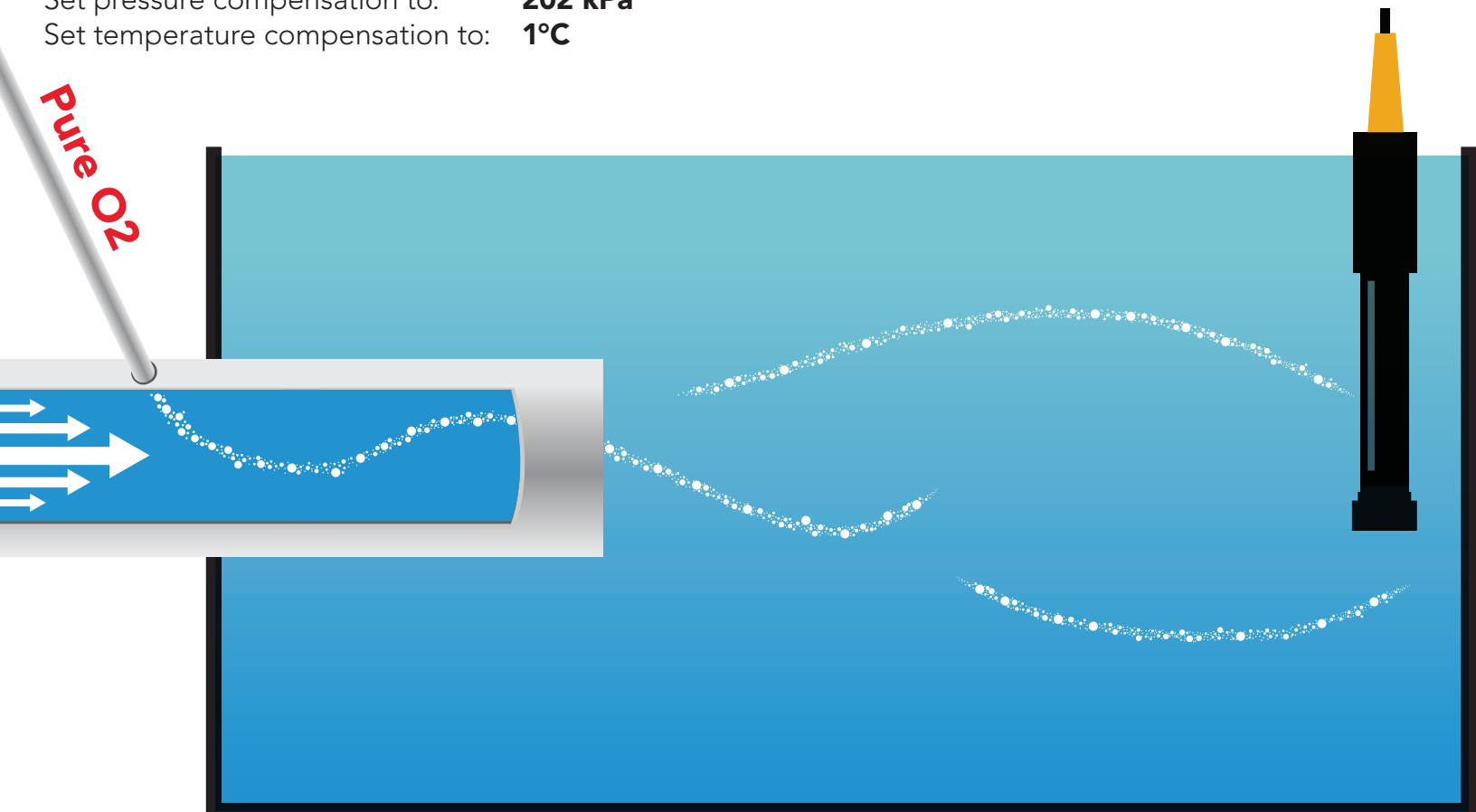
One of the drawbacks from using a galvanic probe is that it consumes a **VERY** small amount of the oxygen it reads. Therefore, a small amount of water movement is necessary to take accurate readings. **Approximately 60 ml/min.**

