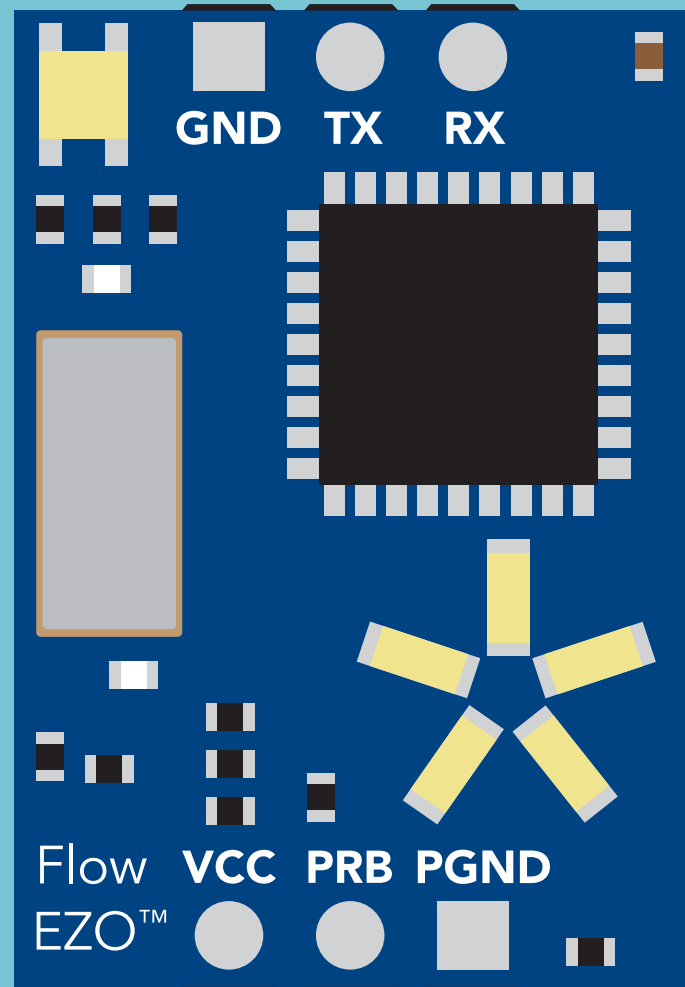


# EZO-FLO<sup>TM</sup>

## Embedded Flow Meter Totalizer

Reads	<b>Total flow and flow rate</b>
Preprogrammed	<b>Works with all Atlas Scientific flow meters</b>
Programmable	<b>Can work with most off-the-shelf flow meters</b>
Visual display	<b>Real time turbine rotation</b>
Data protocol	<b>UART &amp; I<sup>2</sup>C</b>
Default I <sup>2</sup> C address	<b>104 (0x68)</b>
Operating voltage	<b>3.3V – 5V</b>
Data format	<b>ASCII</b>



# Temporary Update



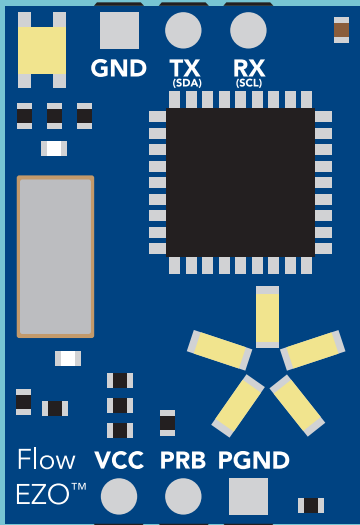
**Due to significant component shortages some flow meters may no longer work at 3.3V.**

While the EZO-FLO still works at 3.3 volts, that may no longer be enough voltage to drive the actual flow meters. Some batches of flow meters will run at 5V and other will not; this is secondary to sporadic component availability.

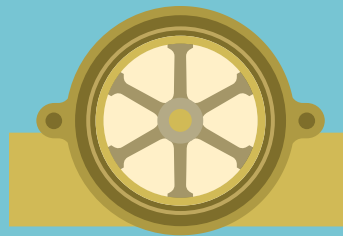


**"I promise to solve fake problems, but do nothing about real ones."**

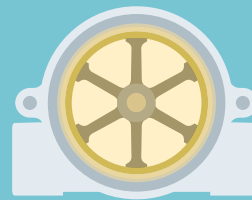
# Attention



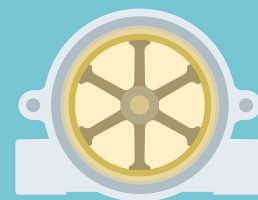
The EZO-FLO™ circuit is fully compatible with any flow meter sold by Atlas Scientific.



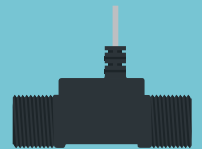
**3/4"**  
Flow Meter



**1/2"**  
Flow Meter  
(default)



**1/4"**  
Flow Meter



**3/8"**  
Flow Meter

See page **17** to see how set the flow meter in

**UART**  
mode

See page **43** to see how set the flow meter in

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

The EZO-FLO™ circuit is also compatible with most off the shelf, volumetric flow meters. See page **60** for more information about how to use the EZO-FLO™ with your own flow meter.



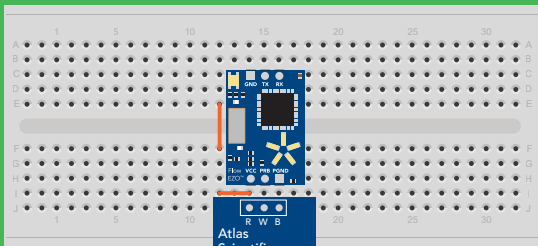
# STOP

**SOLDERING THIS DEVICE VOIDS YOUR WARRANTY.**

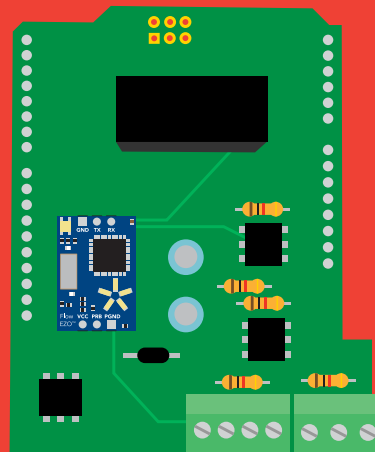
**This is sensitive electronic equipment. Get this device working in a solderless breadboard first. Once this device has been soldered, it is no longer covered by our warranty.**

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

**Get this device working in a solderless breadboard first!**



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



# Table of contents

Circuit dimensions	6	Operating principle	7
Power consumption	6	Default state	9
Absolute max ratings	6	Available data protocols	10

## UART

UART mode	12
Receiving data from device	13
Sending commands to device	14
LED color definition	15
<b>UART quick command page</b>	<b>16</b>
Set flow meter type	17
LED control	18
Find	19
Continuous reading mode	20
Single reading mode	21
Clearing the total volume	22
Change flow rate display	23
Conversion factor	24
Enable/disable parameters	25
Naming device	26
Device information	27
Response codes	28
Reading device status	29
Sleep mode/low power	30
Change baud rate	31
Protocol lock	32
Factory reset	33
Change to I <sup>2</sup> C mode	34
Manual switching to I <sup>2</sup> C	35

