AtlasScientific™

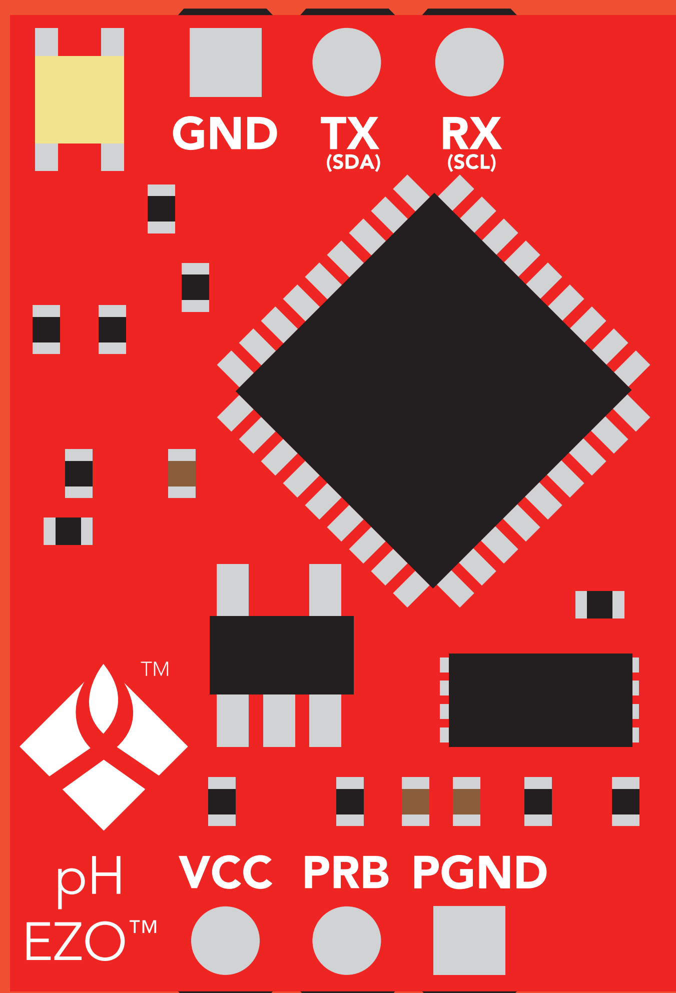
Environmental Robotics

V 3.7

Revised 9/16/16

pH EZO™ Circuit

Range	**.001 – 14.000**
Max rate	**1 reading per sec**
Supported probes	**Any type & brand**
Calibration	**1, 2, 3 point**
Temp compensation	**Yes**
Data protocol	**UART & I²C**
Default I²C address	**(0x63)**
Operating voltage	**3.3V – 5V**
Data format	**ASCII**



Patent pending

This is an evolving document, check back for updates.

STOP

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.

Reads

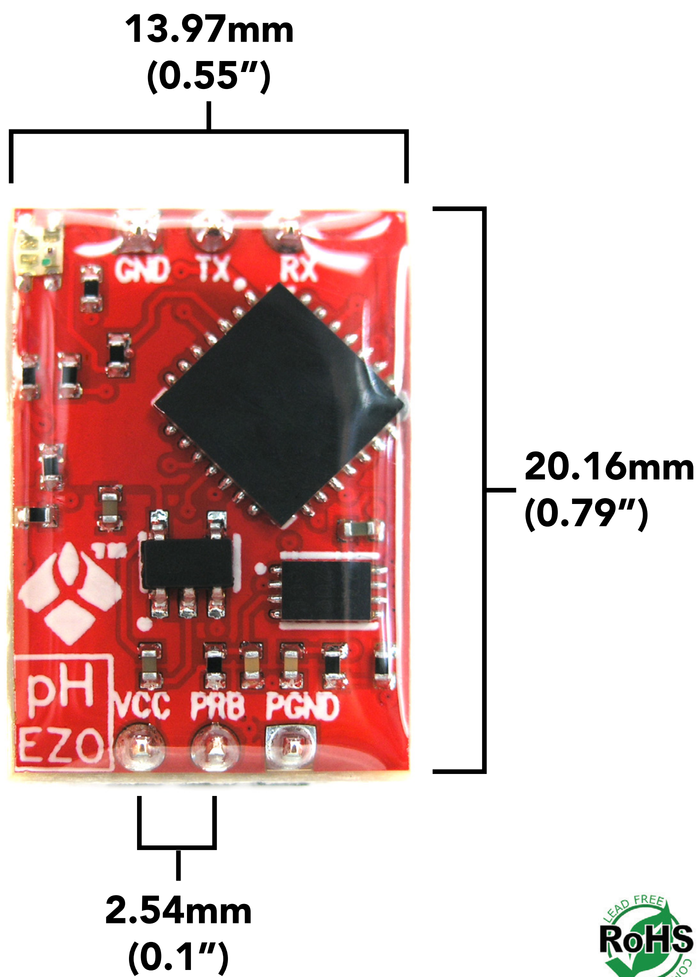
- Full range pH reading from .001 – 14.000
- Accurate pH readings down to the thousandths place **(+/- 0.02)**
- Temperature dependent or temperature independent readings
- Flexible calibration protocol supports single point, 2 point, or 3 point calibration
- Calibration required only once per year with Atlas Scientific pH probe
- Single reading or continuous reading modes
- **Data format is ASCII**

Two data protocols

- UART asynchronous serial connectivity
- (RX/TX voltage swing 0-VCC)
- I^2C (default I^2C address 0x63)
- Compatible with any microprocessor that supports UART, or I^2C protocol
- Operating voltage: 3.3V to 5V
- Works with any off-the-shelf pH probe

Sleep mode power consumption

- 0.995mA at 3.3V



The Atlas Scientific™ pH EZO™ circuit, is our 6th generation embedded pH circuit. This pH EZO™ circuit, offers the highest level of stability and accuracy. With proper configuration the pH EZO™ circuit, can meet, or exceed the accuracy and precision found in most bench top laboratory grade pH meters. The pH EZO™ circuit, can work with any off-the-shelf pH probe/sensor/electrode.

This device reads pH from a pH probe/sensor/electrode. This device does not include a pH probe/sensor/electrode.

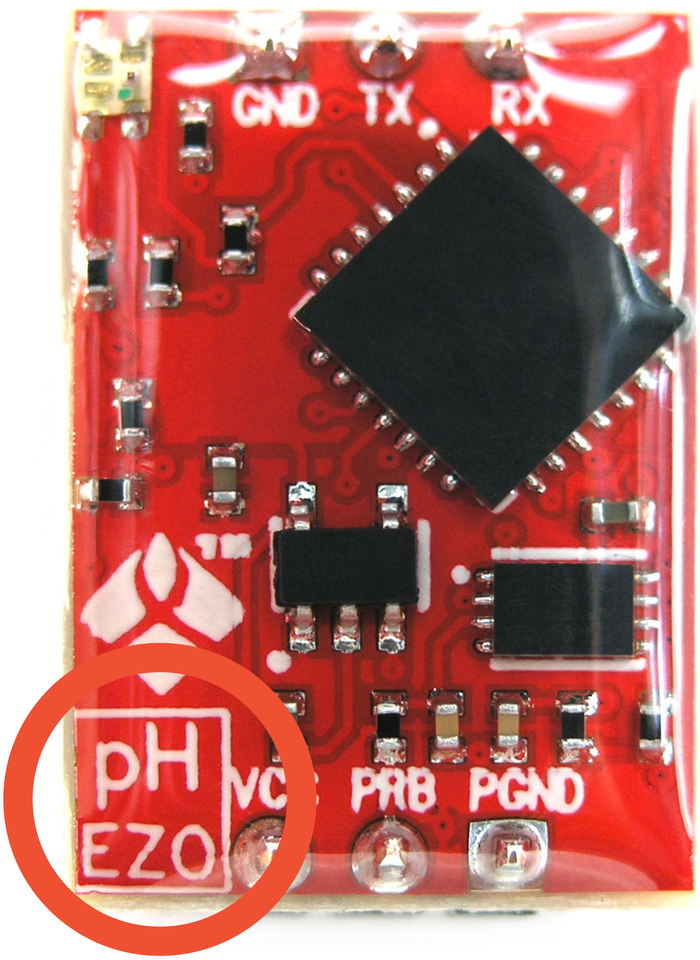
AtlasScientific™
Environmental Robotics
pH EZO™ Circuit

Contents

UART Mode

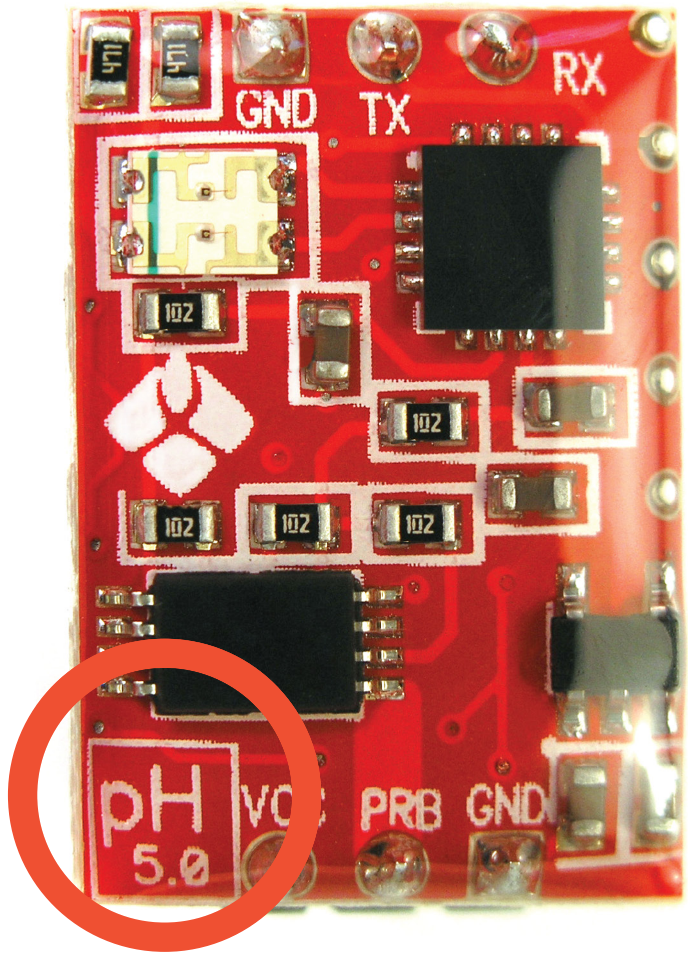
I²C Mode

Circuit identification



EZO™ Class Circuit

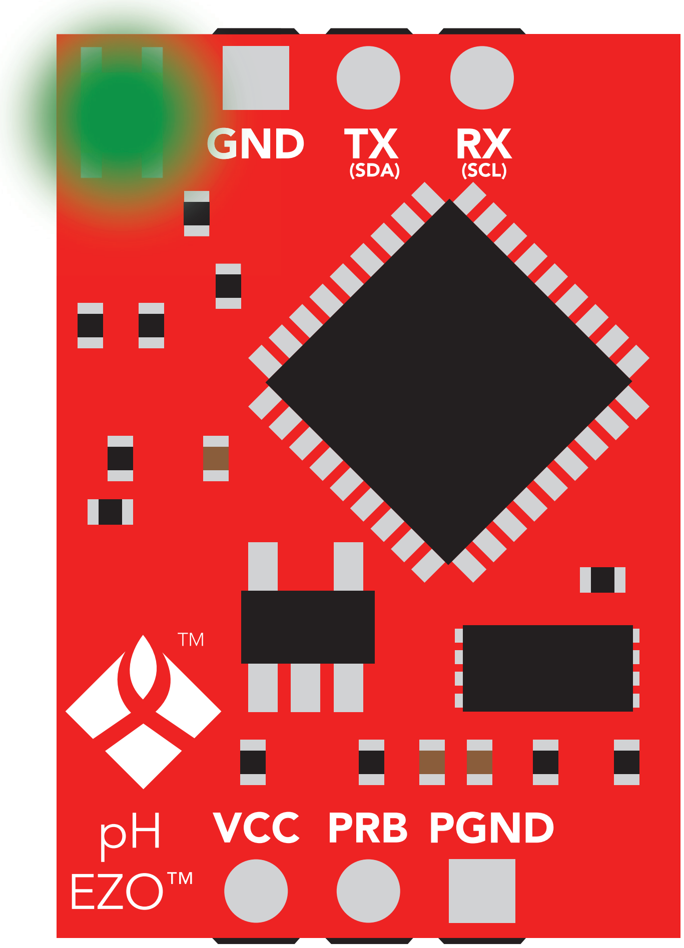
If your Atlas Scientific™ pH Circuit says "pH EZO" you are viewing the correct datasheet.

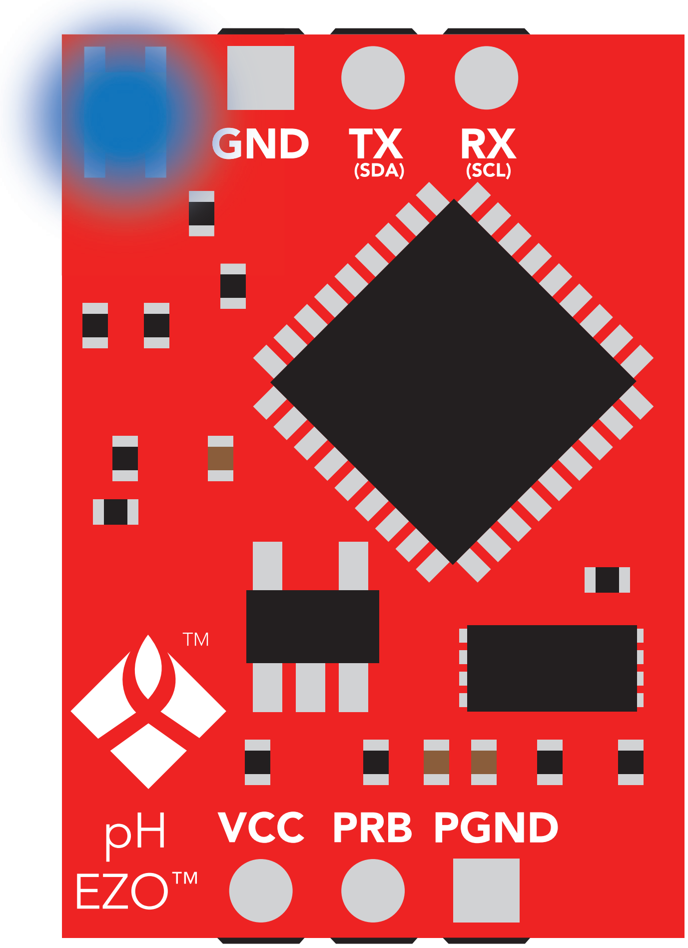


Legacy Circuit

If your Atlas Scientific™ pH Circuit says "pH 3.0, 4.0, or 5.0" you are viewing the *incorrect datasheet*. Many functions will not will not work on legacy circuits.

To view the legacy datasheet **click here**.





UART Mode

I²C Mode

System overview

The EZO™ class pH circuit, is a small footprint computer system that is specifically designed to be used in robotics applications where the embedded systems engineer requires accurate and precise measurements of pH.

The EZO™ class pH circuit, is capable of reading pH, down to the thousandths place.

Example

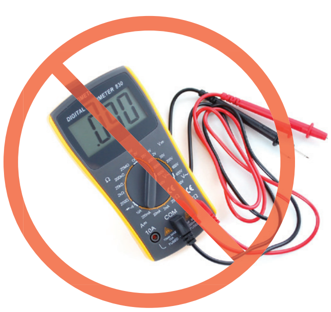
pH = 4.768

In order to offer such resolution, considerable effort has been put into the design of the Atlas Scientific EZO™ class pH circuit. Components used, PCB topography and board metallurgy are all factors in achieving precise, high resolution readings. The Atlas Scientific EZO™ class pH circuit, converts a current generated by hydrogen ion activity into the pH. The current that is generated from the hydrogen ion activity is the reciprocal of that activity and can be predicted using this simple equation:

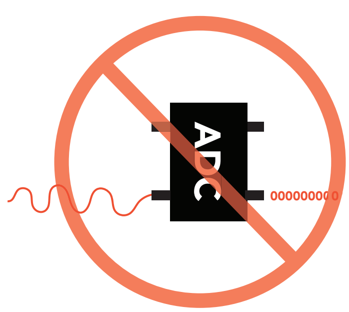
$$E = E^0 + \frac{RT}{F}\ln(\alpha_{H+}) = E^0 - \frac{2.303RT}{F}pH$$

Where **R** is the ideal gas constant.
T is the temperature in Kelvin.
F is the Faraday constant.