## Hyper saturation with pure oxygen

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

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

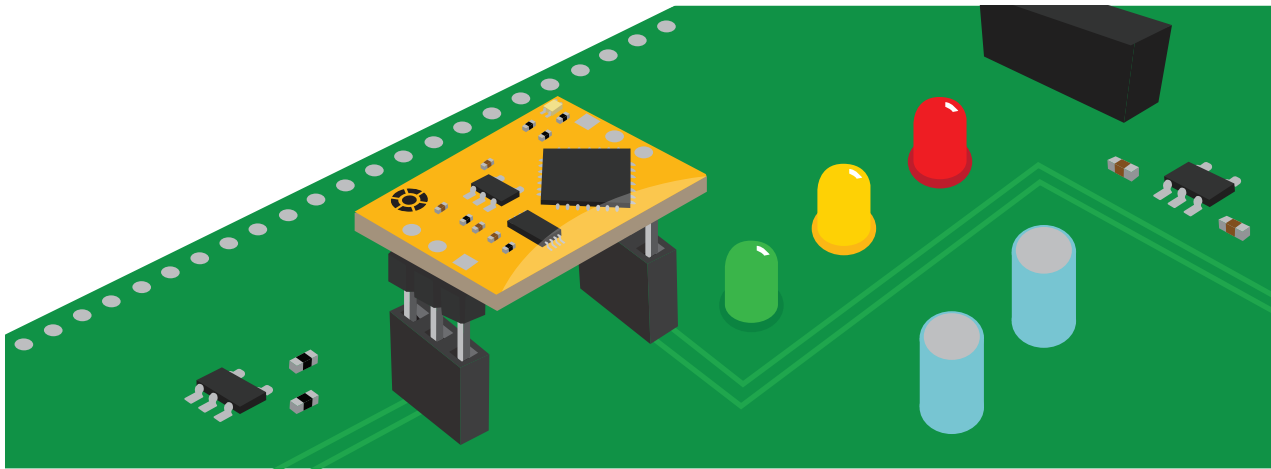
Set pressure compensation to: **202 kPa**  
Set temperature compensation to: **1°C**



# Soldering

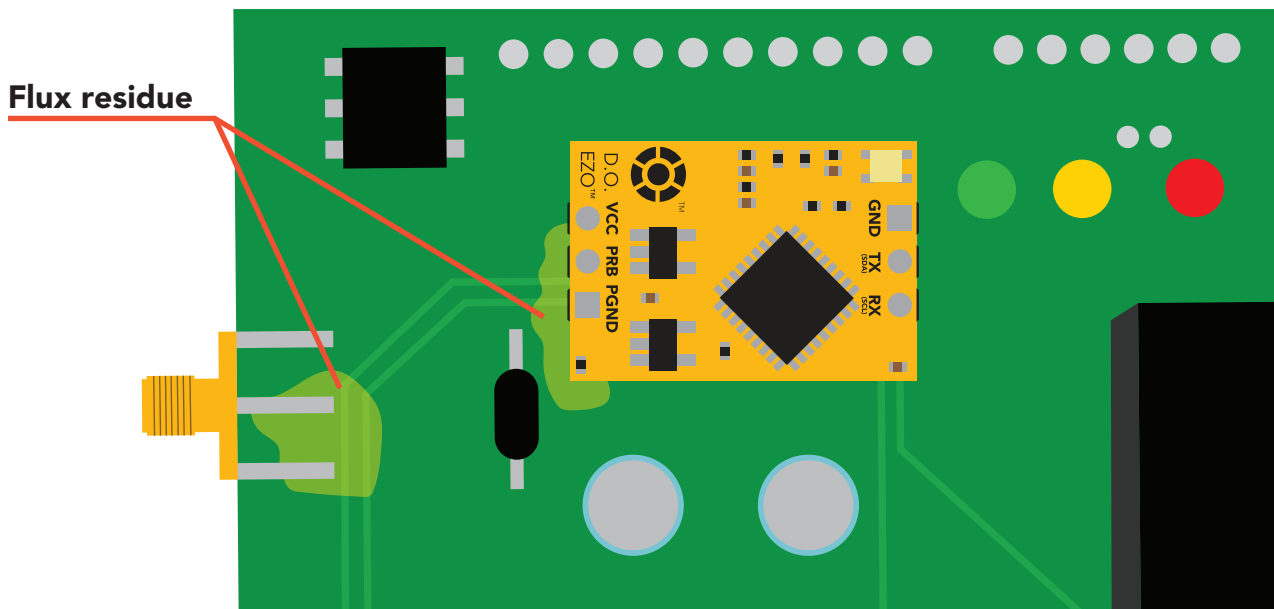
**Do not directly solder an EZO circuit to your PCB.** If something goes wrong during the soldering process it may become impossible to correct the problem. It is simply not worth the risk.