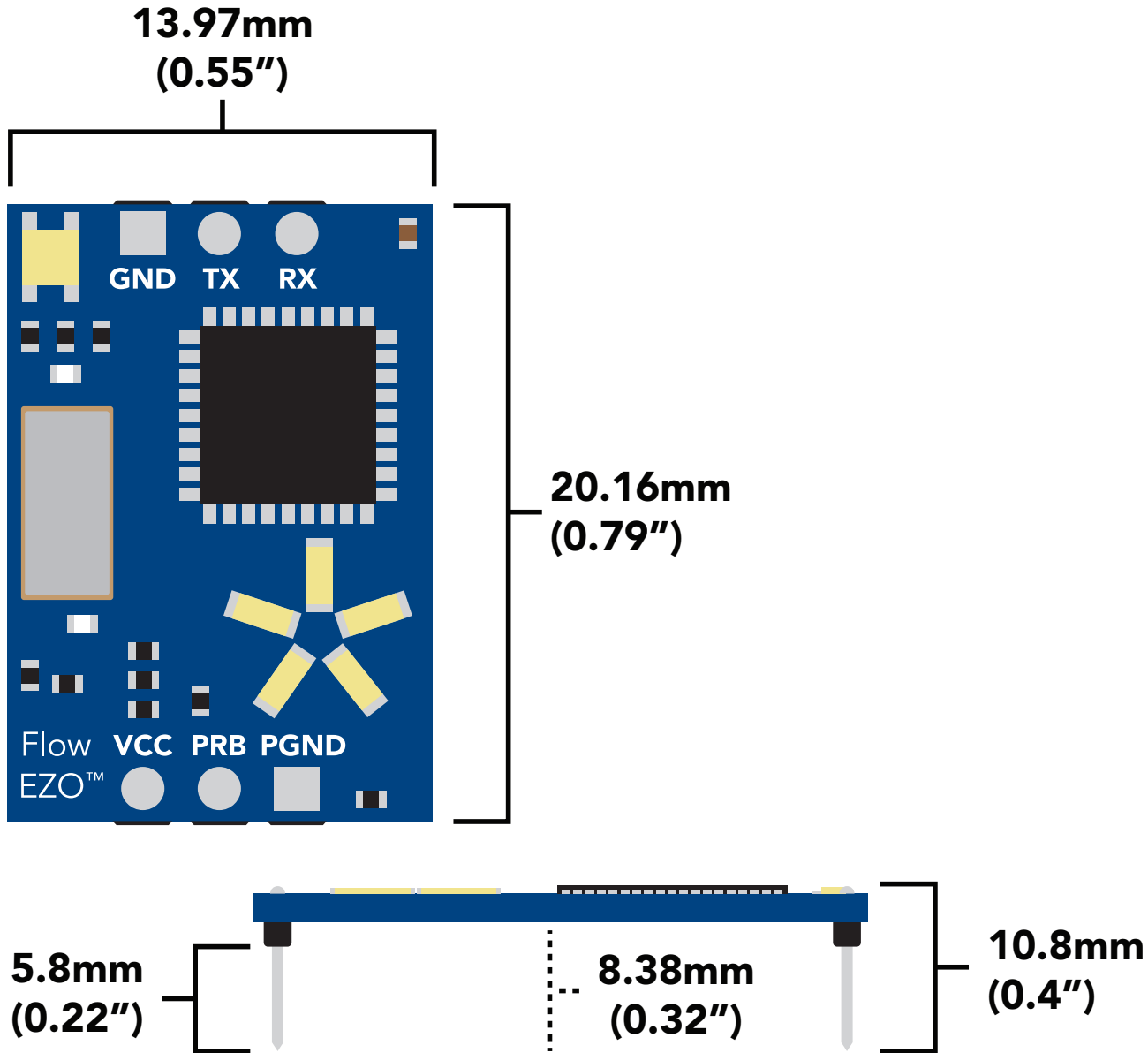
## I<sup>2</sup>C

I <sup>2</sup> C mode	37
Sending commands	38
Requesting data	39
Response codes	40
LED color definition	41
<b>I<sup>2</sup>C quick command page</b>	<b>42</b>
Set flow meter type	43
LED control	44
Find	45
Taking reading	46
Clearing the total volume	47
Change flow rate display	48
Conversion factor	49
Enable/disable parameters	50
Naming device	51
Device information	52
Reading device status	53
Sleep mode/low power	54
Protocol lock	55
I <sup>2</sup> C address change	56
Factory reset	57
Change to UART mode	58
Manual switching to UART	59

## CUSTOM FLOW METER

Compatability	61	Circuit footprint	71
Take notice	63	Datasheet change log	72
Programing	64	Warranty	74
Setting the K values	65		
Setting the time base	67		
Setting the onboard resistors	69		

# EZO™ circuit dimensions



## Power consumption

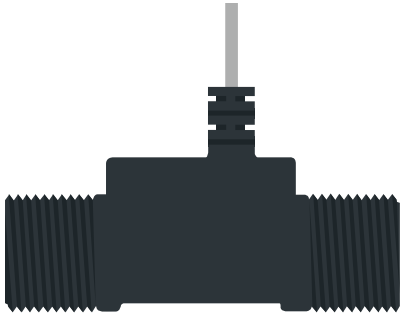
	LED	MAX	STANDBY	SLEEP
5V	ON	21.0 mA	20.5 mA	300µA
	OFF	17.0 mA	16.5 mA	
3.3V	ON	16.6 mA	16.1 mA	131µA
	OFF	15.0 mA	15.0 mA	

## Absolute max ratings

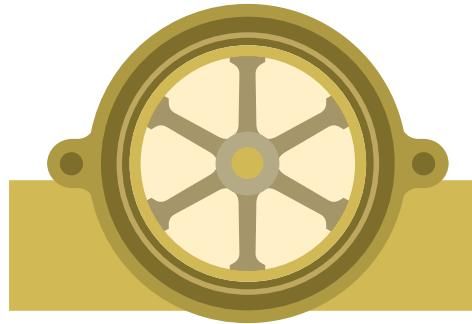
Parameter	MIN	TYP	MAX
Storage temperature (EZO™ FLO)	-40 °C		125 °C
Operational temperature (EZO™ FLO)	-30 °C	25 °C	100 °C
VCC	3.3V	5V	5.5V

# Operating principle

The most common types of volumetric flow meters on the market today are turbine and paddled wheel flow meters.

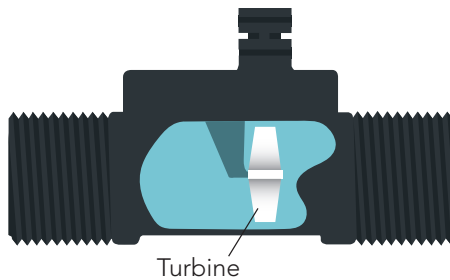
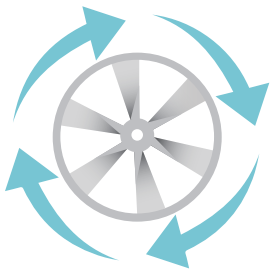


**Turbine flow meter**



**Paddled wheel flow meter**

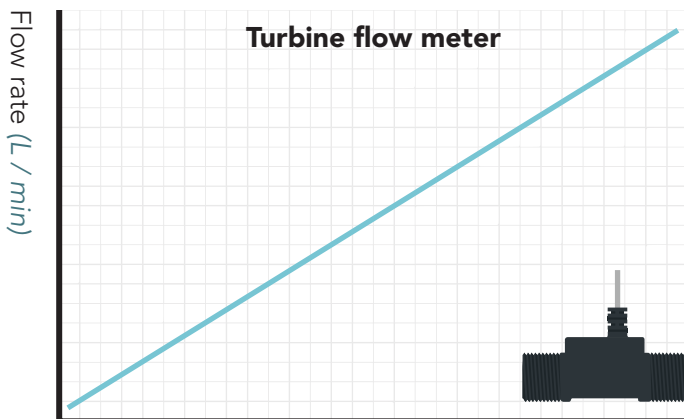
Generally speaking, turbine flow meters are the simplest to work with and offer the highest accuracy. With this type of flow meter, each rotation of the turbine represents a volume of liquid passing through the meter.



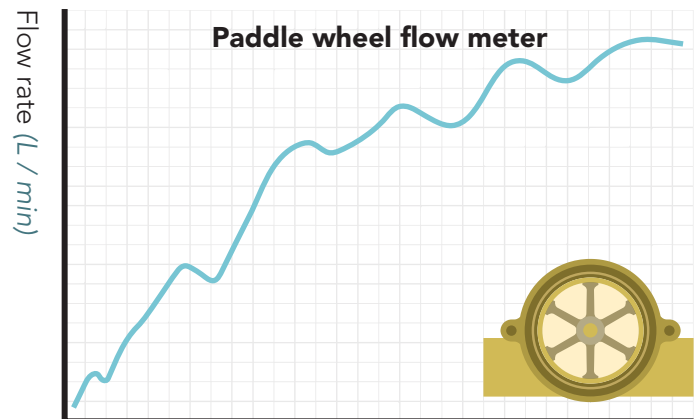
**1 rotation = 367 $\mu$ L**

Although these flow meters are highly accurate and easy to work with, they are only cost-effective in small sizes. (A turbine flow meter just twice the size of the one pictured above, cost six times as much).

Unlike turbine flow meters, paddled wheel flow meters use frequency to calculate water flow. The frequency is a representation of the water current traveling through the flow meter. Most times the relationship between water current (frequency) and volume is not linear, and complex math must be used to derive the flow rate.



Flow meter output (*Rotation*)



Flow meter output (*Hz*)

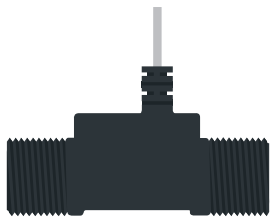
No matter what type of flow meter is used, the output from that flow meter must be rapidly calculated and totalized continuously. The computer system that converts the output of a flow meter to a meaningful value is called a flow meter totalizer.



Flow meter totalization should always be done on a separate computer system that has been specifically designed to calculate the flow rate continuously. If not, the engineer runs the risk of missing a few pulses here and there while the computer system is performing other tasks. This can lead to VAST miscalculations in flow rates over a relatively short amount of time.

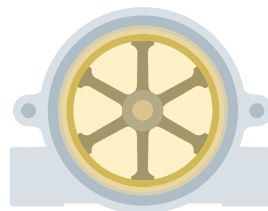
# Atlas Scientific flow meters

Although this device can be used with many different types of flow meters, Atlas Scientific has preprogrammed the EZO-FLO™ to work with 4 different types of flow meters. These flow meters have been selected because of their quality, durability, accuracy, and repeatability.