It is important for the embedded systems engineer to keep in mind that it is not possible to simply read the current coming off of a pH probe and convert that voltage into a pH using an ADC.



Result will **always** read zero.



Result will **always** read zero.

Power consumption

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

Absolute maximum ratings*

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

***Note:** Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. Exposure to maximum rating conditions for extended periods may affect device reliability

Pin Out

GND Return for the DC power supply

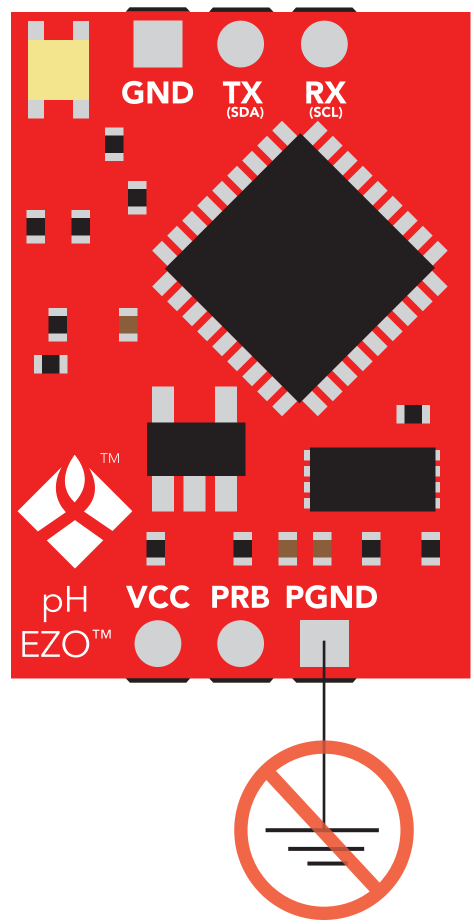
Vcc Operates on 3.3V – 5.5V

TX / SDA All EZO™ class circuits can operate in either UART mode, or I^2C mode

The default state is UART mode.
In UART mode, this pin acts as the transmit (TX) line.
The default baud rate is 9600, 8 bits, no parity, no flow control, one stop bit. If standard RS232 voltage levels are desired, connect an RS232 converter such as a MAX232.
If the devices is in I^2C mode, this pin acts as the Serial Data Line (SDA). The I^2C protocol requires an external pull up resistor on the SDA line (resistor not included).

RX / SCL All EZO™ class circuits can operate in either UART mode, or I^2C mode.

The default state is UART mode.
In UART mode, this pin acts as the receive (RX) line.
If the devices is in I^2C mode, this pin acts as the Serial Clock Line (SCL). The I^2C protocol requires an external pull up resistor on the SCL line (resistor not included).



PRB This pin connects to the output lead of a pH probe/ sensor/electrode

PGND This pin connects to the ground lead of a pH probe/ sensor/electrode

This pin is not ground.
Do not tie this pin to system ground

Device operation

When an EZO™ class circuit is first powered up the boot sequence will begin. This is indicated by the LED moving from **Red** to **Green** to **Blue**. The boot up sequence takes 1 second. Once the device has booted up the circuit will output:

*RS<CR>
*RE<CR>
Indicating the device is ready for operation.

The LED will enter its default blink pattern *(see page 10)* indicating the device is operational and actively taking readings.

Default state

Mode
UART

Baud rate
9600 bps
8 data bits
1 stop bit
no parity
no flow control

Reading time
1 reading every second

Probe type
Any off the shelf single, or double junction pH probe/sensor/electrode

LEDs:
Enabled
Steady **Green**= Power on/ standby
Red double blink = Command received and not understood
Green double blink per data packet = Continuous data streaming
Cyan = taking a reading

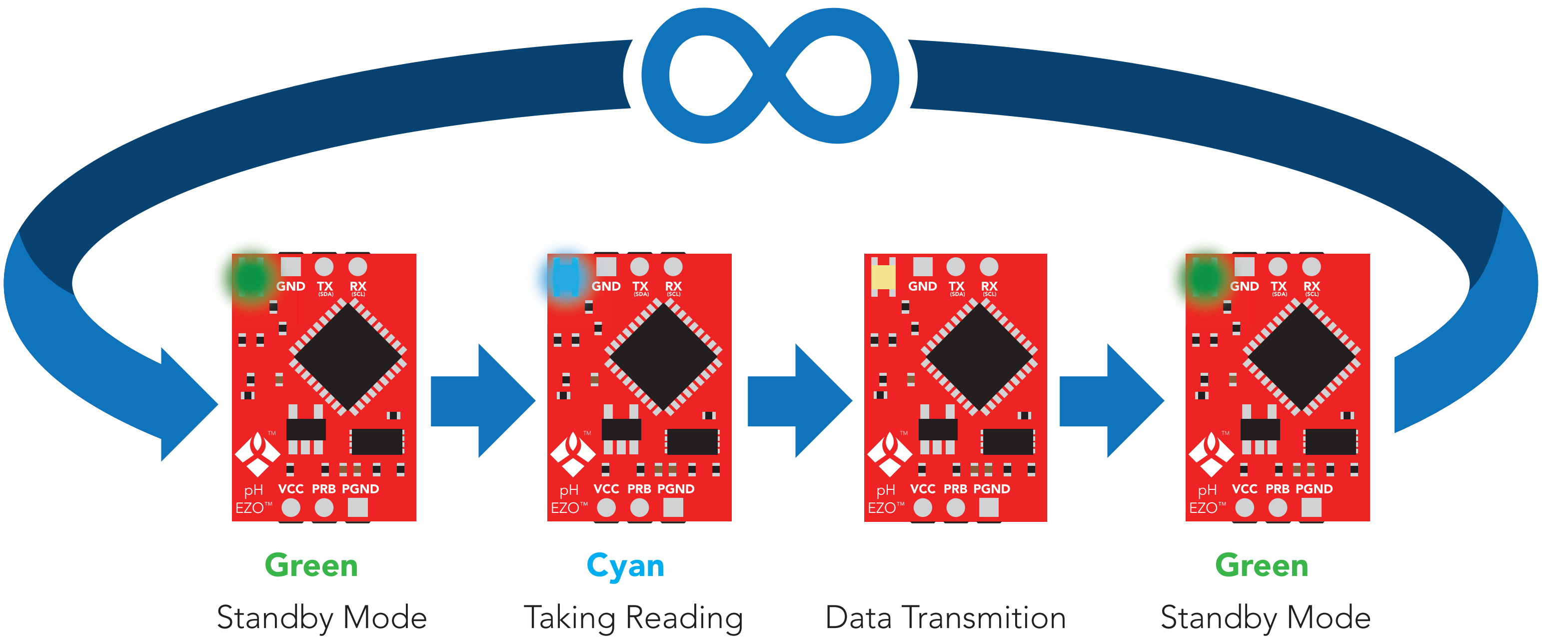
Data output
pH

Units	Data Type	Format
pH	Floating point	String

Encoding
ASCII characters followed by a carriage return <CR>
Maximum string length: 10 characters

If the response code is enabled the EZO™ class circuit will respond "*OK<CR>" after a command is acknowledged. If an unknown command is sent the pH Circuit will respond "*ER<CR>" this will happen whether or not response codes are enabled.

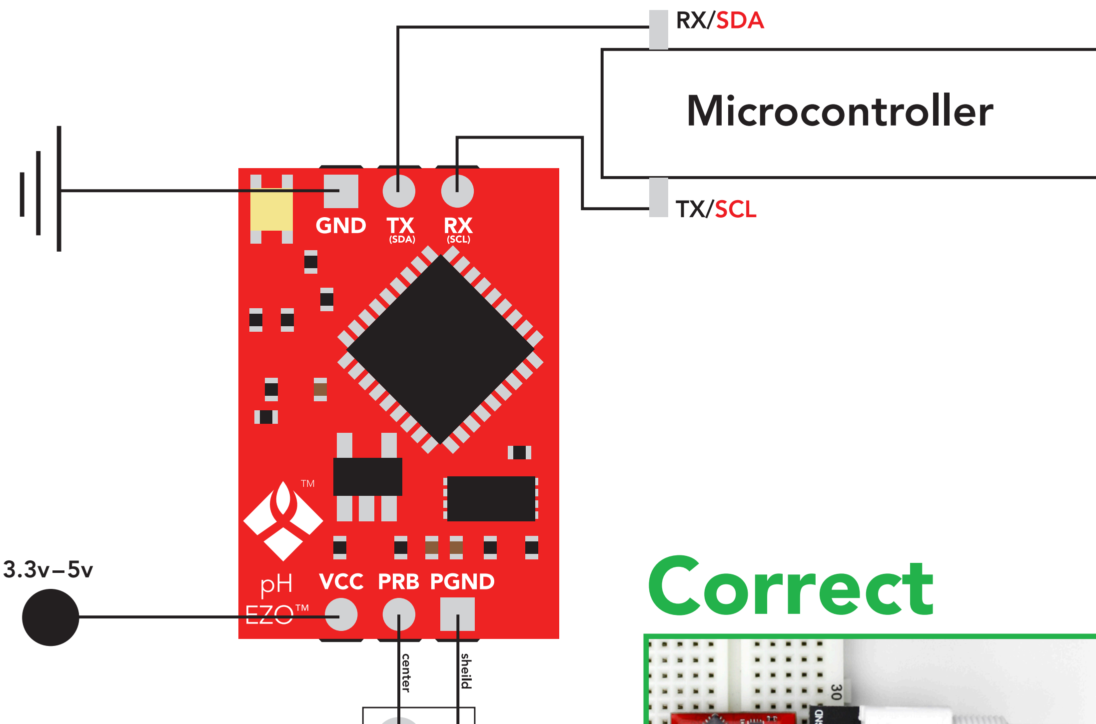
Default LED blink pattern



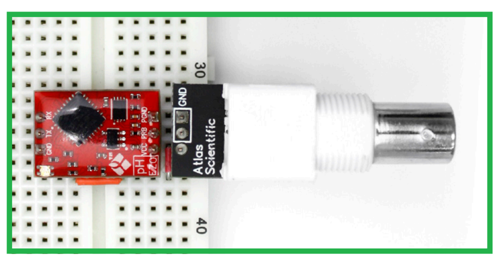
This is the LED pattern for Continous Mode (which is the default state)
This can only happen when the device is in UART mode.

Wiring diagram

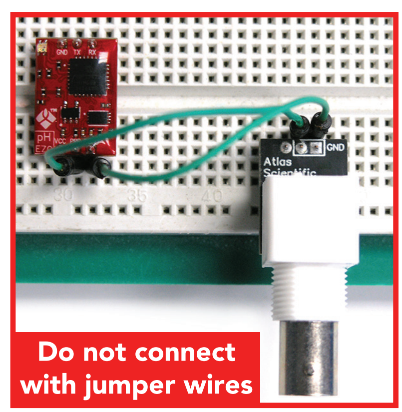
- To connect the Circuit to your microcontroller, follow the diagram below.
- Make sure your Circuit and microcontroller share a common ground.
- TX on your Circuit connects to RX on your microcontroller.
- If in I²C mode connect SDA to SDA and SCL to SCL
- *4.7k pull up resistor on SDA and SCL may be required