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



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

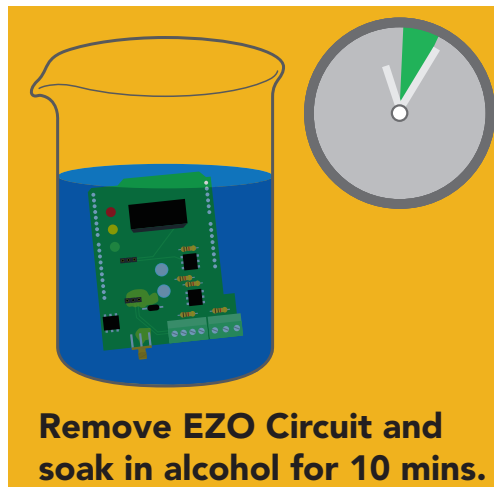
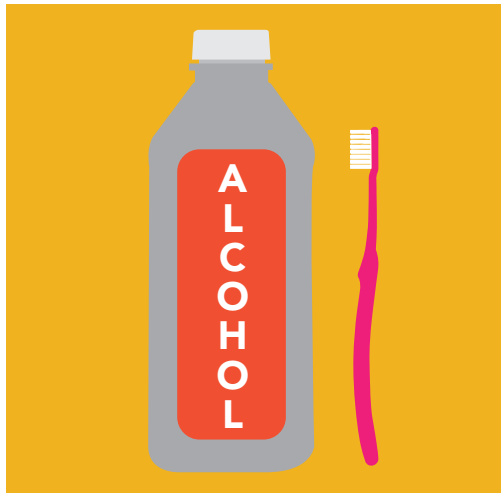
**Flux residue will severely affect your readings.** Any Flux residue that comes in contact with the PRB pins or your probes connector will cause a “flux short”.



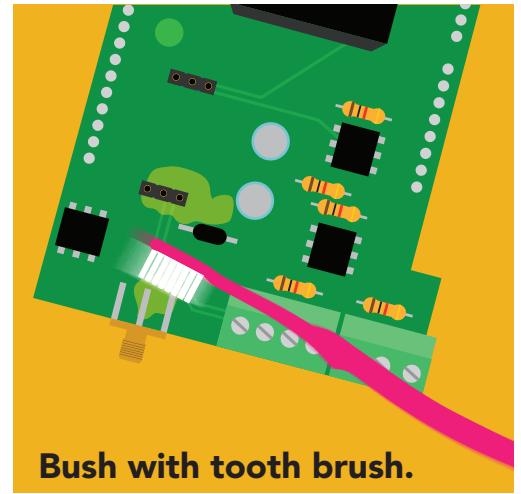
You **MUST** remove all the flux residue from your PCB after soldering.