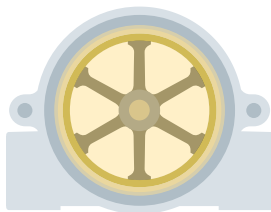
## 3/8" Flow Meter

**Flow rate** 760 mL – 7.6 L / min  
**Accuracy** ±2%  
**Inlet/outlet port** 3/8 NPT male  
**Operating pressure** 0 – 200 PSI  
**Default output** Liters / L per min  
**Operating temperature** -20°C to 80°C  
**Approvals** NSF 61 (Drinking Water Safe)



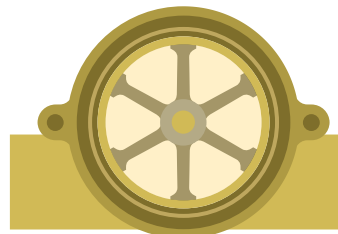
## 1/4" Flow Meter

**Flow rate** 378 mL – 19 L / min  
**Accuracy** ±10%  
**Inlet/outlet port** 1/4 NPT female  
**Operating pressure** 0 – 100 PSI  
**Default output** Liters / L per min  
**Operating temperature** -29°C to 82°C  
**Approvals** NSF 61 (Drinking Water Safe)



## 1/2" Flow Meter

**Flow rate** 378 mL – 19 L / min  
**Accuracy** ±10%  
**Inlet/outlet port** 1/2 NPT female  
**Operating pressure** 0 – 100 PSI  
**Default output** Liters / L per min  
**Operating temperature** -29°C to 82°C  
**Approvals** NSF 61 (Drinking Water Safe)



## 3/4" Flow Meter

**Flow rate** 19 L – 114 L / min  
**Accuracy** ±10%  
**Inlet/outlet port** 3/4 NPT female  
**Operating pressure** 0 – 200 PSI  
**Default output** Liters / L per min  
**Operating temperature** -29°C to 100°C  
**Approvals** NSF 61 (Drinking Water Safe)



# Default state

# UART mode

**Baud** 9,600

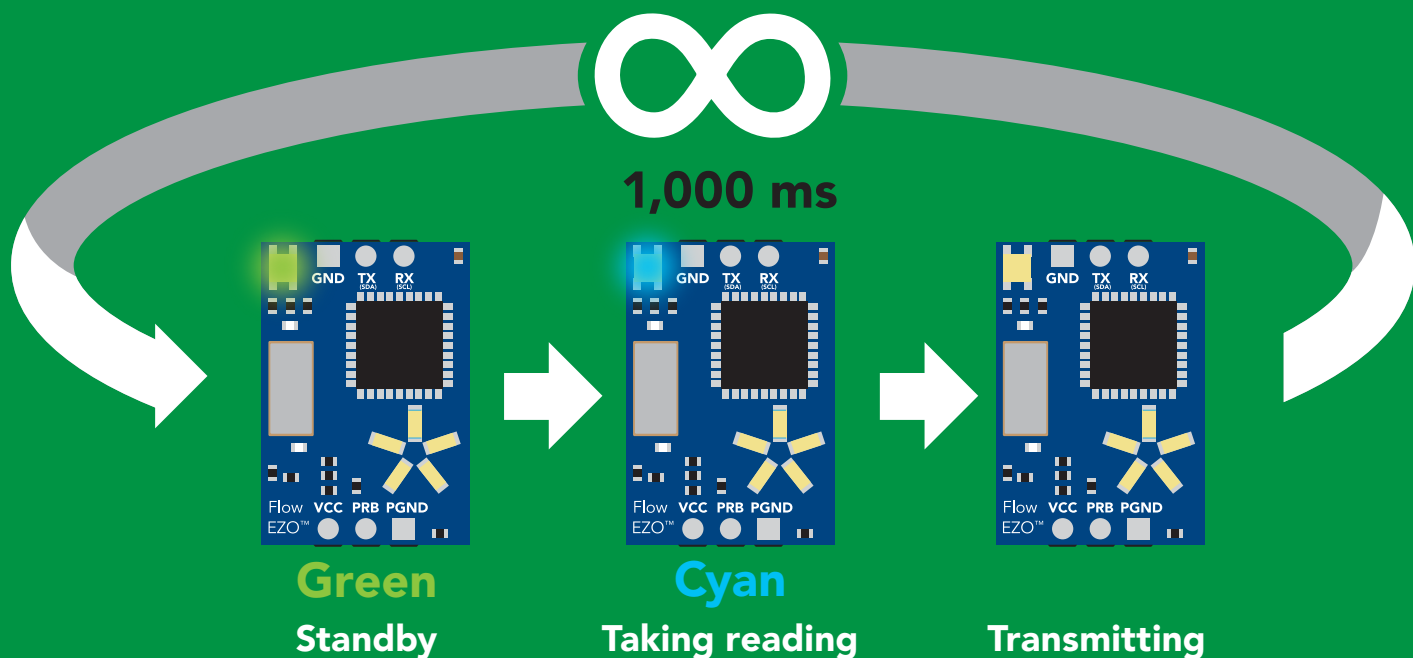
**Readings** continuous

**Flow meter** 1/2" Flow meter

**Units** total volume

**Speed** 1 reading per second

**LED** on



See page 22 to enable the secondary output:  
**flow rate per (min, sec or hour)**

See page 16 to set your flow meter type.

# ✓ Available data protocols

# UART

Default

# I<sup>2</sup>C

# X Unavailable data protocols

# SPI

# Analog

# RS-485

# Mod Bus

# 4–20mA

# UART mode

## Settings that are retained if power is cut

- Baud rate
- Continuous mode
- Conversion factor
- Device name
- Enable/disable response codes
- Flow meter settings
- 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

- All calculated flow
- Find
- Sleep mode

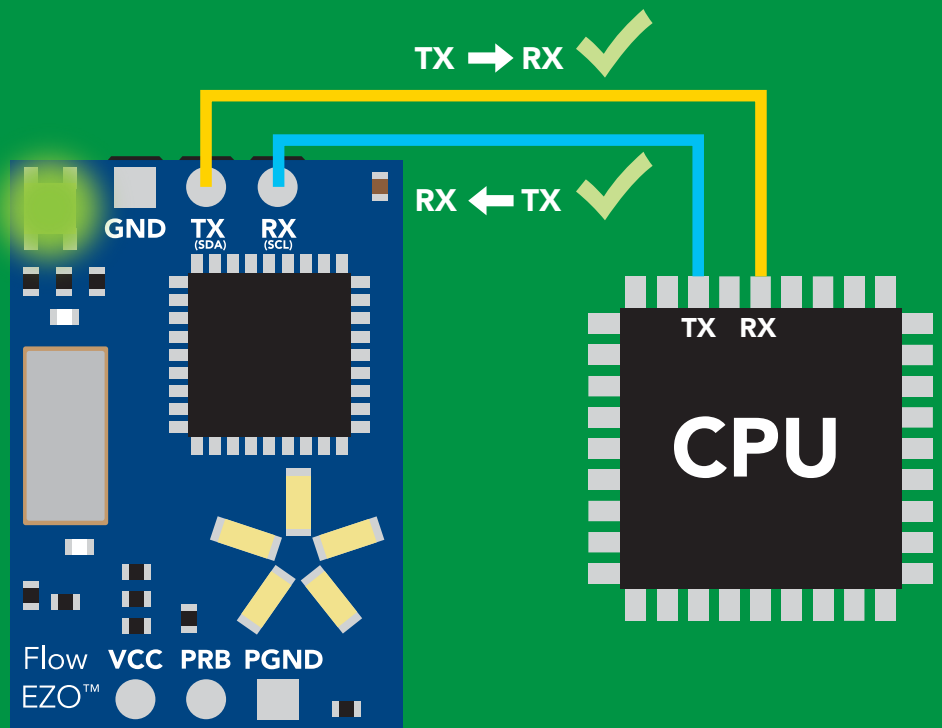
# UART mode

8 data bits      no parity  
1 stop bit        no flow control

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



**Vcc** 3.3V – 5.5V



## Data format

<b>Reading</b>	<b>Total volume</b> and <b>Flow rate per</b> (sec, min or hour)	<b>Terminator</b>	<b>carriage return</b>
<b>Units</b>	<b>Liters</b> and <b>liters per min</b>	<b>Data type</b>	<b>floating point</b>
<b>Encoding</b>	<b>ASCII</b>	<b>Decimal places</b>	<b>2</b>
<b>Format</b>	<b>string</b>	<b>Smallest string</b>	<b>3 characters</b>
		<b>Largest string</b>	<b>32 characters</b>