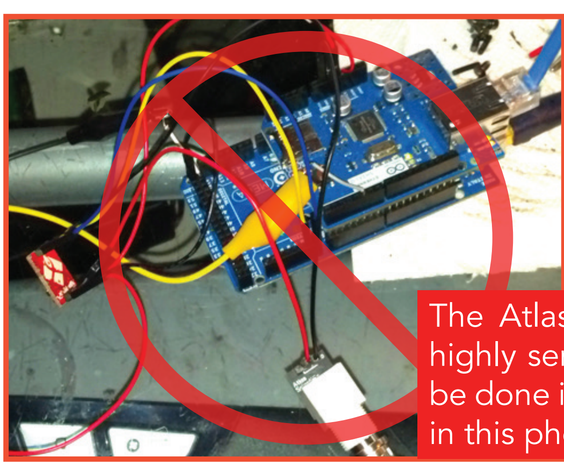


Correct

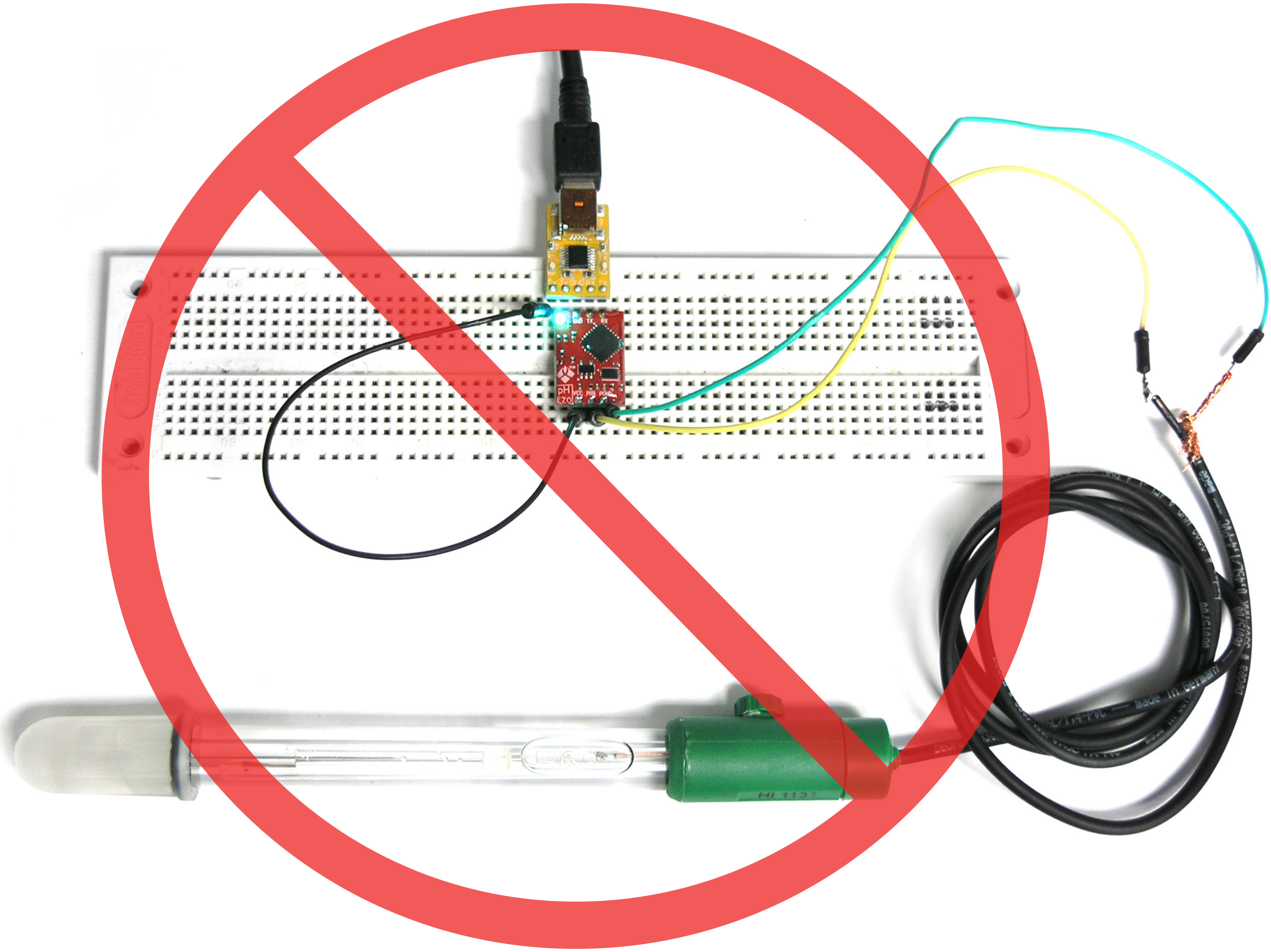


Incorrect



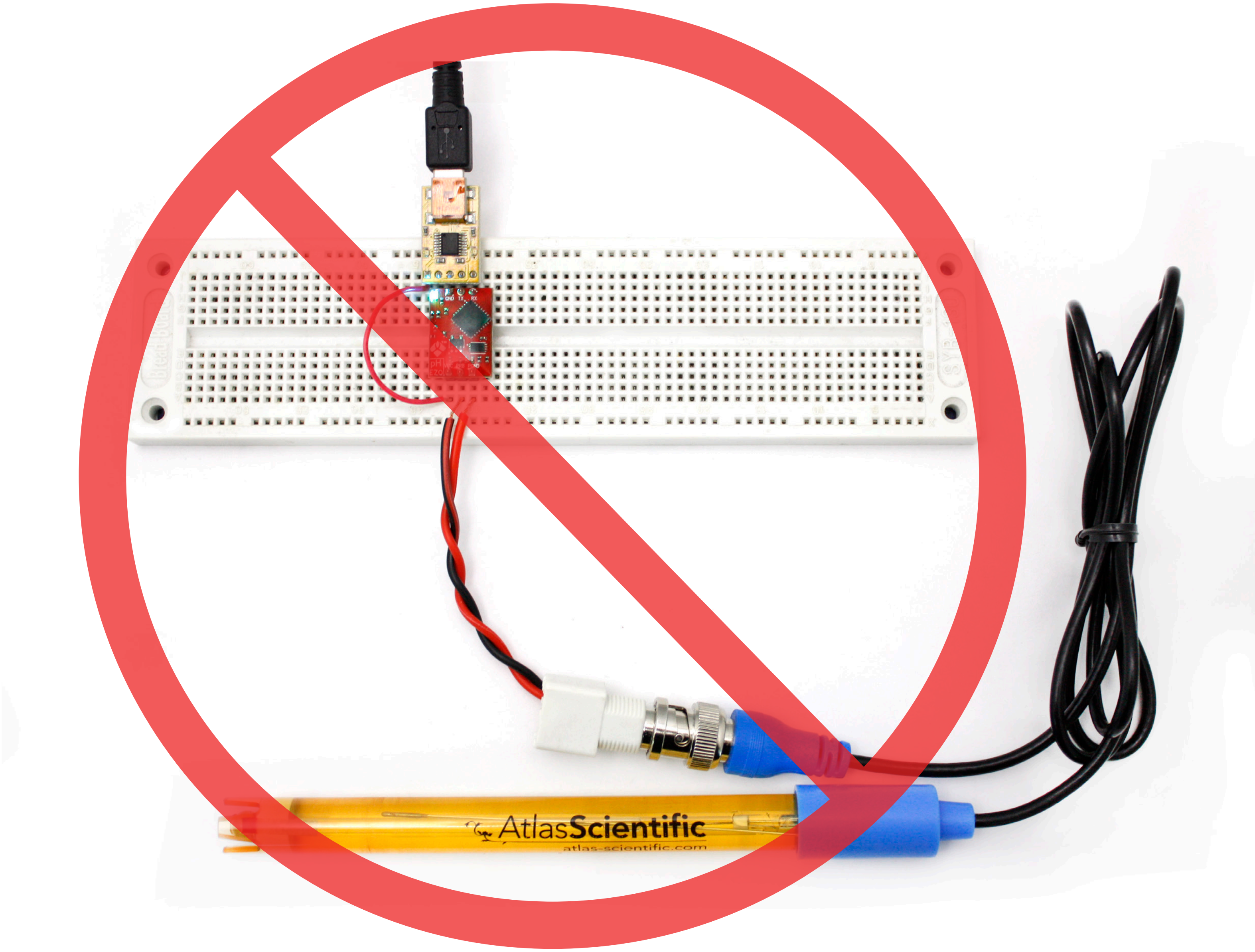


The Atlas Scientific™ EZO™ class pH circuit is highly sensitive equipment. Debugging should be done in a bread board; Not like what is show in this photo.



DO NOT CUT THE CABLE WITHOUT REFERING TO THIS DOCUMENT!

NEVER EXTEND THE CABLE WITH CHEAP JUMPER WIRES.



DO NOT MAKE YOUR OWN UNSHIELDED CABLES
REFER TO THIS DOCUMENT!

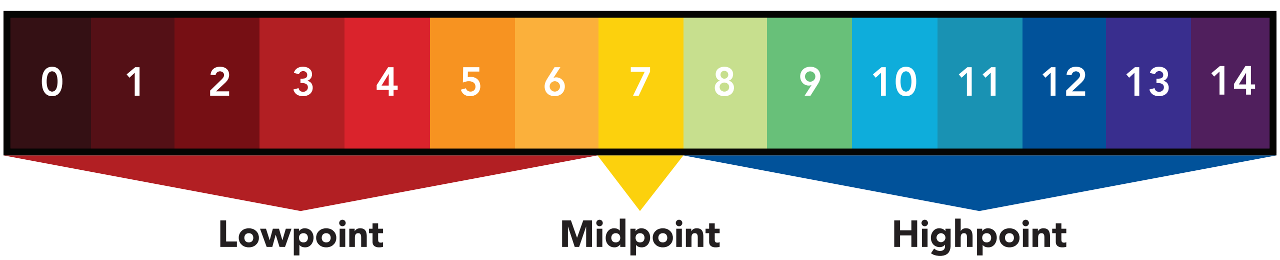
NEVER EXTEND THE CABLE
WITH CHEAP JUMPER WIRES.
ONLY USE SHIELDED CABLES

Calibration theory

The Atlas Scientific EZO™ class pH circuit, has a flexible calibration protocol, allowing for single point, two point, or three point calibration.

The first calibration point must be a pH 7
This is known as the calibration midpoint. This is also the only calibration point used in a single point calibration.

The other two points can be any value, but they must be on opposite sides of the pH scale. These two points are known as the low calibration point and the high calibration point.



Using a commercially available pH 7 calibration solution that is not exactly pH 7.00 as pH 7.01 is not an issue.

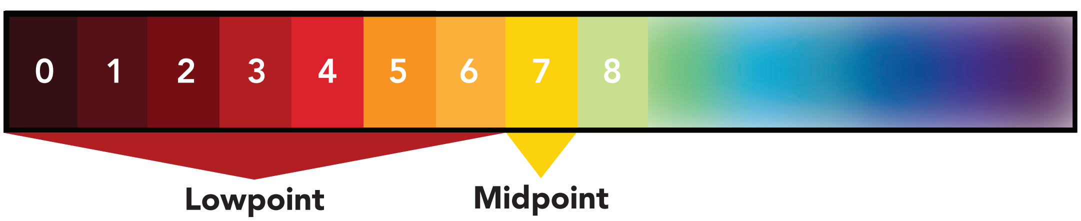
Because the pH of calibration solutions change when they are not at standard temperature (25˚C), any pH value can be entered in as the pH 7.00 value.

Generally speaking it is not advised to set the pH 7.00 calibration (the calibration midpoint) to a different value. This should only be used if the temperature of the water to be measured will continuously be very cold (< 10˚C), or very hot (> 45˚C).

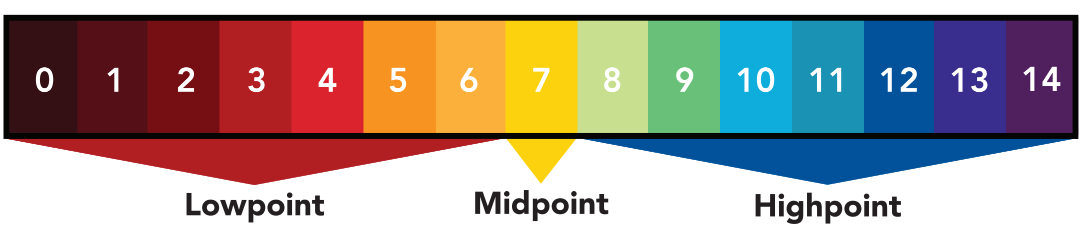
No calibration

One point calibration

Two point calibration will provide high accuracy between pH 7 and the second point calibrated against, such as a pH 4.



Three point calibration will provide high accuracy over the full pH range. 3 point calibration at pH 4, 7 and 10 should be considered the standard.



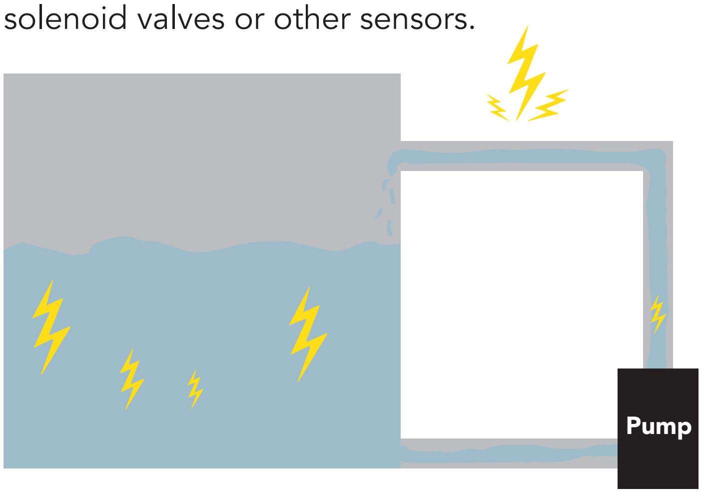
Only the calibration at pH 7 is mandatory. The other calibration points can be any value. The further apart these values are, the greater the accuracy.

Design considerations

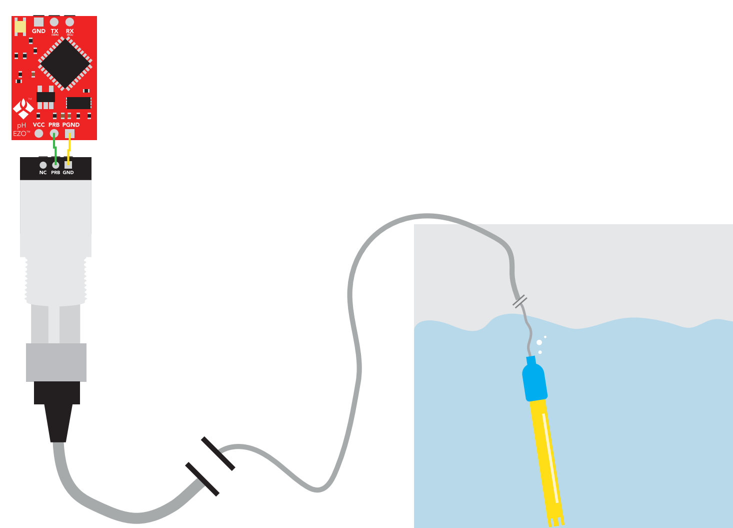
The Atlas Scientific EZO™ pH circuit is a micro-computer system that is specifically designed to be embedded into a larger system. The EZO™ pH circuit is not a completed product. The embedded systems engineer is responsible for building a completed working product.

Power and data isolation

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



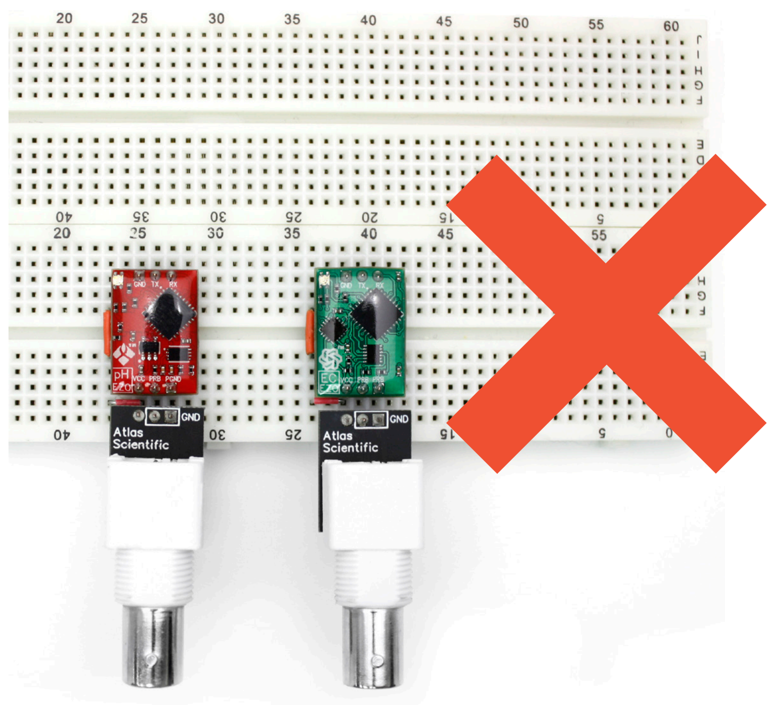
When electrical noise is interfering with the pH readings it is common to see rapidly fluctuating readings or readings that are consistently off. To verify that electrical noise is causing inaccurate readings place the pH probe in a cup of water by itself. The readings should stabilize quickly, confirming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



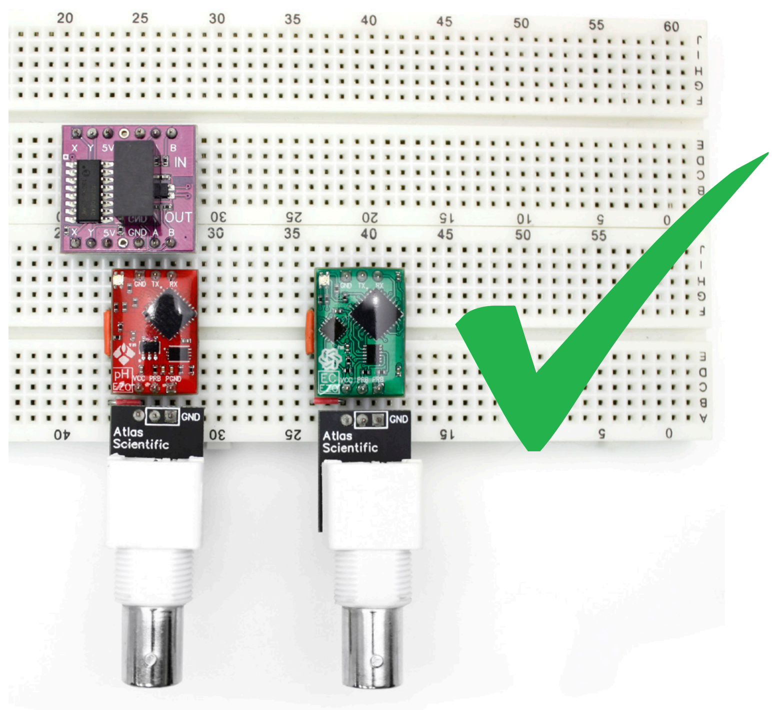
When reading pH and Conductivity together, it is strongly recommended that the EZO™ pH circuit is electrically isolated from the EZO™ Conductivity circuit.

Without isolation, Conductivity readings will effect pH accuracy.

Incorrect



Correct



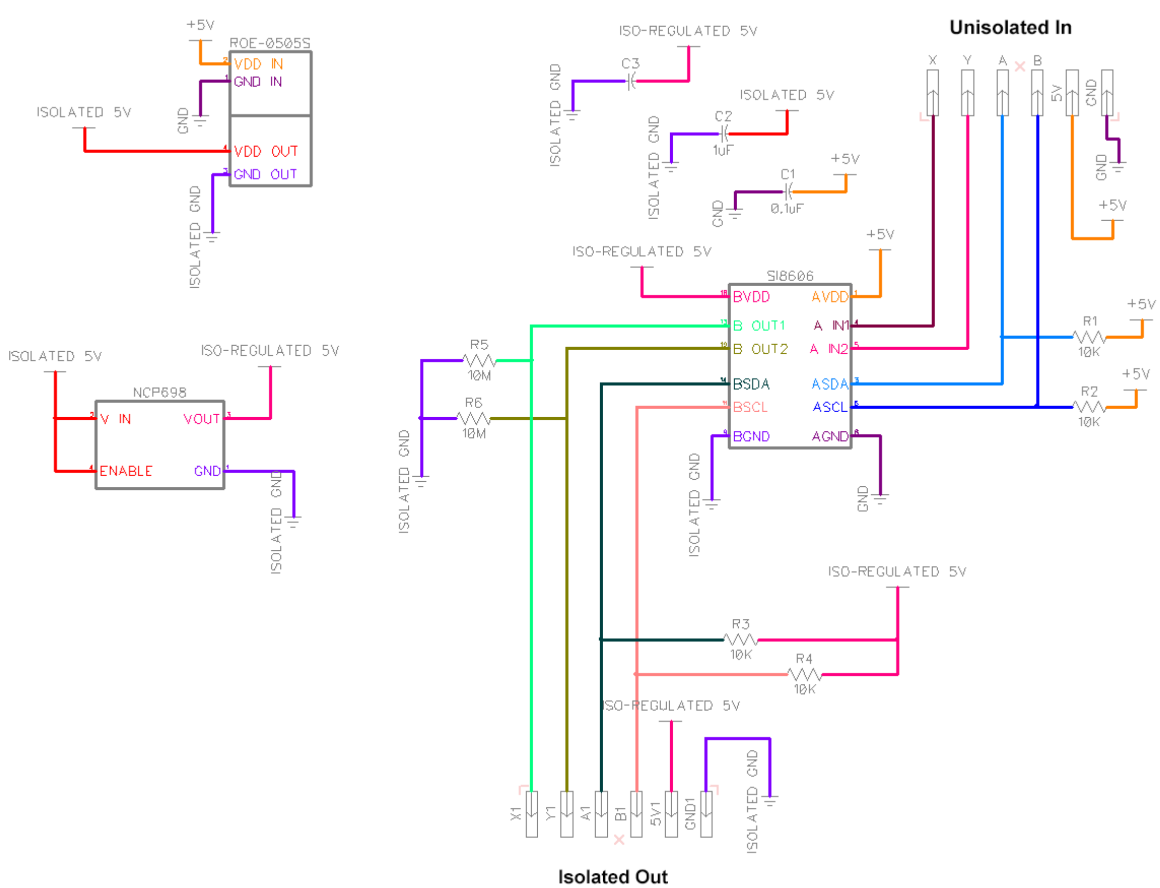
To correct this problem the power and data lines need to be electrically isolated. There is no one single method of doing this. This is just one of many ways to do so.