# Soldering

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



**Remove EZO Circuit and soak in alcohol for 10 mins.**



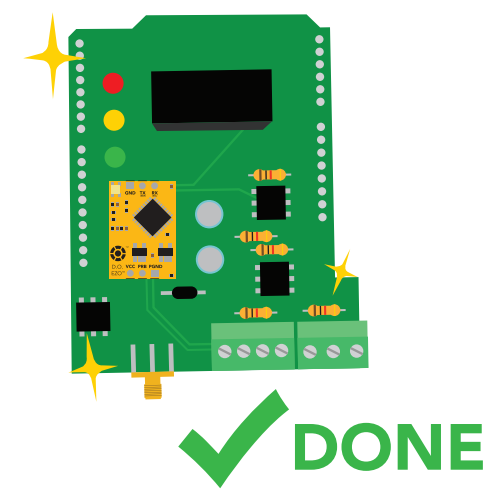
**Bush with tooth brush.**



**Soak in alcohol for 5 mins.**



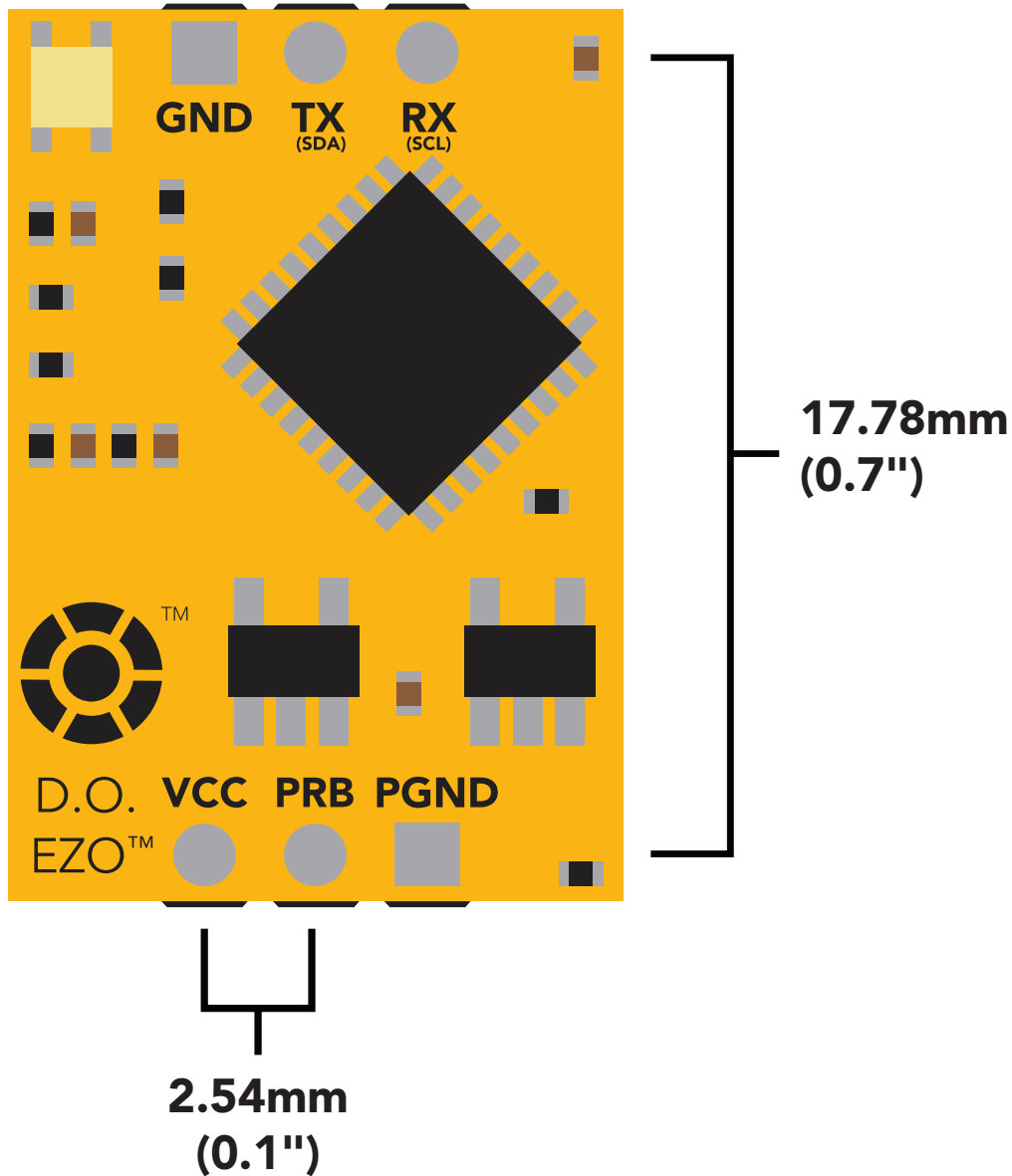
**Let it dry in the air.**



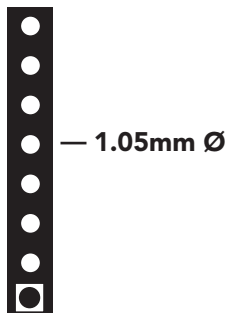
## What does a flux short look like?

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

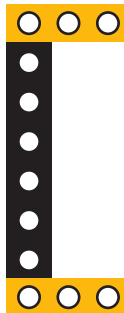
# EZO™ circuit footprint



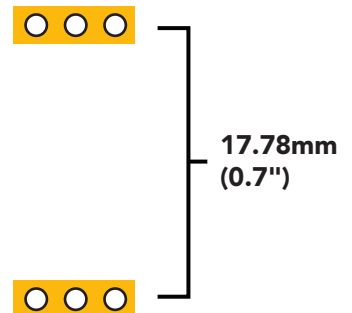
**1** In your CAD software place a 8 position header.



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



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



# Datasheet change log

## Datasheet V 5.6

Revised entire document.

## Datasheet V 5.5

Revised naming device info on pages 32 & 59.

## Datasheet V 5.4

Revised artwork within datasheet.

## Datasheet V 5.3

Moved Default state to pg 13.

## Datasheet V 5.2

Updated firmware changes on page 70.

## Datasheet V 5.1

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

## Datasheet V 5.0

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

## Datasheet V 4.9

Corrected temperature compensation typo on pages 26 & 52.

## Datasheet V 4.8

Revised isolation schematic on pg. 10

## Datasheet V 4.7

### **Added new command:**

"RT,n" for Temperature compensation located on pages 26 (UART) & 52 (I<sup>2</sup>C).

Added firmware information to Firmware update list.

# Datasheet change log

## Datasheet V 4.6

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