# Receiving data from device

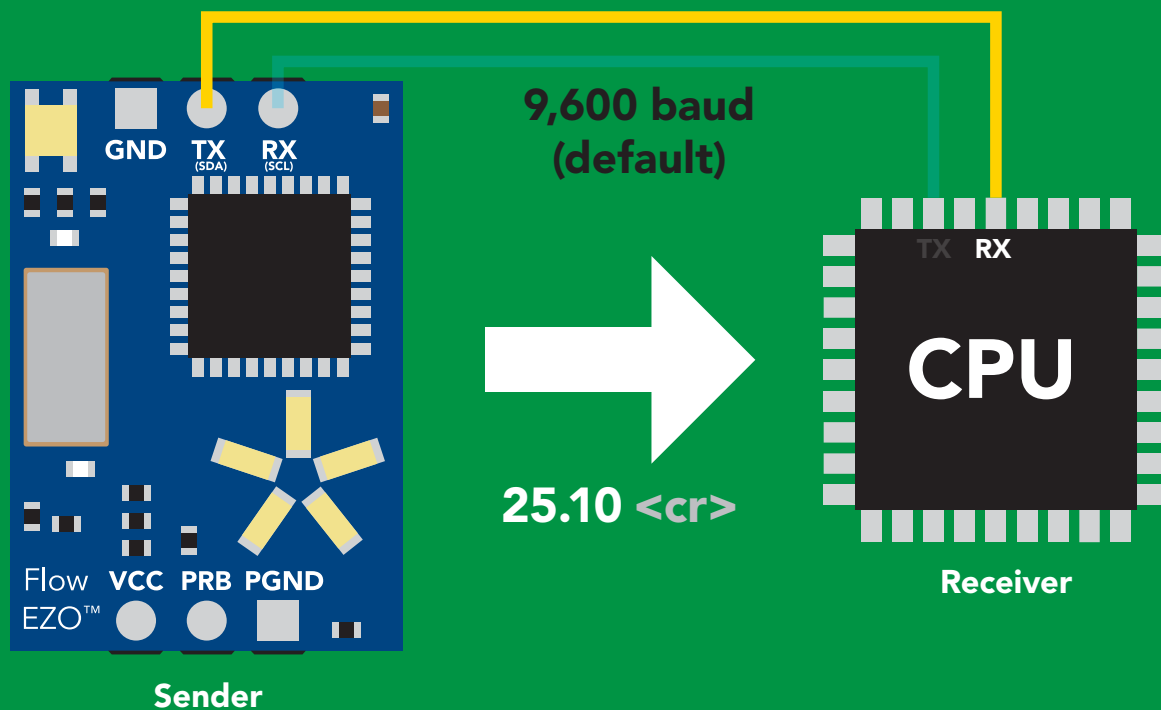
2 parts

ASCII data string

Command

Carriage return <cr>

Terminator



## Advanced

ASCII: 2 5 . 1 0 <cr>

Hex: 32 35 2E 31 30 0D

Dec: 50 53 46 49 48 13

# Sending commands to device

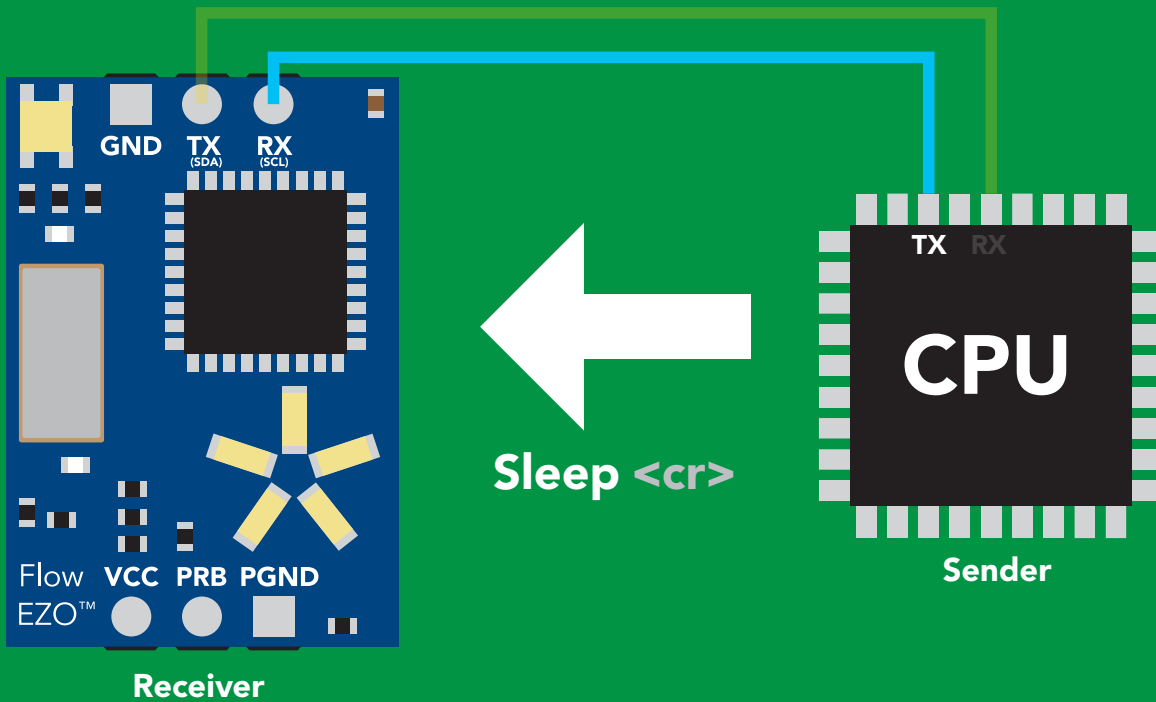
2 parts

**Command (not case sensitive)**

ASCII data string

**Carriage return <cr>**

Terminator



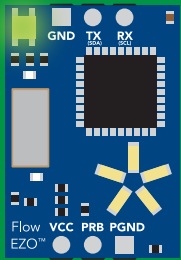
## Advanced

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

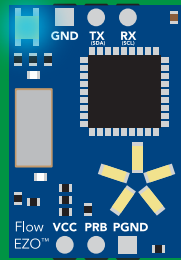
Hex: **53 6C 65 65 70** **0D**

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

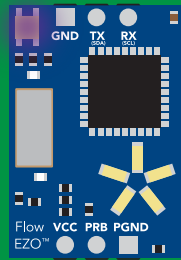
# LED color definition



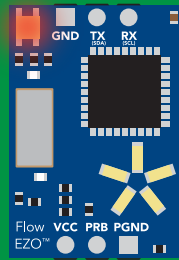
**Green**  
UART standby



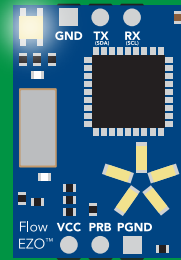
**Cyan**  
Taking reading



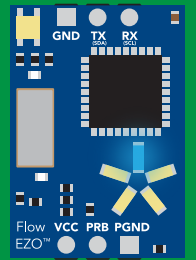
**Purple**  
Changing baud rate



**Red**  
Command not understood



**White**  
Find



**Blue**  
Set flow meter type

**5V**

LED ON  
**+2.6 mA**

**3.3V**

**+0.7 mA**

# UART mode

## command quick reference

All commands are ASCII strings or single ASCII characters.

Command	Function		Default state
Baud	change baud rate	pg. 31	9,600
C	enable/disable continuous reading	pg. 20	enabled
CF	conversion factor	pg. 24	n/a
Clear	clearing the total volume	pg. 22	n/a
Factory	enable factory reset	pg. 33	n/a
Find	finds device with blinking white LED	pg. 19	n/a
Frp	change flow rate calculation	pg. 23	minute
i	device information	pg. 27	n/a
I2C	change to I <sup>2</sup> C mode	pg. 34	not set
L	enable/disable LED	pg. 18	enabled
Name	set/show name of device	pg. 26	not set
O	enable/disable parameters	pg. 25	all enabled
Plock	enable/disable protocol lock	pg. 32	disabled
R	returns a single reading	pg. 21	n/a
Set	set flow meter type	pg. 17	n/a
Sleep	enter sleep mode/low power	pg. 30	n/a
Status	retrieve status information	pg. 29	n/a
*OK	enable/disable response codes	pg. 28	enable