This schematic shows exactly how we isolate data and power using two parts, the data isolator part # SI8606, and the Power isolator part # ROE-0505S.

The SI8606 is a digital isolator with two bidirectional channels, which makes it excellent for use with I2C and UART protocols. This Part requires isolated power and pull ups on both channels on the isolated and non-isolated inputs. Pull up resistors can be anything from 3k to 10k.

The ROE-0505s is an isolated DC/DC converter that can handle 5V @ 1W. This part uses a Transformer that provides a 1:1 ratio (5V in and 5v out) however we have seen that 5V in produces 5.4V out and we recommend using a 5V regulator on its output. We use part# NCP698SQ50T1G.

Note: the Isolated Ground is different from the non-isolated Ground, these two lines should not be connected together.

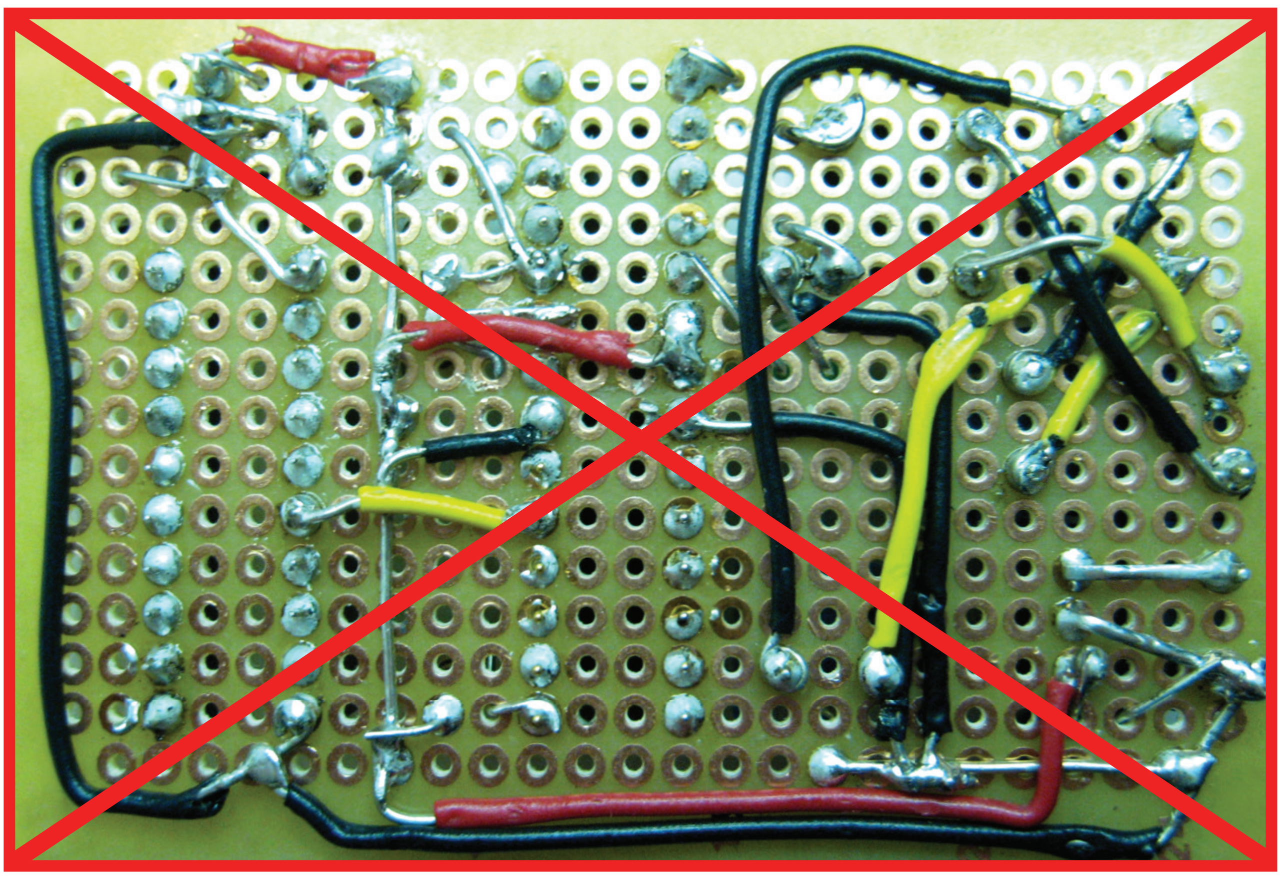


Board mounting

The Atlas Scientific EZO™ pH circuit should be tested in a **bread board** with different colored jumper wires connecting to each pin of the EZO™ pH circuit.

The EZO™ pH circuit should not have wires for other devices in your system laying on top of it. If long term use is desired a PCB should be made to hold the device.

Perfboards *(sometimes called Protoboards)* should never be used.



Micro-shorts and bleeding voltages are very common when using such boards. Achieving stable reading can be quite difficult or impossible.

Perfboards *(sometimes called Protoboards)* will void your devices warranty. No support will be given.

Known Issues

1. After powering up the EZO™ class pH circuit the first few readings (~2-10) will not show the correct pH. It takes a few readings for the device so stabilize after power up.

2. Turning off the EZO™ class pH circuit by sinking the VCC pin to ground will cause the pH readings to take ~1 to 5 minutes to return to normal after the VCC pin has been reconnected to the supply voltage. Because the data lines (RX/TX or SDA/SCL) are inactive high sinking the VCC pin to ground will cause the voltage from the two data lines to run backwards through the EZO™ class pH circuit. This will turn off the EZO™ class pH circuit but it will not stop power flowing. To properly shut down the pH circuit for long term hibernation cut the ground line.

3. After powering up the EZO™ class pH circuit when it is in UART mode the first command sent to it will comeback as an error. This is because the UART buffer will show that it has received a character during power up. Simply send a blank character to the pH circuit after it is powered up, this will clear the buffer.

UART Mode

™

UART mode command quick reference

There are a total of 15 different commands that can be given to the EZO™ class pH circuit.

All commands are ASCII strings or single ASCII characters

Command	Function	Default state
C	Enable / Disable or Query continuous readings (pg.25)	Enabled
Cal	Performs calibration (pg.28)	User must calibrate
Factory	Factory reset (pg.39)	N/A
I	Device information (pg.34)	N/A
I2C	Sets the I^2C ID number (pg.40)	Not set
L	Enable / Disable or Query the LEDs (pg.24)	LEDs Enabled
Name	Set or Query the name of the device (pg.34)	Not set
Plock	Enables / Disables the protocol lock feature (pg.42)	Disabled
R	Returns a single reading (pg.26)	N/A
Response	Enable / Disable or Query response code (pg.35)	Enabled
Serial	Set the baud rate (pg.38)	9600
Sleep	Enter low power sleep mode (pg.37)	N/A
Slope	Returns the slope of the pH probe (pg.33)	N/A
Status	Retrieve status information (pg.36)	N/A
T	Set or Query the temperature compensation (pg.27)	25˚C

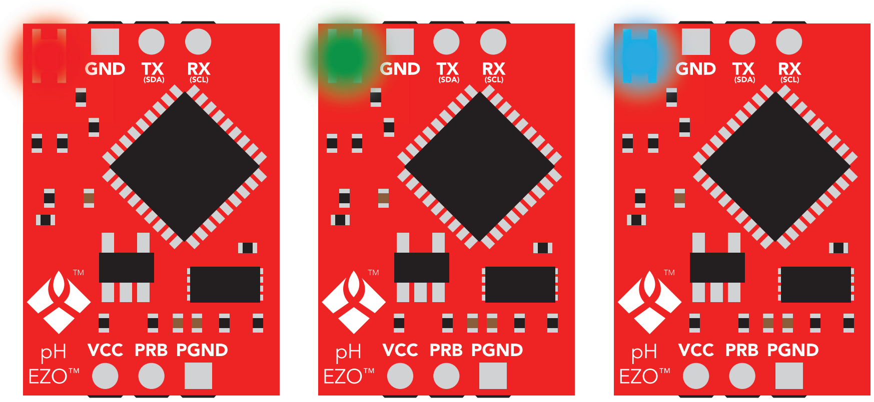
UART command definitions

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

LED control

All EZO™ class circuits have a tri-color LED, used to indicate device operation.

UART mode LED color definitions:
Steady **Green** = Power on/ standby
Red double blink = Command received and not understood
Green blink =Data transmission sent
Cyan = taking a reading



Command syntax

L,1<CR> LED enable
L,0<CR> LED disable
L,?<CR> Query the LED

Device response

L,1 <CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
The Led will be enabled and the green power on/ standby LED will turn on

L,0 <CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
The Led will be disabled

L,? <CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
?L,1<CR> if the LED is enabled
?L,0<CR> if the LED is disabled

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Continuous reading mode

All EZO™ class circuits are capable of continuous mode operation. In continuous mode, the device will output its readings, one after the other continuously until the continuous mode disable command has been issued. All EZO™ class circuits are defaulted to operate in continuous mode. If the LEDs are enabled, each time a data transmission occurs, the green LED will blink.

Command syntax

C,1<CR> Continuous mode enable
C,0<CR> Continuous mode disable
C,?<CR> Query continuous mode

Device response

C,1 <CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
The EZO™ class pH circuit, will output a numeric string containing the pH once per second

pH<CR> (1 second)
pH<CR> (2 seconds)
pH<CR> (n* seconds)

C,0 <CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
Continuous data transmission will cease.

C,? <CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
?C,1<CR> if continuous mode is enabled.
?C,0<CR> if continuous mode is disabled.

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Single reading mode

All EZO™ class circuits are capable of taking a single reading upon request. If the LEDs are enabled, each time a data transmission occurs, the green LED will blink.

Command syntax

R<CR> Returns a single reading

Device response

(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
The EZO™ class pH circuit, will output a single string containing a pH reading 1 second after the command was issued.

pH<CR> (1 second)

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Temperature compensation

In order to achieve the most accurate possible readings, the temperature of the liquid being measured must be transmitted to the EZO™ class pH circuit. The embedded systems engineer must keep in mind that the EZO™ class pH circuit, cannot read the temperature from a pH probe or from a temperature probe. Another device must be used to read the temperature. EZO™ class pH circuit, has its default temperature set at 25°C. The temperature at which to compensate against can be changed at any time using the "T" command.

Temperature is always in Celsius

The temperature compensation data is **NOT** stored to EEPROM. When the power to the EZO-pH circuit is cut, the temperature compensation value will return to 25°C.

Command syntax

(Using an example temperature 19.5)

T,19.5<CR>	Where the temperature is any value; floating point or int, in ASCII form
T,?<CR>	Query the set temperature

Device response

T,19.5<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
There is no other output associated output with this command

T,?<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
?T,19.5 <CR>

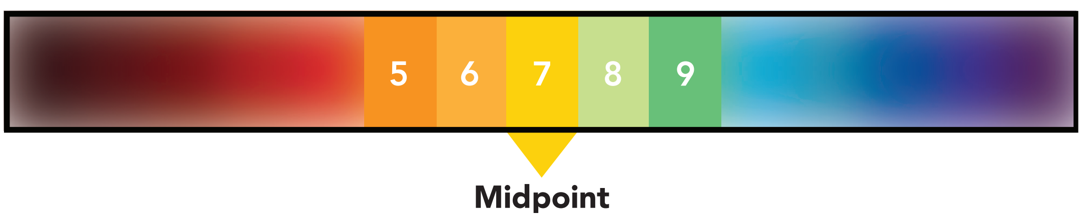
<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Calibration

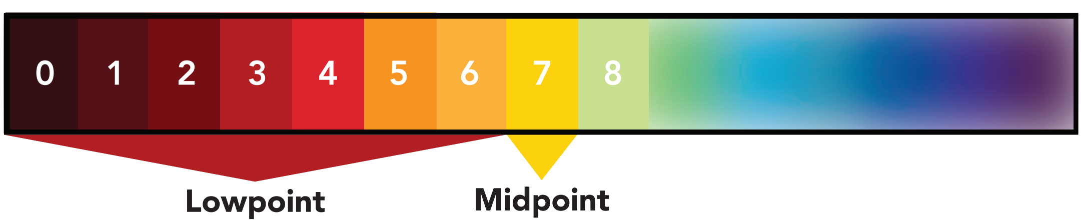
The Atlas Scientific EZO™ class pH circuit, has a flexible calibration protocol, allowing for single point, two point, or three point calibration.

During calibration, it is required that ***pH 7 calibration be done first***. Calibration can be done at a maximum of 3 points. These three points are known as the low calibration point, the middle calibration point and the high calibration point. Where pH 7.XX must be the first calibration point. This is known as the middle calibration point.

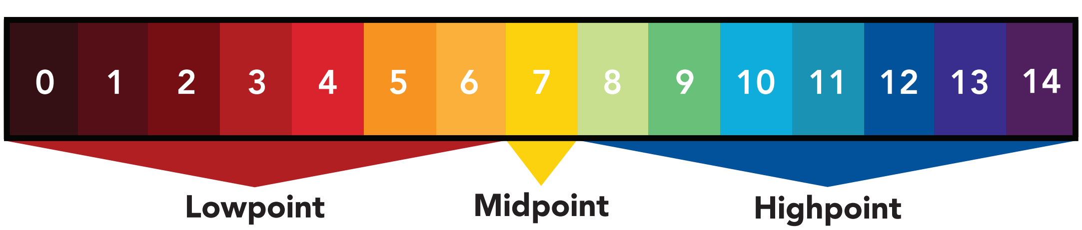
One point calibration



Two point calibration will provide high accuracy between pH 7 and the second point calibrated against, such as a pH 4.



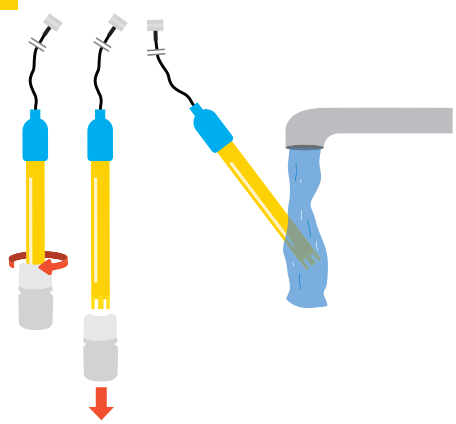
Three point calibration will provide high accuracy over the full pH range. 3 point calibration at pH 4, 7 and 10 should be considered the standard.



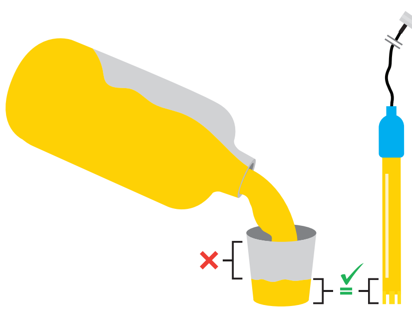
<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Midpoint calibration

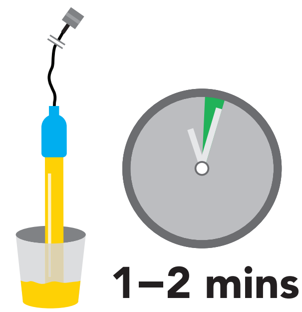
Step 1



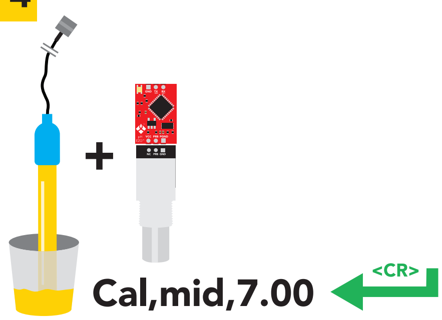
Step 2



Step 3



Step 4

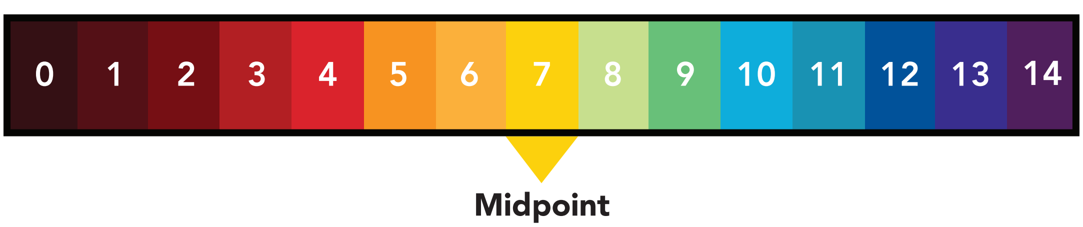


Step 5

Command syntax

Cal,mid,X.XX<CR> Where X.XX is any floating point value that represents the pH midpoint.