## Datasheet V 4.5

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

## Datasheet V 4.4

Removed note from certain commands about firmware version.

## Datasheet V 4.3

Added information to calibration theory on pg 7.

## Datasheet V 4.2

Revised definition of response codes on pg 44.

## Datasheet V 4.1

Updated firmware changes on pg. 66.

## Datasheet V 4.0

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

# Firmware updates

V2.15 – (Sept 8, 2022)

- Internal update for new part compatibility.

# Warranty

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

## The debugging phase

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

**It is important for the embedded systems engineer to keep in mind that the following activities will void the EZO™ class Dissolved Oxygen circuit warranty:**

- **Soldering any part of the EZO™ class Dissolved Oxygen circuit.**
- **Running any code, that does not exclusively drive the EZO™ class Dissolved Oxygen circuit and output its data in a serial string.**
- **Embedding the EZO™ class Dissolved Oxygen circuit into a custom made device.**
- **Removing any potting compound.**

# Reasoning behind this warranty

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

## Please keep this in mind:

- 1. All Atlas Scientific™ devices have been designed to be embedded into a custom made system by you, the embedded systems engineer.**
- 2. All Atlas Scientific™ devices have been designed to run indefinitely without failure in the field.**
- 3. All Atlas Scientific™ devices can be soldered into place, however you do so at your own risk.**

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