# Set flow meter type

## Command syntax

**Set,3/8** <cr> set to 3/8" flow meter

**Set,1/4** <cr> set to 1/4" Flow meter

**Set,1/2** <cr> set to 1/2" Flow meter **default**

**Set,3/4** <cr> set to 3/4" Flow meter

**Set,?** <cr> show set flow meter

## Example

**Set,1/4** <cr>

## Response

**\*OK** <cr>

**Set,?** <cr>

**?Set,1/4"** <cr> **or** **?Set,0** <cr>  
1/4" flow meter no flow meter

**or Set,custom** <cr>  
set to a custom flow meter

# LED control

## Command syntax

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

L,0 <cr> LED off

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

## Example

## Response

L,1 <cr>

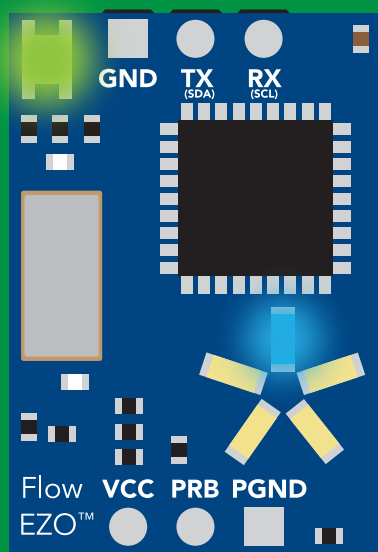
\*OK <cr>

L,0 <cr>

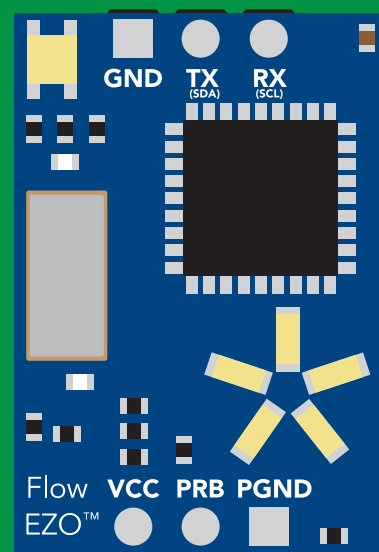
\*OK <cr>

L,? <cr>

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



L,1



L,0

# Find

## Command syntax

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

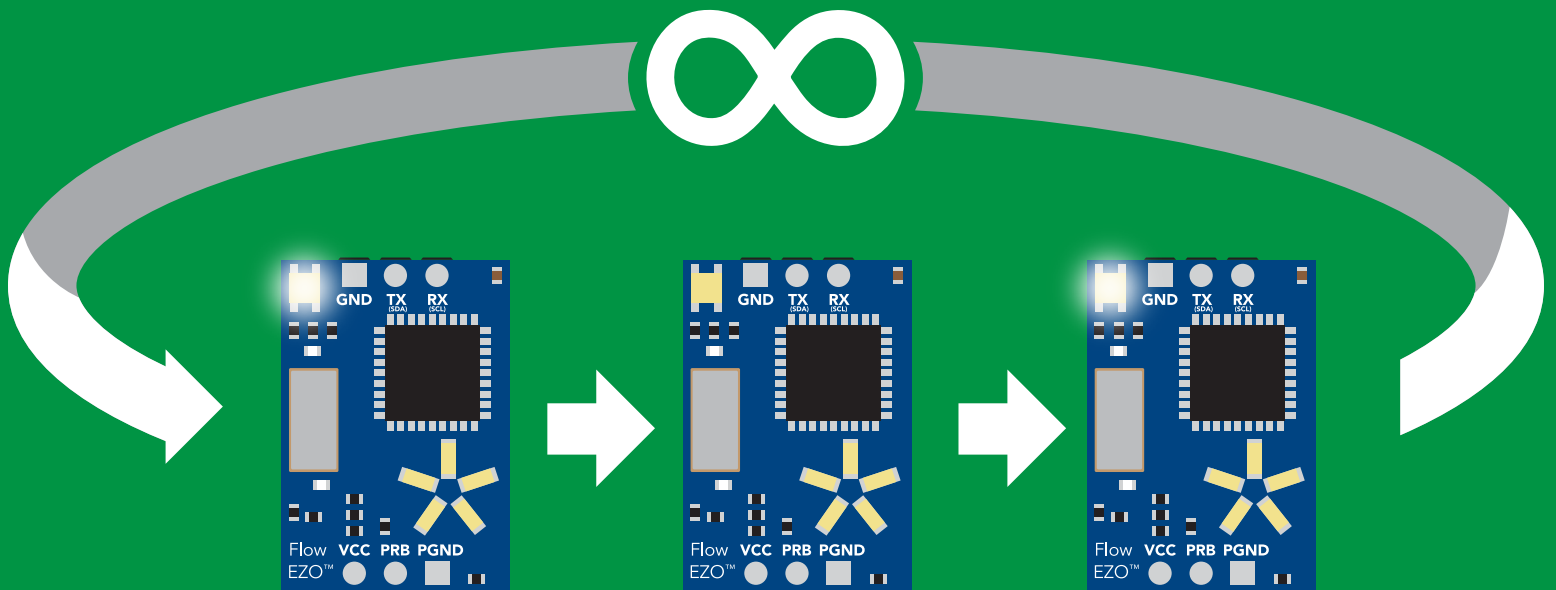
Find <cr> LED rapidly blinks white, used to help find device

## Example

## Response

Find <cr>

\*OK <cr>



# Continuous reading mode

## Command syntax

- C,1 <cr>** enable continuous readings once per second **default**
- C,n <cr>** continuous readings every n seconds (n = 2 to 99 sec)
- C,0 <cr>** disable continuous readings
- C,? <cr>** continuous reading mode on/off?

## Example

## Response

**C,1 <cr>**

**\*OK <cr>**  
**Volume, flow rate (1 sec) <cr>**  
**Volume, flow rate (2 sec) <cr>**  
**Volume, flow rate (n sec) <cr>**

**C,30 <cr>**

**\*OK <cr>**  
**Volume, flow rate (30 sec) <cr>**  
**Volume, flow rate (60 sec) <cr>**  
**Volume, flow rate (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

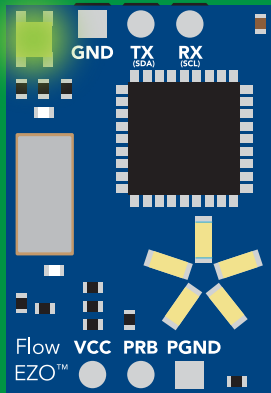
R <cr> takes single reading

### Example

R <cr>

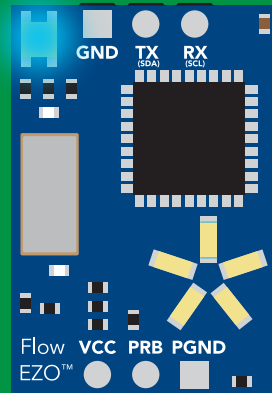
### Response

101.34 <cr>  
\*OK <cr>



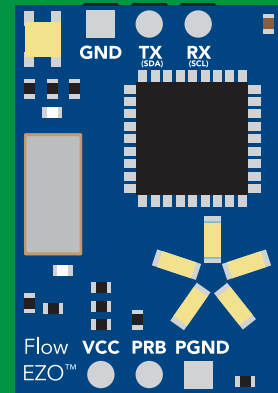
Green

Standby



Cyan

Taking reading



Transmitting



600 ms

# Clearing the total volume

## Command syntax

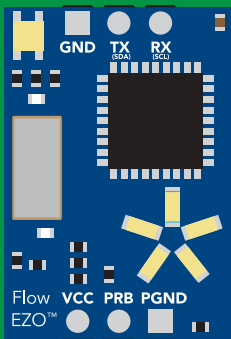
**Clear** <cr> clears the total volume, resets counter to 0.00

### Example

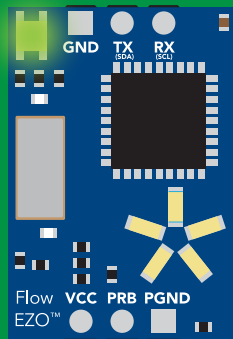
**Clear** <cr>

### Response

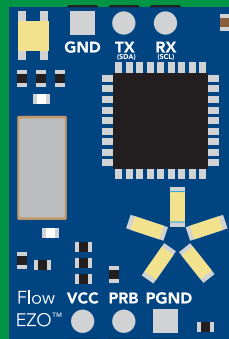
**\*OK** <cr>



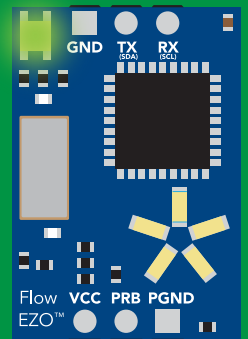
**R** <cr>



**534.26** <cr>



**Clear** <cr>



**0.00** <cr>

# Change flow rate display

This command changes the time base of the flow rate.

Total volume / flow rate



## Command syntax

**Frp,s** <cr> calculate flow rate per second

**Frp,m** <cr> calculate flow rate per minute **default**

**Frp,h** <cr> calculate flow rate per hour

**Frp,?** <cr> calculate flow rate per?

## Example

## Response

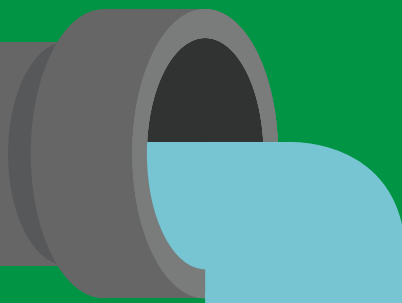
**Frp,h** <cr>

**\*OK** <cr>

**Frp,?** <cr>

**?Frp,h** <cr>

**\*OK** <cr>



— flow rate per second

— flow rate per minute

— flow rate per hour

# Conversion factor

By default all readings are in L/LPM. The Conversion factor command lets you convert the readings to a different measurement.