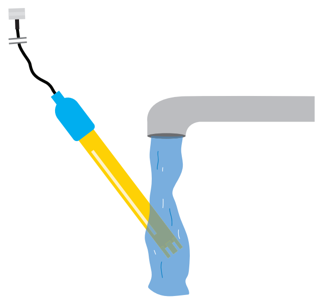
In most cases this should be 7.00



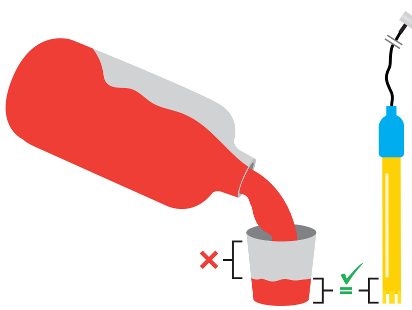
<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Lowpoint calibration

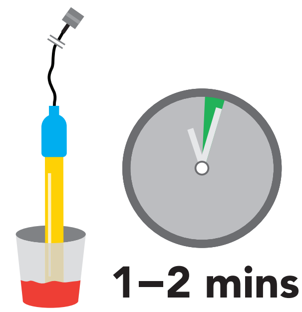
Step 1



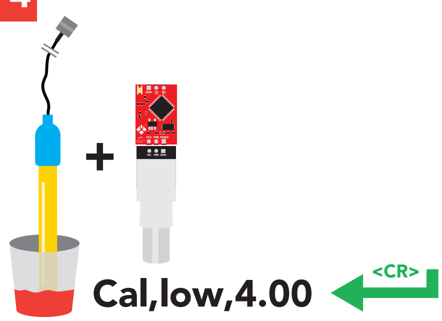
Step 2



Step 3



Step 4



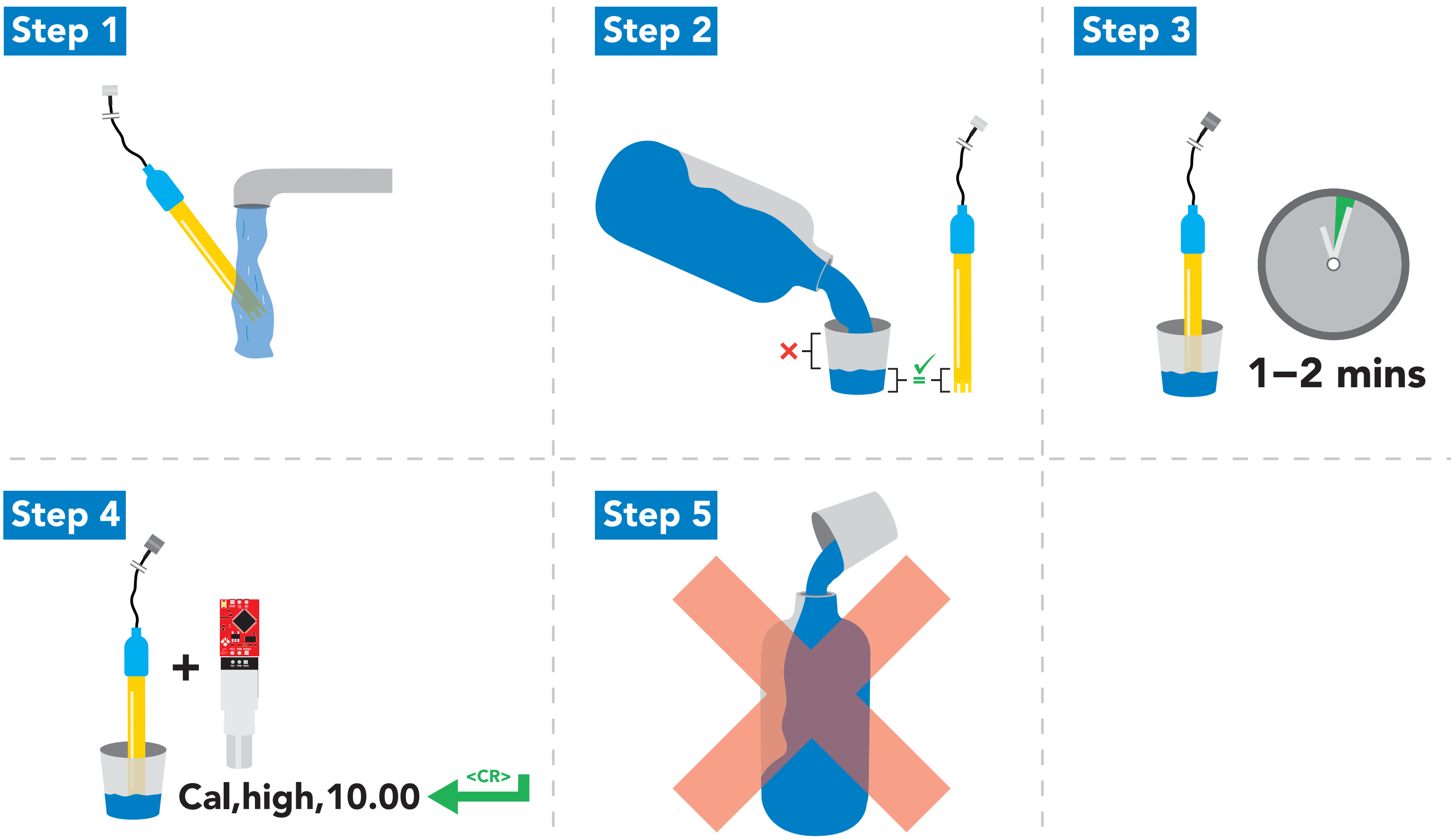
Step 5

Command syntax

Cal,low,X.XX<CR> Where X.XX is any floating point value that represents a low calibration point. (pH 1 to pH 6) In most cases this should be 4.00

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Highpoint calibration



Command syntax

Cal,high,XX.XX<CR> Where XX.XX is any floating point value that represents a high calibration point. (pH 8 to pH 14) In most cases this should be 10.00

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Command syntax

Cal,?<CR>	Query the calibration
Cal,clear<CR>	Clears all calibration data

Device response

Cal,clear<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
There is no other output associated output with this command.

Cal,mid,X.XX<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
The LED will turn **Cyan** during the calibration.

Cal,low,X.XX<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
The LED will turn **Cyan** during the calibration.

Cal,high,XX.XX<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
The LED will turn **Cyan** during the calibration.

Cal,?<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")

If not calibrated:	?CAL,0
If single point calibration:	?CAL,1
If two point calibration:	?CAL,2
If three point calibration:	?CAL,3

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

Issuing a cal,low or cal,high command can be done at any time and will have no effect on other previously set calibration points.

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Slope – Advanced feature

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

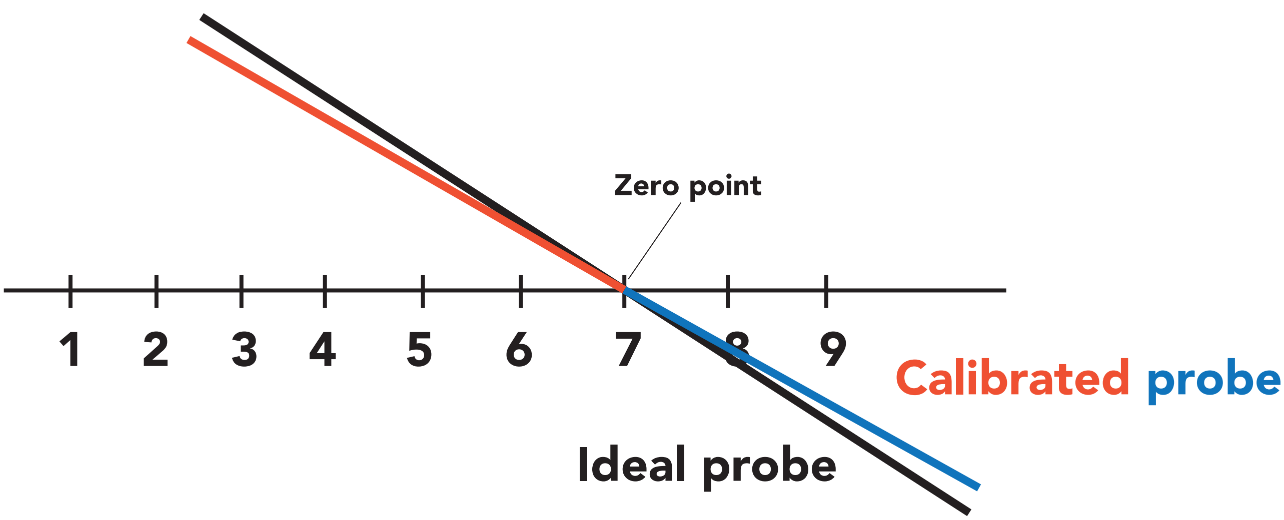
Command syntax

SLOPE,?<CR> Returns the slope of the pH probe

Device response

?SLOPE,99.7,100.3
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")

Where, 99.7% is how closely the slope of the acid calibration line matched the "ideal" pH probe. And, 100.3% is how closely the slope of the base calibration matches the "ideal"pH probe.



<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Device identification

All EZO™ class circuits are capable of being assigned a name. This is a simple way to identify the device in a system that consists of multiple EZO™ class circuits. A name can consist of any combination of ASCII characters, with a length of 1 to 16 characters long, ***no blank spaces***.

Command syntax

NAME,nnn<CR> Sets the device name, where nnn is the given name.
NAME,?<CR> Query the device name

Device response

NAME,DEVICE_1<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
There is no other output associated output with this command.

NAME,?<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
?NAME, DEVICE_1<CR>

Device information

The EZO™ class circuit can identify itself by device type and firmware version.
This is done by transmitting the "I" command.

Command syntax

I<CR> Device information

Device response

?I,pH,1.0<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")

Where pH = device type
1.0 = firmware version number

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Response codes

The Atlas Scientific EZO™ class circuits, have 7 response codes to help the user understand how the device is operating, and to aid in the construction of a state machine to control the EZO™ class circuit. All EZO™ class devices indicate a response code has been triggered, by transmitting a string with the prefix "*" and ending with a carriage return <CR>.

A list of response codes

***ER**	An unknown command has been sent
***OV**	The circuit is being overvolted (VCC>=5.5V)
***UV**	The circuit is being undervolted (VCC<=3.1V)
***RS**	The circuit has reset
***RE**	The circuit has completed boot up
***SL**	The circuit has been put to sleep
***WA**	The circuit has woken up from sleep

Only the response code "*OK" can be disabled.
Disabling this response code is done using the "response" command.

Command syntax

RESPONSE,1<CR>	Enable response code (default)
RESPONSE,0<CR>	Disable response code
RESPONSE,?<CR>	Query the response code

Device response

RESPONSE,1<CR>
EZO™ class circuit will respond "*OK<CR>"

RESPONSE,0<CR>
There is no response to this command

RESPONSE,?<CR>

?RESPONSE,1<CR> If the response code is enabled
?RESPONSE,0<CR> If the response code is disabled

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Reading the status of the device

The Atlas Scientific™ EZO™ class circuit, is able to report its voltage at the VCC pin and reason the device was last restarted.

Restart codes

P power on reset
S software reset
B brown out reset
W watchdog reset
U unknown

Command syntax

STATUS<CR>

Device response

(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")

?STATUS,P,5.038<CR>

Where: P is the reason for the last reset event
Where: 5.038 is the its voltage at the VCC

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Low power state

To conserve energy in between readings, the Atlas Scientific™ EZO™ class circuit, can be put into a low power sleep state. This will turn off the LEDs and shut down almost all of the internal workings of the EZO™ class circuit. The power consumption will be reduced to 1.16 mA at 5V and 0.995 mA at 3.3V. **To wake the EZO™ class circuit, send it any character.**

After the device is woken up, **4** consecutive readings should be taken before the readings are considered valid.

Command syntax

SLEEP<CR> Enter low power sleep state

Device response

(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
*SL<CR>

Device response to wake up:
*WA<CR>

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Change baud rate

The Atlas Scientific EZO™ class circuit, has 8 possible baud rates it can operate at.
The default baud rate is

9600 bps
8 data bits
1 stop bit
no parity
no flow control

Baud rate changes will be retained even if power is cut.

Data bits, stop bits, parity and flow control are fixed and cannot be changed.

1. 300 bps
2. 1200 bps
3. 2400 bps
4. 9600 bps
5. 19200 bps
6. 38400 bps
7. 57600 bps
8. 115200 bps

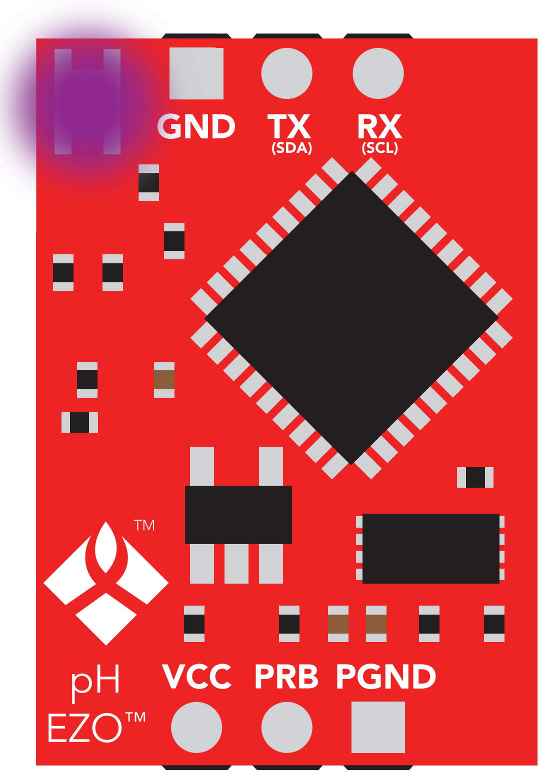
Command syntax

(Using an example baud rate of 38400)
SERIAL,38400<CR>

Device response

(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>") The EZO™ class circuit will respond with a **Purple** LED double blink. The EZO™ class circuit will then restart at the new baud rate.

The LED blink will happen even if the LEDs are disabled.



<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Factory reset

All EZO™ class circuits, are capable of resetting themselves to the original factory settings. Issuing a factory reset will:

Reset the calibration back to factory default
Reset default temperature to 25°C
Set debugging LED to on.
Enable response codes

This command will not change the set baud rate.

Command syntax

Factory<CR> Factory reset

Device response

(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
The EZO™ class circuit, will respond: *RE<CR>

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Switch from UART mode to I^2C mode

Transmitting the command I^2C,[n] will set the EZO™ class circuit into I^2C mode from UART mode. **Where [n] represents any number from 1-127.** The I^2C address is sent in decimal ASCII form. Do not send the address in hexadecimal ASCII form.

Command syntax

(Using as example an I^2C ID number of 99)
I2C,99<CR>

Device response

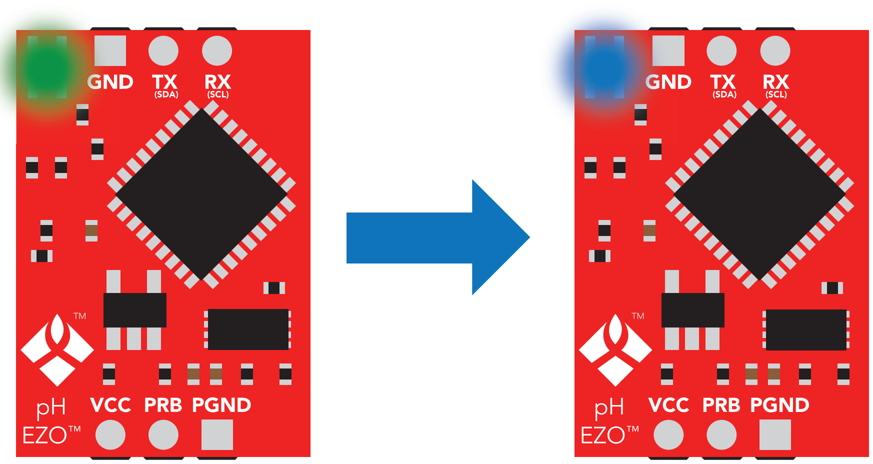
If an address > 127 is given

*ER Indicating an error has occurred

If an address >0 and <128 is given
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")

*RS<CR> The device will restart in I^2C mode

The **Green** LED used to indicate that the device is powered and awaiting an instruction will now change to **Blue**.