Conversion factor range= 0.001 - 1,000,000

Liters x CF = converted reading

**Example conversion factors:**

Liters to milliliters =1,000

Liters to gallon = 0.264

## Command syntax

default conversion factor = 1

**CF,n** <cr> set conversion factor

**CF,?** <cr> show conversion factor

### Example

### Response

**R** <cr>

**5.74 (liters)** <cr>  
**\*OK** <cr>

**CF, 0.264** <cr>

**\*OK** <cr>

**R** <cr>

**1.51 (gallons)** <cr>  
**\*OK** <cr>

**CF, ?** <cr>

**?CF,0.264** <cr>  
**\*OK** <cr>



# Enable/disable parameters from output string

## Command syntax

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

## Example

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

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

O,? <cr>

## Response

\*OK <cr> enable / disable total volume

\*OK <cr> enable / disable flow rate

?O,TV,FR <cr> if both are enabled

### Parameters

TV total volume  
FR flow rate

### 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

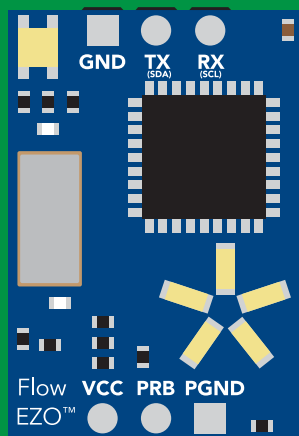
Name,zzt <cr>

\*OK <cr>

Name,? <cr>

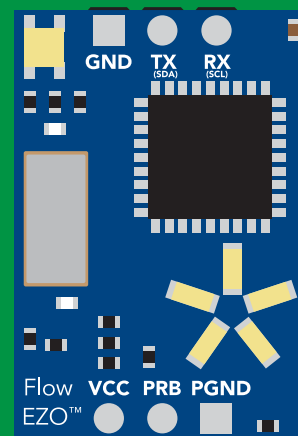
?Name,zzt <cr>  
\*OK <cr>

Name,zzt



\*OK <cr>

Name,?



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

# Device information

## Command syntax

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

### Example

```
i <cr>
```

### Response

```
?i,FLO,2.00 <cr>  
*OK <cr>
```

## Response breakdown

```
?i,  FLO,  2.00  
      ↑      ↑  
    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>

**25.10** <cr>  
**\*OK** <cr>

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

no response, **\*OK** disabled

**R** <cr>

**25.10** <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

16.9 mA

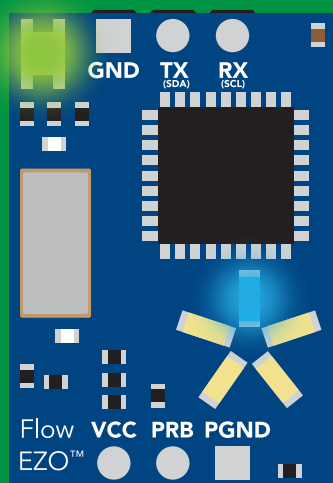
SLEEP

300µA

3.3V

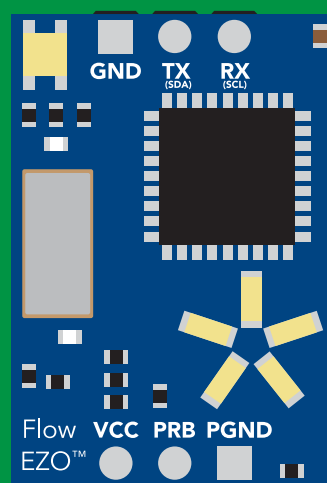
13.2 mA

131µA



Standby  
15.40 mA

Sleep <cr>



Sleep  
3.00 mA

# Change baud rate

## Command syntax

Baud,n <cr> change baud rate

### Example

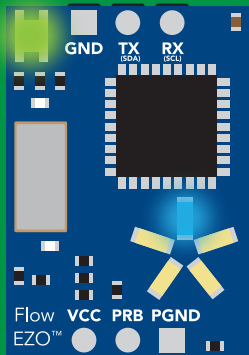
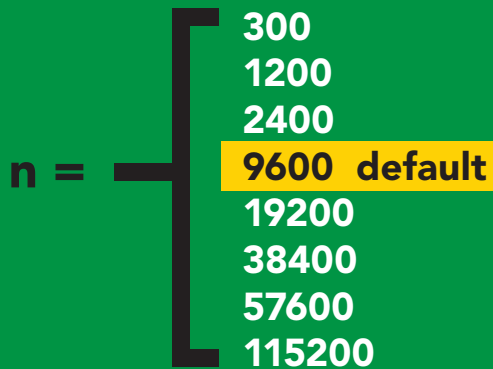
Baud,38400 <cr>

\*OK <cr>

Baud,? <cr>

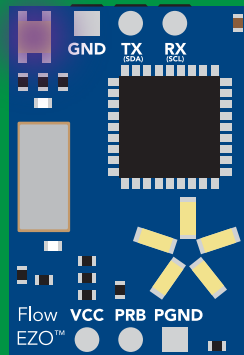
?Baud,38400 <cr>

\*OK <cr>



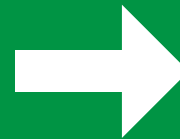
Standby

Baud,38400 <cr>

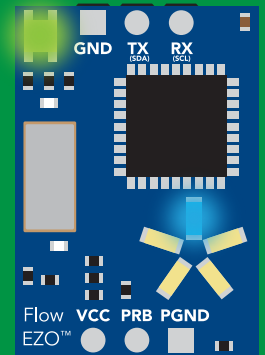


Changing  
baud rate

\*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>

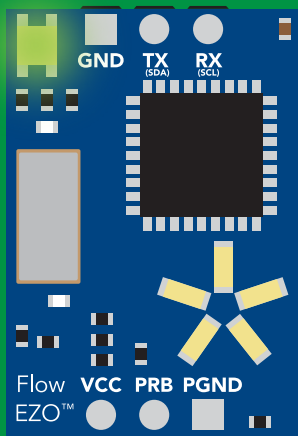
Plock,0 <cr>

\*OK <cr>

Plock,? <cr>

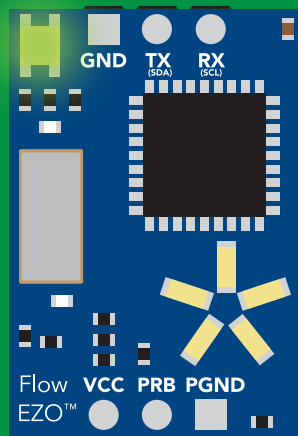
?Plock,1 <cr> or ?Plock,0 <cr>

### Plock,1



\*OK <cr>

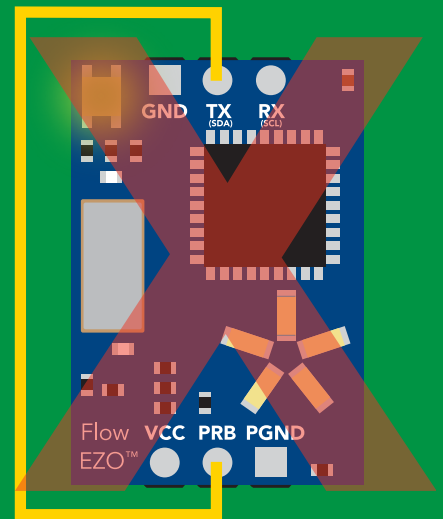
### I2C,100



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

\*ER <cr>

### Short



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



# Factory reset

## Command syntax

Clears all flow meter settings  
Total volume  
LED on  
"\*OK" enabled

Factory <cr> enable factory reset

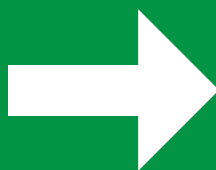
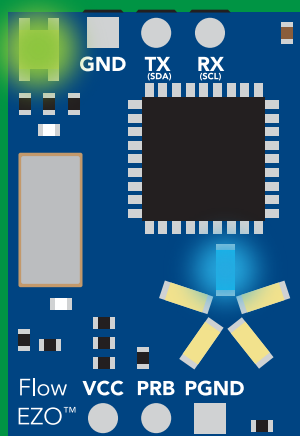
## Example

## Response

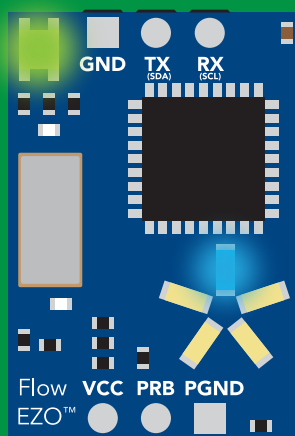
Factory <cr>

\*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 104 (0x68)

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

## Response

I2C,100 <cr>

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

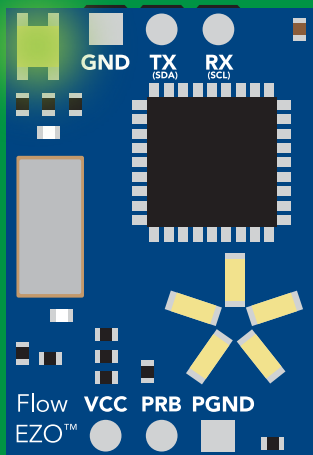
## Wrong example

## Response

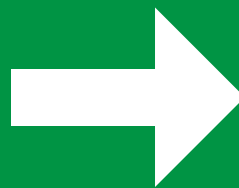
I2C,139 <cr> n ≠ 127

\*ER <cr>

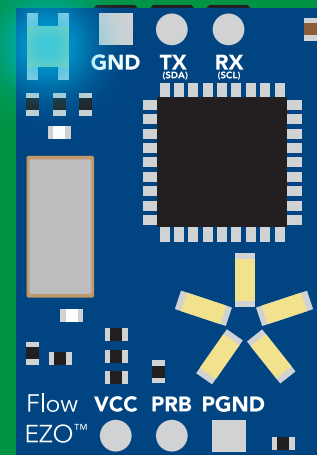
I2C,100



Green  
\*OK <cr>



(reboot)



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

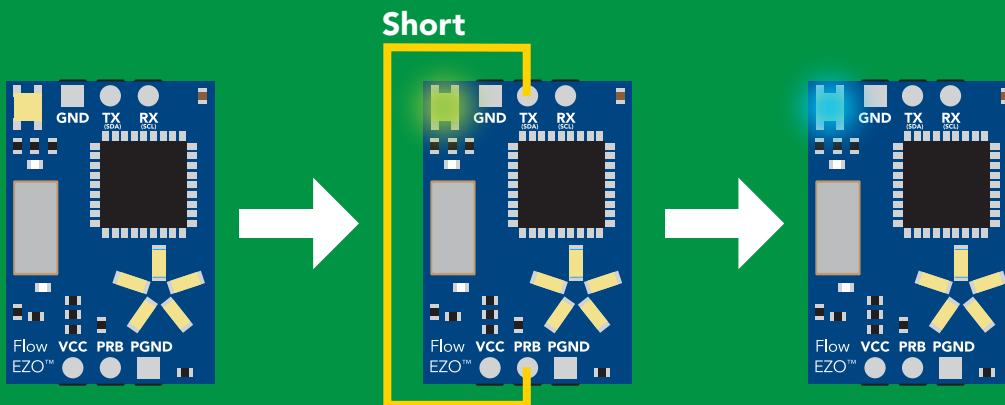
# Manual switching to I<sup>2</sup>C

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

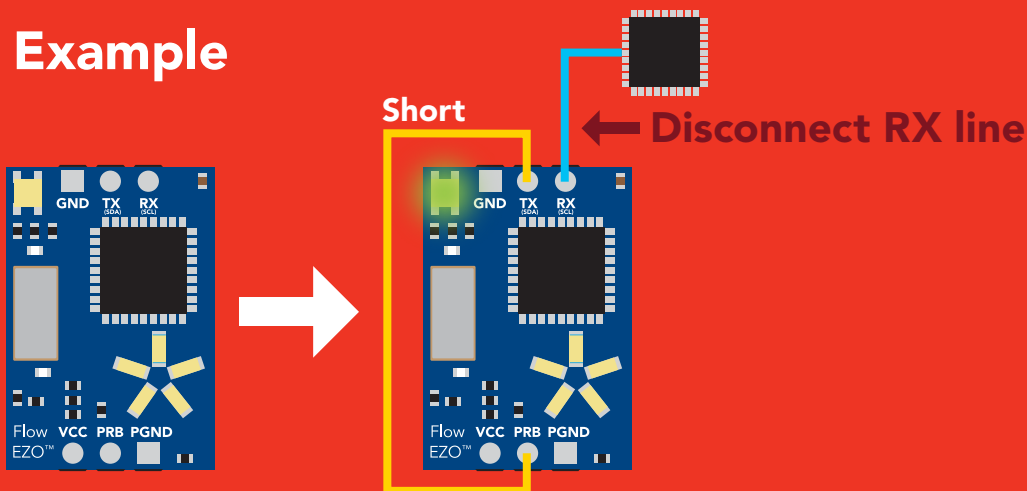
Connecting TX to PRB only works for the EZO-RTD™ and the EZO-FLO™ circuits

Manually switching to I<sup>2</sup>C will set the I<sup>2</sup>C address to 104 (0x68)

## 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

- Change I<sup>2</sup>C address
- Conversion factor
- Flow meter settings
- Hardware switch to UART mode
- LED control
- Protocol lock
- Software switch to UART mode

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

- All calculated flow
- Find
- Sleep mode

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

I<sup>2</sup>C address (0x01 – 0x7F)  
**104 (0x68) default**

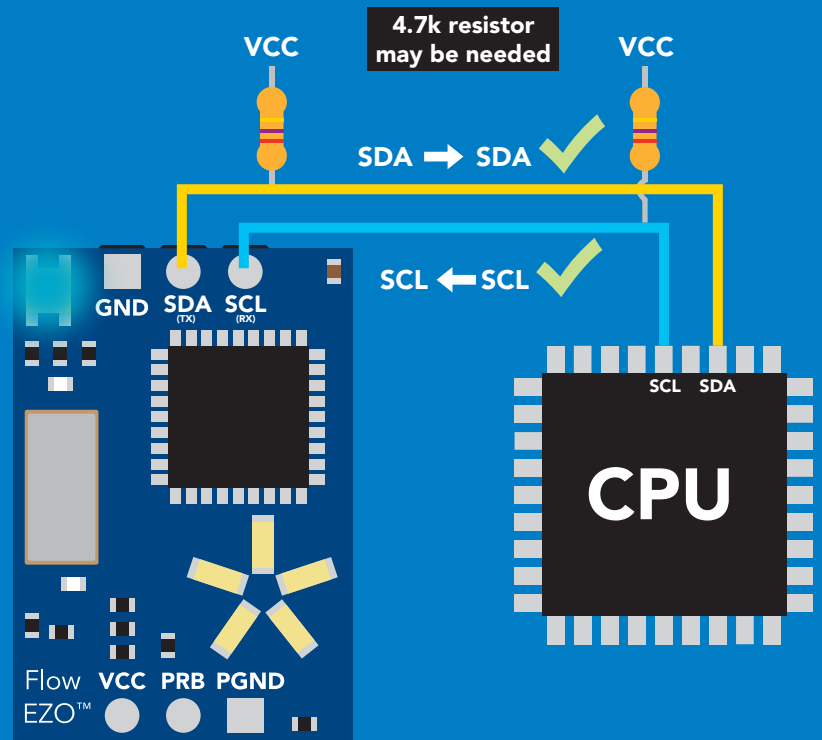