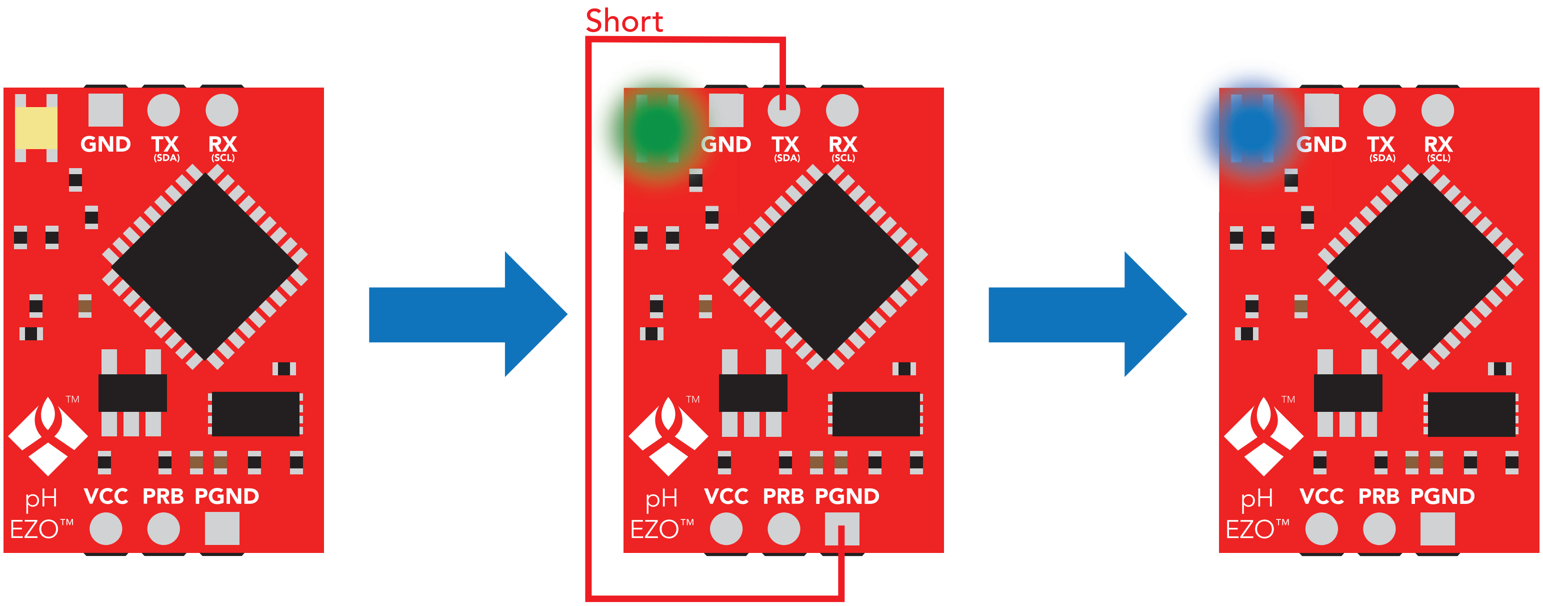


<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Manual switching to I²C mode

All EZO™ class circuits can be manually switched from UART mode, to I²C mode. **If this is done the EZO™ class pH circuit, will set its I²C address to 99 (0x63).**

1. Cut the power to the device
2. Disconnect any jumper wires going from TX and RX to the master micro controller
3. Short the PGND pin to the TX pin
4. Power the device
5. Wait for LED to change from **Green** to **Blue**
6. Remove the short from the probe pin to the TX pin
7. Power cycle the device
8. The device is now I²C mode



<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Protocol lock

This feature is available on All EZO™ class circuits running firmware version 1.95 or higher.

When the protocol lock feature is enabled all changes to the communication protocol are blocked. This means that whatever communication mode the device is in (I^{2}C or UART); that communication protocol cannot be changed by any means. Furthermore, changes to the devices baud rate or I^{2}C address also cannot be changed.

By default the protocol lock is: **DISABLED**
Changes to this setting are retained even if the power is cut.

Command syntax

PLOCK,1<CR>	Enables the protocol lock feature
PLOCK,0<CR>	Disables the protocol lock feature
PLOCK,?<CR>	Query the state of the lock

Device response

PLOCK,1<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
There is no other output associated output with this command.

PLOCK,0<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")
There is no other output associated output with this command.

PLOCK,?<CR>
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")

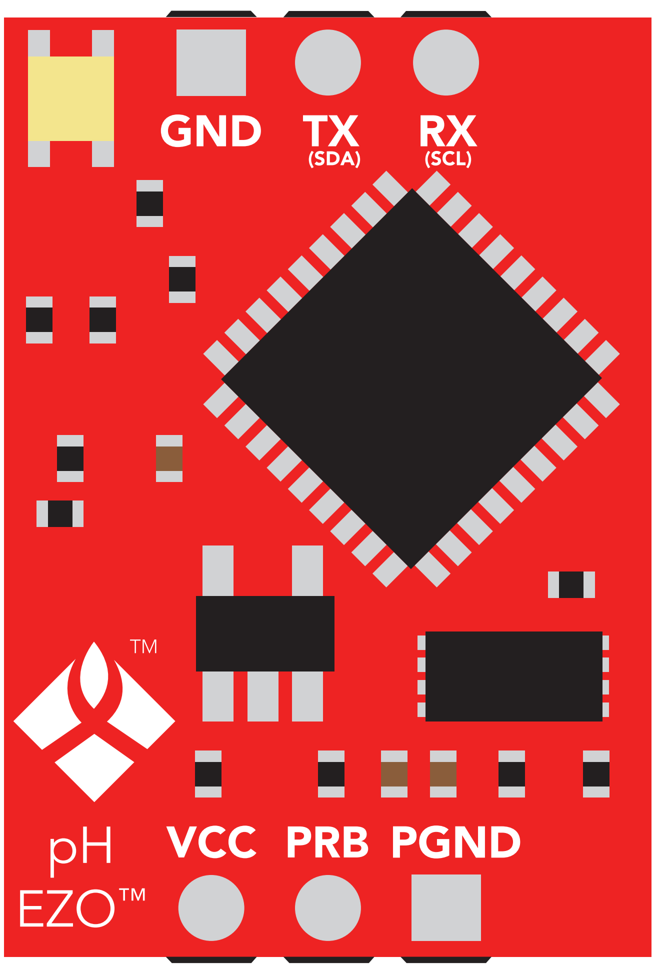
?PLOCK,1<CR>	if the lock in enabled
?PLOCK,0<CR>	if the lock in disabled

<CR> represents a carriage return (ASCII 13). The user does not transmit the literal string "<CR>". Commands are not case sensitive.

Once the protocol lock is enable the following commands / actions will no longer work

Manual switching to I²C or UART mode
Software switching from I²C or UART mode
UART baud rate change
I²C address change

If the protocol lock is enabled attempting to manually switch to UART or I²C mode will have no effect. If any of the software commands are issued to switch mode, baud rate or I²C address the EZO class device will return *ER<CR>



Atlas Scientific™
Environmental Robotics

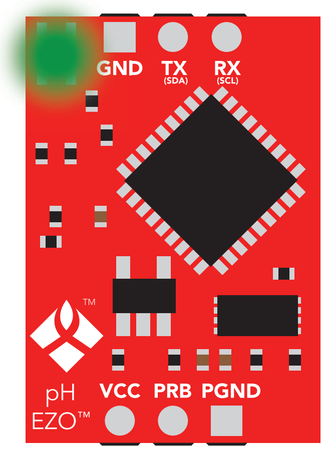
pH Circuit
EZO™

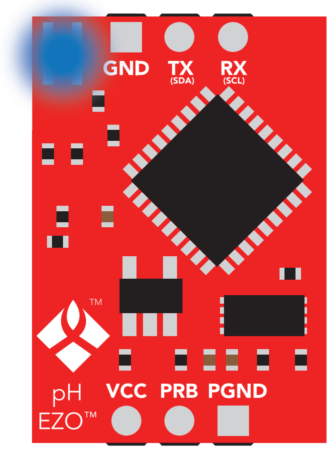
I²C Mode

I²C mode

An I²C address can be any number from 1-127. If the EZO™ Class pH circuit was put into I²C mode by jumping PRB to TX, the I²C address is 99(0x63).

Once an EZO™ class device has been put into I²C mode the **Green** power LED that was used in UART mode will now switch to a **Blue** LED. This indicates the device is now in I²C mode.





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

Communication to the EZO™ class device is controlled by the master. The EZO™ class device as an I²C slave. The slave device is not able to initiate any data transmissions.

An I²C write event is defined as such

start | write command to device address | instruction | stop

In order to get the response from device, it is necessary to initiate a read command. The I²C protocol does not permit the slave device to initiate any data transmissions.

An I²C read event is defined as such

start | read command to device address | data byte | data byte | data byte | stop

Data from a read back event

The first byte of the data read back, is the response code. This byte informs the master of the status of the data about to be read back. For all commands, the first byte of the read data is the response code, which is defined as

Value	Meaning
255	**No Data** – there is no pending request, so there is no data to return from the circuit
254	**Pending** – the request is still being processed. Ensure that you have waited the minimum time to guarantee a response
2	**Failed** – the request failed
1	**Success** – the requested information is ready for transmission. There may be more bytes following this which are returned data

The bytes transmitted after that, will be the requested data. When all the data has been transmitted each additional byte will be a NULL.

Example

A read request when no command has been given.

255	null	(every byte read after the first byte will be NULL)	stop

All I²C mode responses are in ASCII format however, they do not terminate with a <CR> rather, they terminate with a NULL. The Null termination makes data manipulation easier once it has been received.

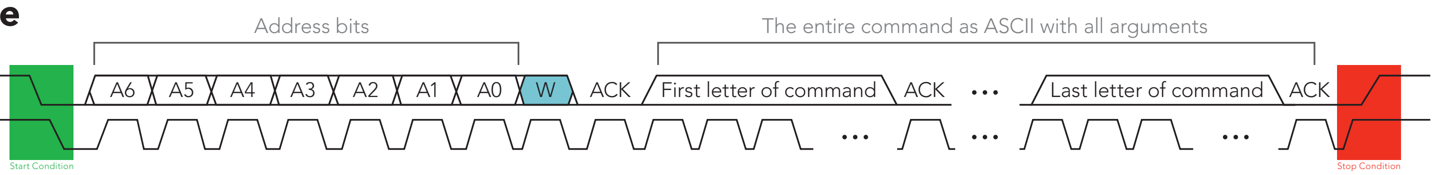
Example
EZO™ class device responds to a request for a reading

12.34 ≠ float	12.34 =byte[7]	
	Byte[0]= 1	(decimal 1)
	byte[1]= "1"	(ASCII 49)
	byte[2]= "2"	(ASCII 50)
	byte[3]= "."	(ASCII 46)
	byte[4]= "3"	(ASCII 51)
	byte[5]= "4"	(ASCII 52)
	byte[6]= NULL	(ASCII 0)

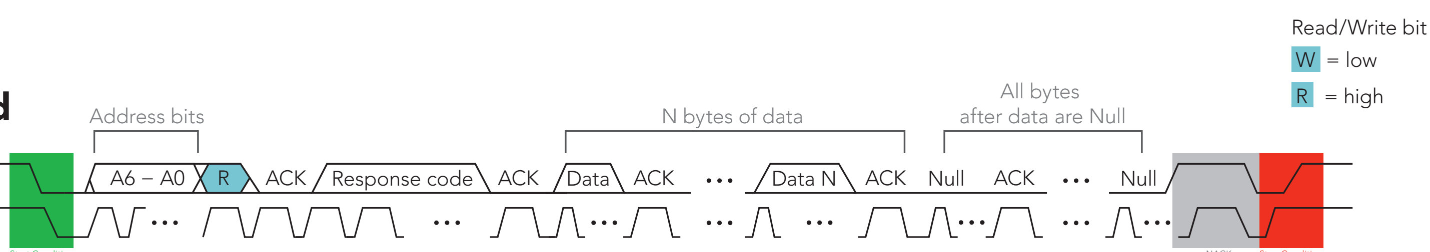
I²C timing

When a command is issued to the EZO™ class device, a certain amount of time must be allowed to pass before the data is ready to be read. Each command specifies the delay needed before the data can be read back. EZO™ class devices do not support I²C clock stretching. All commands are sent to the EZO™ class device in the same ASCII format as in UART mode however, there is no <CR> sent at the end of the transmission.

Write



Read



I²C command quick reference

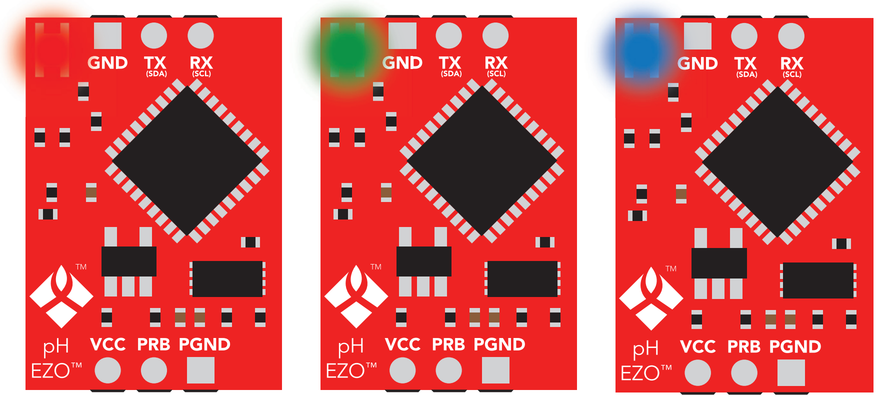
There are a total of 11 different commands that can be given to the EZO™ class pH circuit.

Command	Function
Cal	Performs calibration (pg.52)
Factory	Factory reset (pg.64)
I	Device information (pg.59)
L	Enable / Disable or Query the LEDs (pg.49)
Plock	Enables / Disables the protocol lock feature (pg.66)
R	Returns a single reading (pg.50)
Serial	Switch back to UART mode (pg.63)
Sleep	Enter low power sleep mode (pg.62)
Slope	Returns the slope of the pH probe (pg.58)
Status	Retrieve status information (pg.60)
T	Set or Query the temperature compensation (pg.51)

I²C LED control

All EZO™ class circuits have a tri-color LED used to indicate device operation.

I²C mode LED color definitions:
Steady **Blue** = Power on/ standby
Red double blink = Command received and not understood
Blue blink = Data transmission sent
Green = taking a reading



Command syntax

L,1	LED enable
L,0	LED disable
L,?	Query the LED

Device response

L,1
The Led will be enabled and the blue power on/ standby LED turn on.
After 300ms, an I²C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

1	null

L,0
The Led will be disabled
After 300ms, an I²C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

1	null

L,?
After 300ms, an I²C read command can be issued to get the response code.

1	?	L	,	1	null	(?L,1) if the LED is enabled
1	?	L	,	0	null	(?L,0) if the LED is disabled

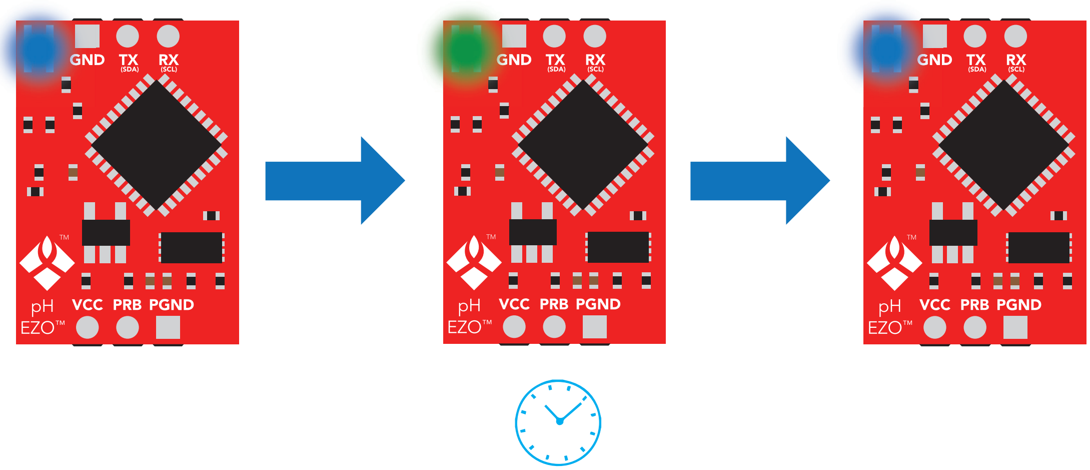
I²C take reading

When a reading is taken, the LED (if enabled) will turn **Green**, indicating that a reading is being taken. Once the reading has been taken, the LED will turn back to **Blue**.

Command syntax

R   Returns a single reading

Time until instruction is processed: 1 second



1 Second time has passed

Device response

After 1 second, an I²C read command can be issued to get the response:

| 1 | pH | null |

pH represents many bytes.

The string will be no longer than 7 bytes.

I²C Temperature compensation

In order to achieve the most accurate possible readings, the temperature of the liquid being measured must be transmitted to the EZO™ class pH circuit. The embedded systems engineer must keep in mind that the EZO™ class pH circuit, cannot read the temperature from a pH probe, or from a temperature probe. Another device must be used to read the temperature. EZO™ class pH circuit, has its default temperature set at 25˚C. The temperature, at which to compensate against, can be changed at any time using the "T" command.

Command syntax

(Using an example temperature 19.5)

T,19.5 Where the temperature is any value; floating point, or int, in ASCII form

T,? Query the set temperature

Time until instruction is processed: 300ms

Device response

T,19.5

After 300ms, an I²C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

1	null

T,?

After 300ms, an I²C read command can be issued to get the response

1	?	T	,	1	9	.	5	null

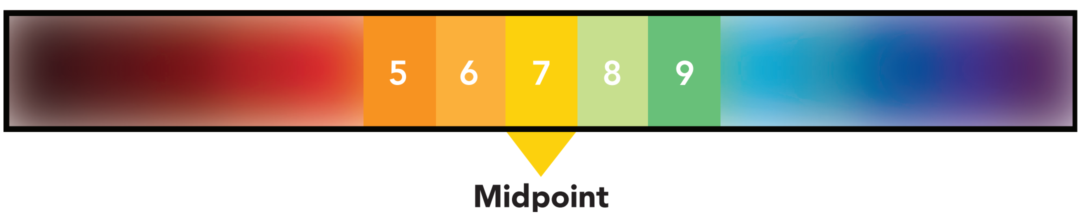
(?T,19.5)

I²C Calibration

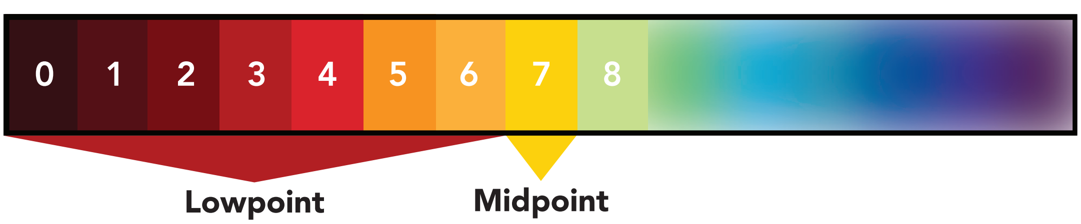
The Atlas Scientific EZO™ class pH circuit, has a flexible calibration protocol, allowing for single point, two point, or three point calibration.

During calibration, it is required that ***pH 7 calibration be done first***. Calibration can be done at a maximum of 3 points. These three points are known as the low calibration point, the middle calibration point and the high calibration point. Where pH 7.XX must be the first calibration point. This is known as the middle calibration point.

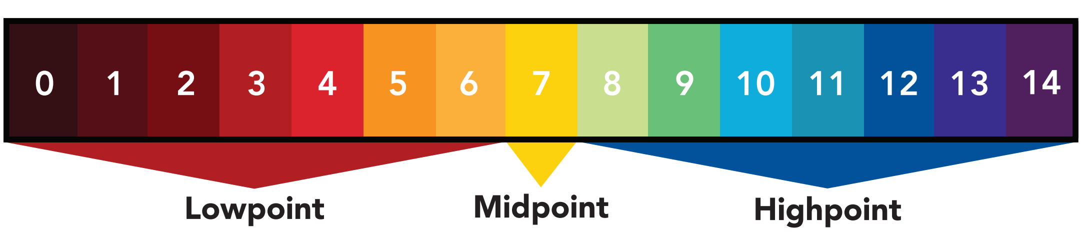
One point calibration



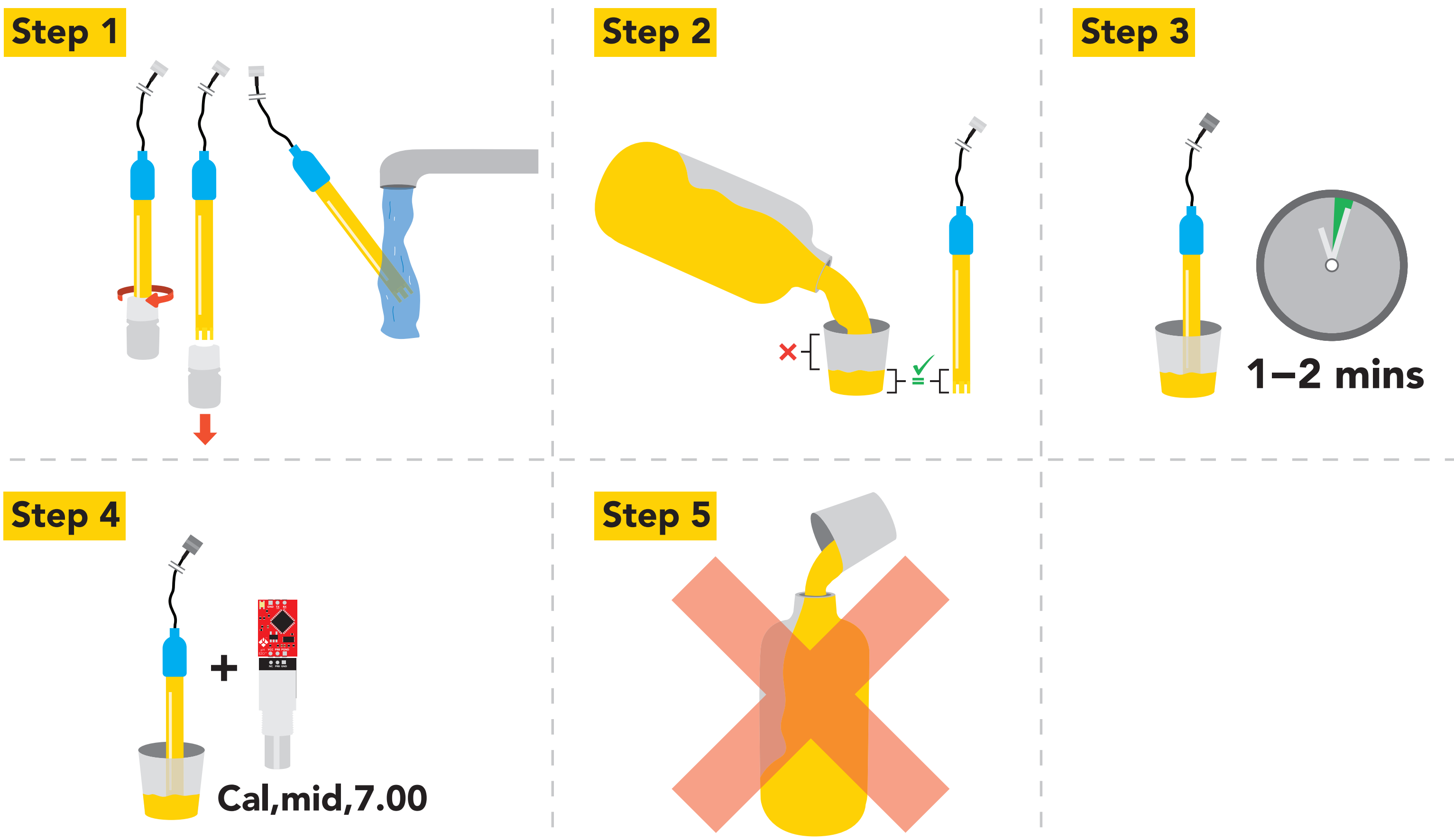
Two point calibration will provide high accuracy between pH 7 and the second point calibrated against, such as a pH 4.



Three point calibration will provide high accuracy over the full pH range. 3 point calibration at pH 4, 7 and 10 should be considered the standard.



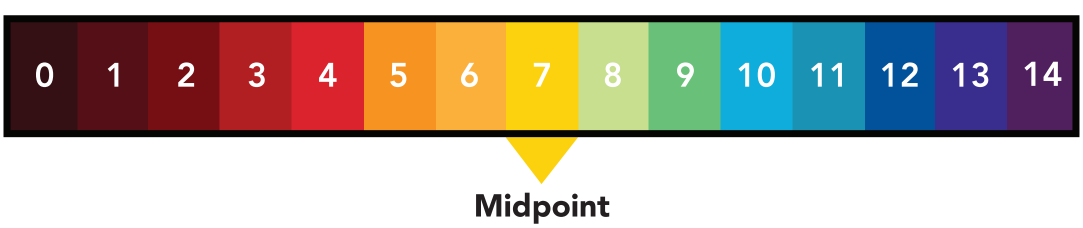
Midpoint calibration



Command syntax

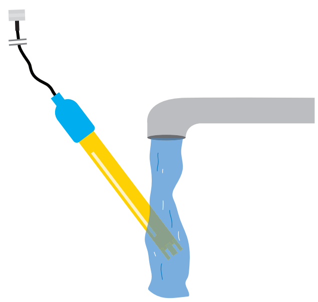
Cal,mid,X.XX Where X.XX is any floating point value that represents the pH midpoint.

In most cases this should be 7.00

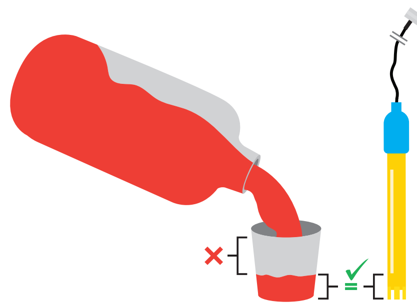


Lowpoint calibration

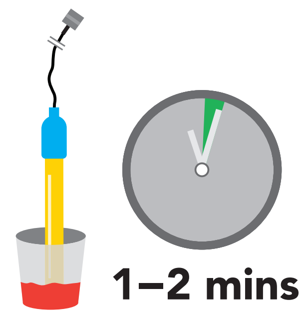
Step 1



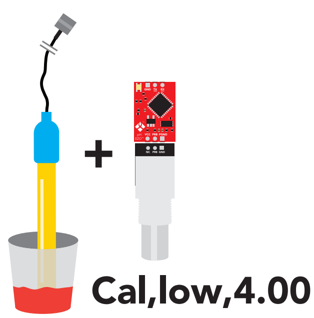
Step 2



Step 3



Step 4

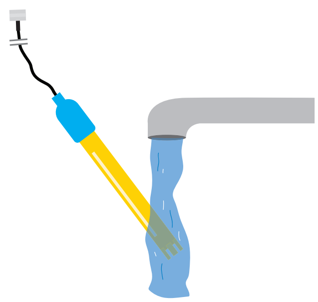


Step 5

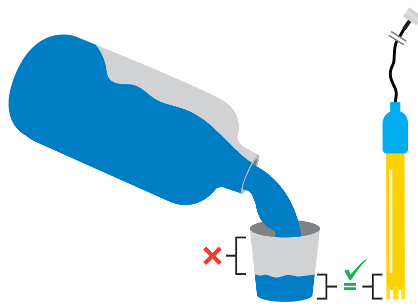
Command syntax
Cal,low,X.XX Where X.XX is any floating point value that represents a low calibration point. (pH 1 to pH 6) In most cases this should be 4.00

Highpoint calibration

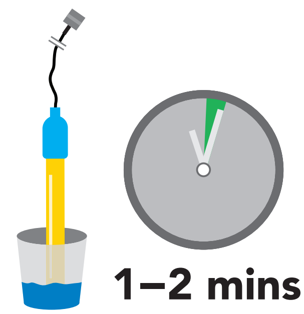
Step 1



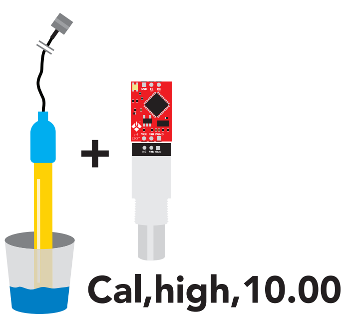
Step 2



Step 3



Step 4



Step 5

Command syntax

Cal,high,XX.XX Where XX.XX is any floating point value that represents a high calibration point. (pH 8 to pH 14) In most cases this should be 10.00

Command syntax

Cal,? Query the calibration
Cal,clear Clears all calibration data

Device response

Cal,clear
The LED will turn **Cyan** during the calibration.
After 300ms, an I²C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

Cal,mid,X.XX
The LED will turn **Cyan** during the calibration.
After 1.6 seconds, an I²C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

Cal,low, X.XX
The LED will turn **Cyan** during the calibration.
After 1.6 seconds, an I²C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

Cal,high, XX.XX
The LED will turn **Cyan** during the calibration.
After 1.6 seconds, an I²C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

Cal,?
After 300ms, an I²C read command can be issued to get the response code.

If not calibrated

?	C	A	L	,	0	null

(?CAL,0)

If single point calibration

?	C	A	L	,	1	null

(?CAL,1)

If two point calibration

?	C	A	L	,	2	null

(?CAL,2)

If three point calibration

?	C	A	L	,	3	null

(?CAL,3)

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

Issuing a cal,low or cal,high command can be done at any time and will have no effect on other previously set calibration points.

Slope – Advanced feature

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

Command syntax

SLOPE,? Returns the slope of the pH probe

Device response

?SLOPE,99.7,100.3
(If the response code is enabled, the EZO™ class circuit will respond "*OK<CR>")

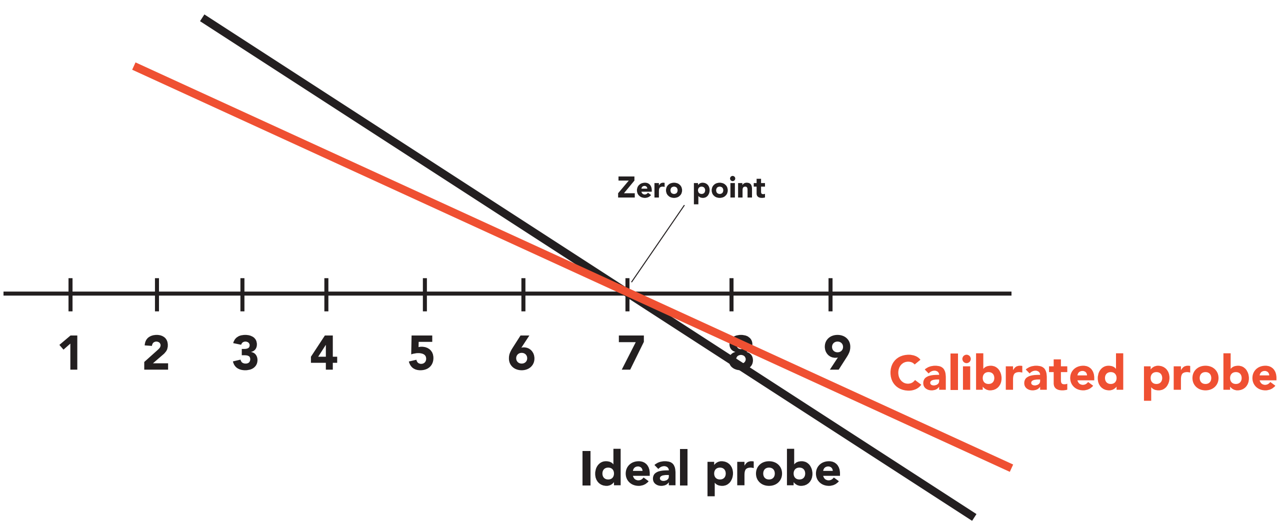
Where, 99.7 is the slope of the acid calibration
And, 100.3 is the slope of the base calibration

After 300ms, an I²C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

1	null

After 300ms, an I2C read command can be issued to get the response

1	?	S	L	O	P	E	,	9	9	.	7	,	1	0	0	.	3	null



I²C Device Info

The EZO™ class circuit, can identify itself by device type and firmware version.
This is done by transmitting the "I" command.

Command syntax
I Device information

Time until instruction is processed 300 ms

Device response:
After 300ms, an I²C read command can be issued to get the response:

?I,PH,1.0

Where pH = device type
1.0 = firmware version number

Reading the status of the device in I^2C mode

The Atlas Scientific™ EZO™ class circuit, is able to report its voltage at the VCC pin and the reason the device was last restarted.

Restart codes

P power on reset
S software reset
B brown out reset
W watchdog reset
U unknown

Command syntax

STATUS
Time until instruction is processed, 300ms

Device response

After 300ms, an I^2C read command can be issued, to get the response

?	S	T	A	T	U	S	,	P	,	5	.	0	3	8	null

?STATUS,P,5.038

Where: P is the reason for the last reset event
Where: 5.038 is the its voltage at the VCC

I^{2}C Address change

Transmitting the command I^{2}C,[n] while the EZO™ class circuit is already in I^{2}C mode will change the devices I^{2}C address. **Where [n] represents any number from 1-127.**

Warning!

After changing the I^{2}C address the EZO™ class circuit will no longer be able respond to any commands from the master device until its code has been updated with the new I^{2}C address

Command syntax

I2c,[n]

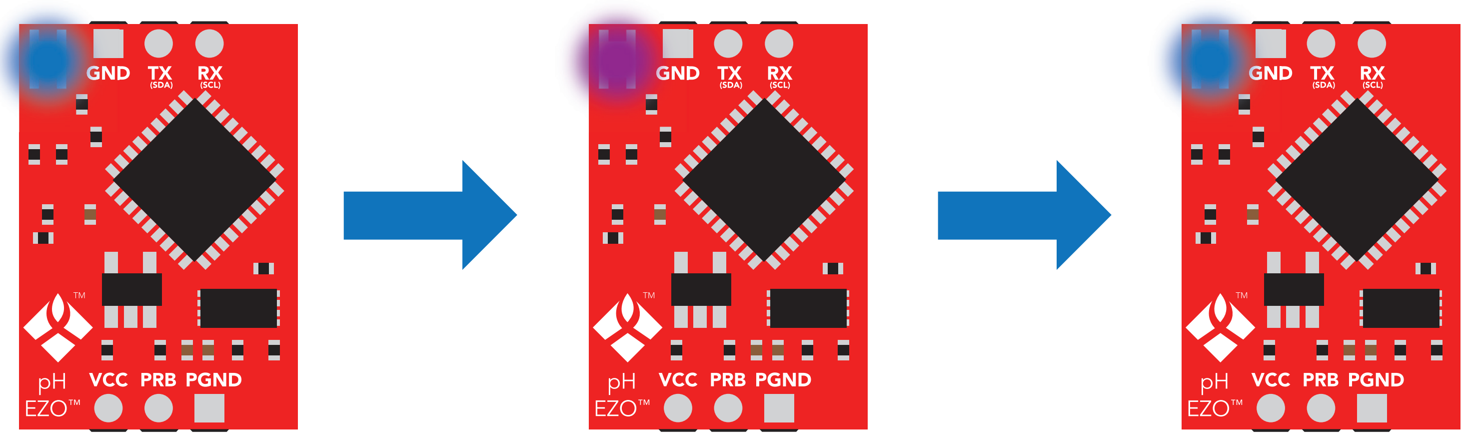
Device response

If an address >0 and <128 is given

If an address of 0 or and address of > 127 is given EZO™ class circuit will issue an error response and not change the I^{2}C address.

The blue led used to indicate the device is in I^{2}C mode will turn purple then the device will restart using its new I^{2}C address.

No calibration information will be lost by changing the I^{2}C address



I²C Low power state

To conserve energy in between readings, the Atlas Scientific pH EZO™ class circuit can be put into a low power sleep state. This will turn off the LEDs and shut down almost all of the internal workings of the EZO™ class circuit. The power consumption will be reduced to 1.16 mA at 5V and 0.995 mA at 3.3V. **To wake the EZO™ class circuit, send it any command.**

After the device is woken up, **4** consecutive readings should be taken before the readings are considered valid.

Command syntax

SLEEP Enter low power sleep state
Time until instruction is processed, 300ms

Device response

If the LEDs are enabled, the **Blue** LED will blink and then turn off.
There is no other output associated with this command.

Switch from I²C mode to UART mode

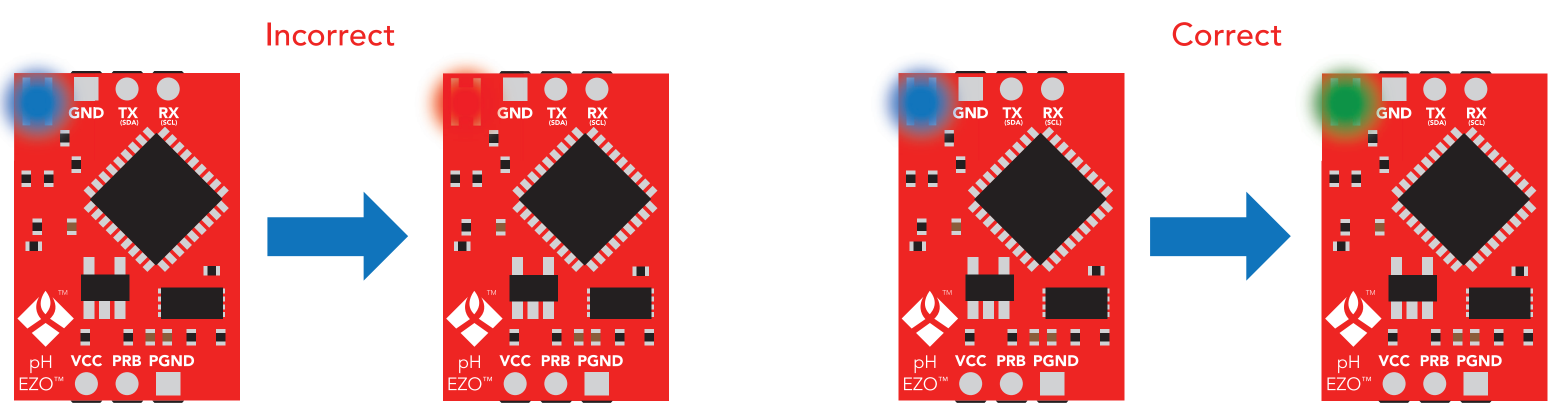
Transmitting the command serial,<n> will set the EZO™ class circuit into UART mode from I²C mode. **Where [n] represents any of one the 8 available baud rates.**

Command syntax
(Using as example a baud rate of 9600)

SERIAL,9600

Device response
If an incorrect baud rate is sent the device will not switch into UART mode and the **Red** LED will flash.

If a correct baud rate is given:
The **Blue** LED used to indicate that the device is powered and awaiting an instruction will now change to **Green**.



Factory reset

All EZO™ class circuits, are capable of resetting themselves to the original factory settings.
Issuing a factory reset will:

Reset the calibration back to factory default
Reset default temperature to 25°C
Set debugging LED to on.
Enable response codes

This command will not change the set I²C address

Command syntax

Factory   Factory reset

Device response

After 300ms the STATUS command can be issued to see that the device was reset.

?	S	T	A	T	U	S	,	S	,	5	.	0	3	8	null

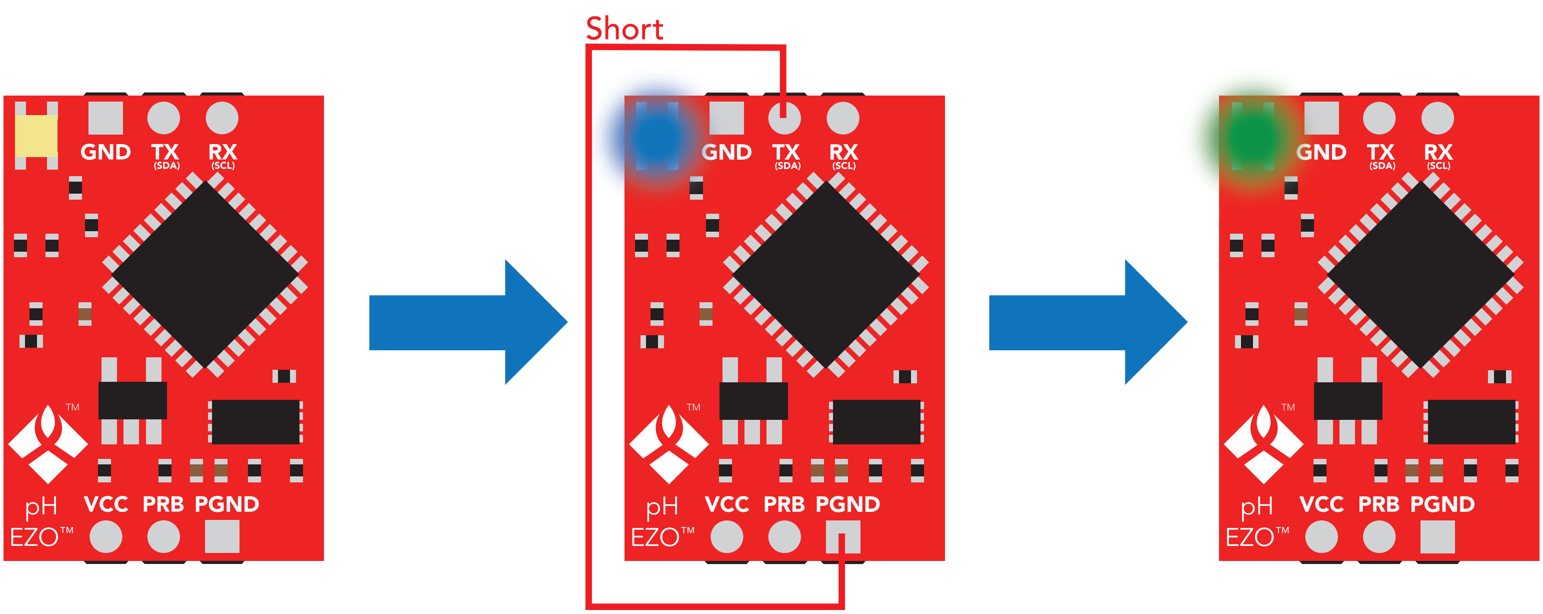
?STATUS,S,5.038

Where: S is the reason for the last reset event (software reset)
Where: 5.038 is the its voltage at the VCC

Manual switching to UART mode

All EZO™ class circuits, can be manually switched from I²C mode to UART mode.
If this is done, the EZO™ class pH circuit, will set its baud rate to 9600.

1. Cut the power to the device
2. Disconnect any jumper wires going from TX and RX to the master micro controller
3. Short the PGND pin to the TX pin
4. Power the device
5. Wait for LED to change from **Blue** to **Green**
6. Remove the short from the probe pin to the TX pin
7. Power cycle the device
8. The device is now UART mode



Protocol lock

This feature is available on All EZO™ class circuits running firmware version 1.95 or higher.

When the protocol lock feature is enabled all changes to the communication protocol are blocked. This means that whatever communication mode the device is in (I^2C or UART); that communication protocol cannot be changed by any means. Furthermore, changes to the devices baud rate or I^2C address also cannot be changed.

By default the protocol lock is: **DISABLED**
Changes to this setting are retained even if the power is cut.

Command syntax

PLOCK,1	Enables the protocol lock feature
PLOCK,0	Disables the protocol lock feature
PLOCK,?	Query the state of the lock

Device response

PLOCK,1
After 300ms, an I^2C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

PLOCK,0
After 300ms, an I^2C read command can be issued to get the response code.
A decimal 1 would indicate the command has been successfully processed.

PLOCK,?
After 300ms, an I^2C read command can be issued to get the response code.

If locked

(?PLOCK,1)

If unlocked

?	P	L	O	C	K	,	0	null

(?PLOCK,0)

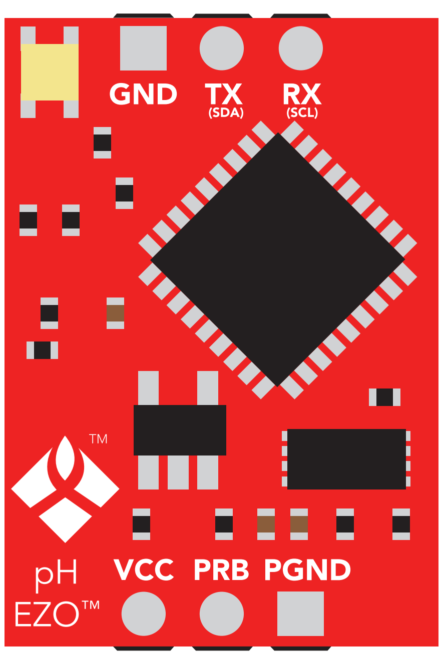
Once the protocol lock is enable the following commands / actions will no longer work

Manual switching to I²C or UART mode
Software switching from I²C or UART mode
UART baud rate change
I²C address change

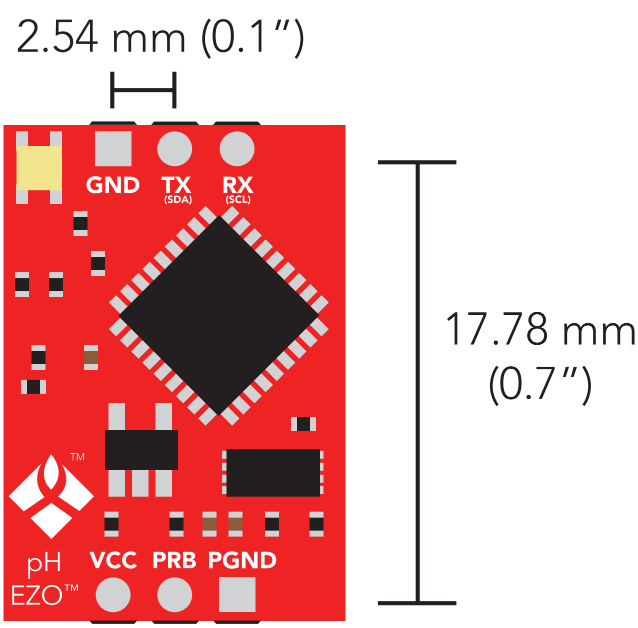
If the protocol lock is enabled attempting to manually switch to UART or I²C mode will have no effect. If any of the software commands are issued to switch mode, baud rate or I²C address the EZO class device will return:

2	null

Indicating that the request failed.

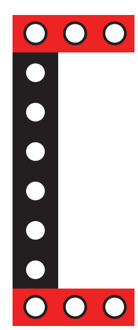


How to make a footprint for the Atlas Scientific™ EZO™ pH circuit

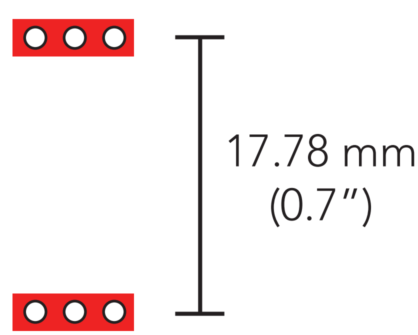


1. In your CAD software place an 8 position header.

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



3. Once this is done, you can delete the 8 position header. Make sure that the two 3 position headers are 17.78mm (0.7") apart from each other.



Datasheet change log

Datasheet V 3.7

Revised cover pages and removed circuit dimensions page.

Datasheet V 3.6

Added I2C timing chart on pg 46.

Datasheet V 3.5

Updated Firmware changes on page 70.

Datasheet V 3.4

Added Protocol Lock function on pages 41 and 65.

Datasheet V 3.3

Revised warning messages on pages 12 and 13.

Datasheet V 3.2

Revised photos on pages 11 and 17.

Datasheet V 3.1

Added Slope command for both UART and I2C on pg 32 and 55 respectively.

Datasheet V 3.0

Expanded Calibration sections for both UART and I2C on pg 27 and 48 respectively.

Datasheet V 2.9

Added I^2C address change command on pg 50.

Datasheet V 2.8

Moved wiring diagram to pg 10

Datasheet V 2.7

"Circuit identification" on page 4.

Datasheet V 2.6

Added

"After the device is woken up, 4 consecutive readings should be taken before the readings are considered valid." to pages 26 and 44

Replaced

"X" command to "Factory"

Datasheet change log

pH circuit firmware changes

V1.0 – Initial release (Aug 1, 2014)

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

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

V1.9 – Factory (April 14, 2015)
- Changed "X" command to "Factory"

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

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

Warranty

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

The debugging phase

The debugging phase as defined by Atlas Scientific™, is the time period when the EZO™ class pH circuit is inserted into a bread board, or shield, and is connected to a microcontroller according to the wiring diagram on pg. 50. Reference this wiring diagram for a connection to USB debugging device, or if a shield is being used, when it is connected to its carrier board.

If the EZO™ class pH circuit is being debugged in a bread board, the bread board must be devoid of other components. If the EZO™ class pH circuit is being connected to a microcontroller, the microcontroller must be running code that has been designed to drive the EZO™ class pH circuit exclusively and output the EZO™ class pH circuit data as a serial string.

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

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

Reasoning behind this warranty

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

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

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

TM

TM