Vcc 3.3V – 5.5V

Clock speed 100 – 400 kHz

SDA 

SCL 





## Data format

Reading Total volume  
and  
Flow rate per  
(sec, min or hour)

Units Liters  
and liters per min

Encoding ASCII

Format string

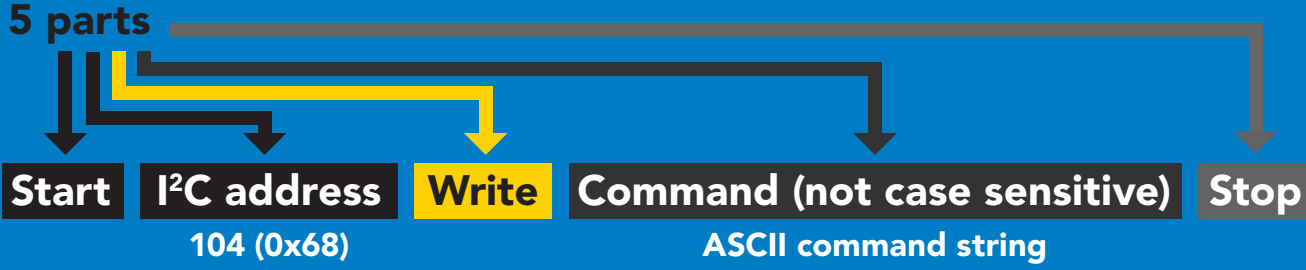
Data type floating point

Decimal places 2

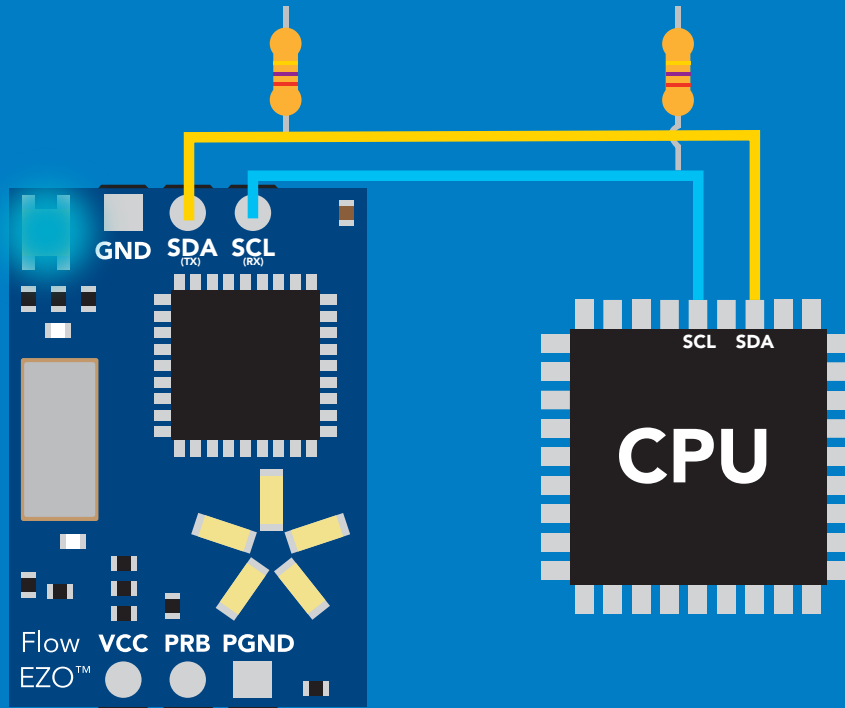
Smallest string 3 characters

Largest string 32 characters

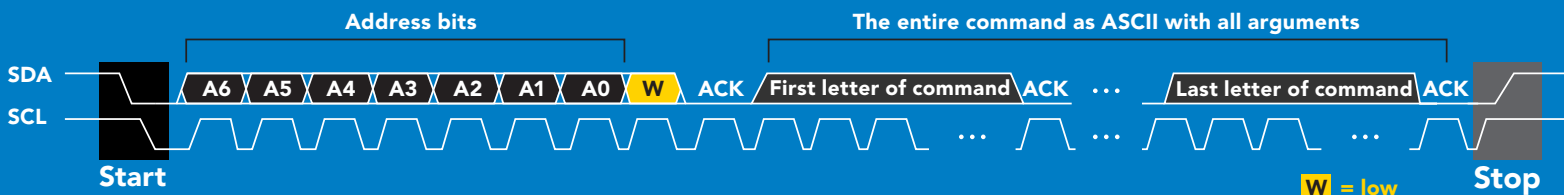
# Sending commands to device



## Example



## Advanced

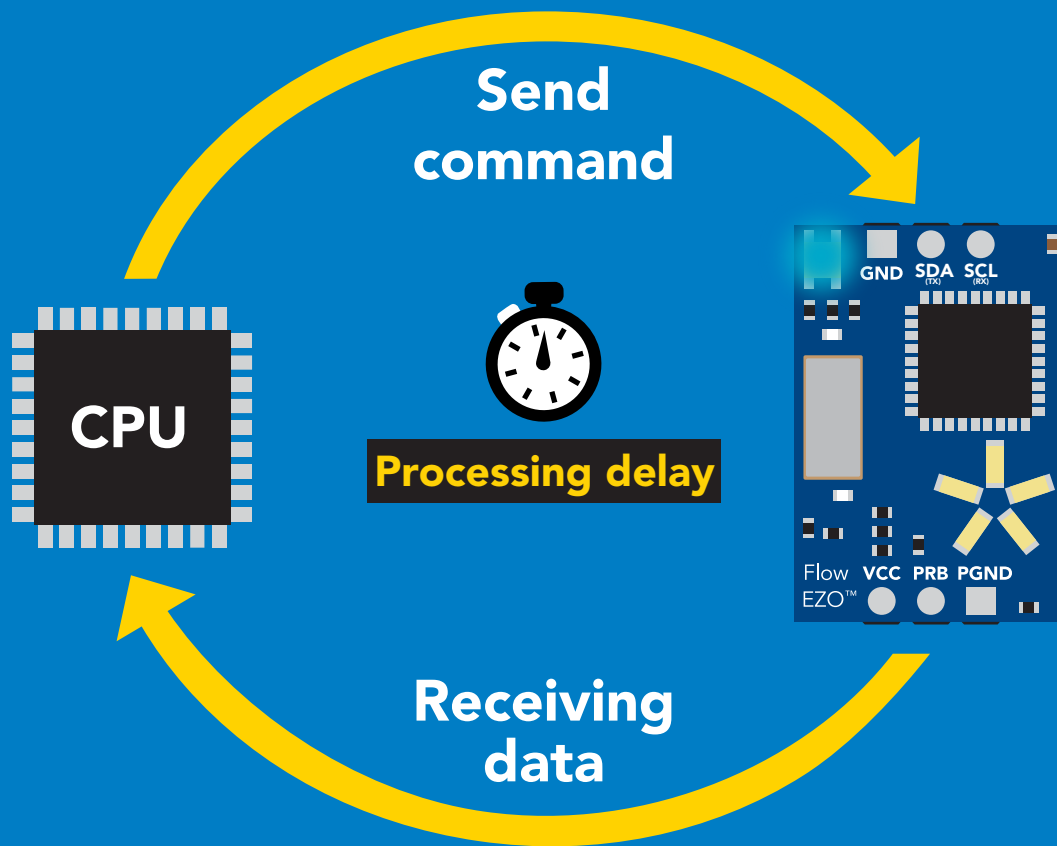




# Response codes

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

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



## Example

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

`delay(300);`



Processing delay

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

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

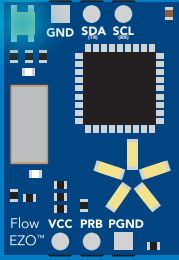
### Response codes

Single byte, not string

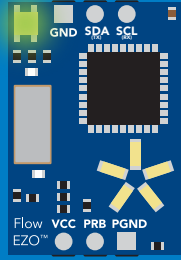
255	no data to send
254	still processing, not ready
2	syntax error
1	successful request



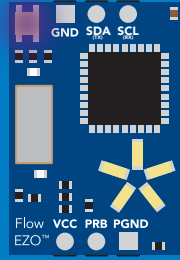
# LED color definition



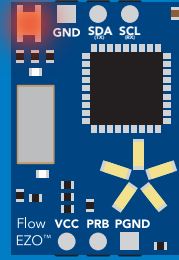
**Blue**  
I2C standby



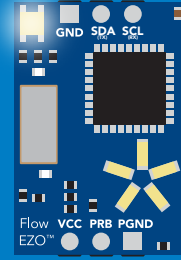
**Green**  
Taking reading



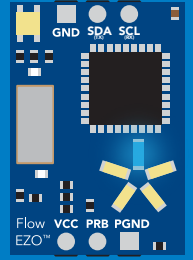
**Purple**  
Changing  
I2C address



**Red**  
Command  
not understood



**White**  
Find



**Blue**  
Set flow  
meter type

**5V**

LED ON  
**+2.6 mA**

**3.3V**

**+0.7 mA**

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

## command quick reference

All commands are ASCII strings or single ASCII characters.

<b>Command</b>	<b>Function</b>	
<b>Baud</b>	switch back to UART mode	<b>pg. 58</b>
<b>CF</b>	conversion factor	<b>pg. 49</b>
<b>Clear</b>	clearing the total volume	<b>pg. 47</b>
<b>Factory</b>	enable factory reset	<b>pg. 57</b>
<b>Find</b>	finds devices with white blinking LED	<b>pg. 45</b>
<b>Frp</b>	change flow rate calculation	<b>pg. 48</b>
<b>i</b>	device information	<b>pg. 52</b>
<b>I2C</b>	change I <sup>2</sup> C address	<b>pg. 54</b>
<b>L</b>	enable/disable LED	<b>pg. 44</b>
<b>Name</b>	set/show name of device	<b>pg. 51</b>
<b>O</b>	enable/disable parameters	<b>pg. 50</b>
<b>Plock</b>	enable/disable protocol lock	<b>pg. 55</b>
<b>R</b>	returns a single reading	<b>pg. 45</b>
<b>Set</b>	set flow meter type	<b>pg. 43</b>
<b>Sleep</b>	enter sleep mode/low power	<b>pg. 54</b>
<b>Status</b>	retrieve status information	<b>pg. 53</b>

# Set flow meter type

## Command syntax

300ms  processing delay

Set,3/8 set to 3/8" flow meter  
Set,1/4 set to 1/4" Flow meter  
Set,1/2 set to 1/2" Flow meter **default**  
Set,3/4 set to 3/4" Flow meter  
Set,? show set flow meter


## Example

Set,1/4

  
Wait 300ms

<b>1</b>	<b>0</b>
Dec	Null

Set,?

  
Wait 300ms

<b>1</b>	<b>?Set,1/4</b>	<b>0</b>	or	<b>1</b>	<b>?Set,0</b>	<b>0</b>
Dec	ASCII	Null		Dec	ASCII	Null
	1/4" flow meter				no flow meter	

or

<b>1</b>	<b>?Set,custom</b>	<b>0</b>
Dec	ASCII	Null
	set to a custom flow meter	

# LED control

## Command syntax

300ms  processing delay

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

## Example

## Response

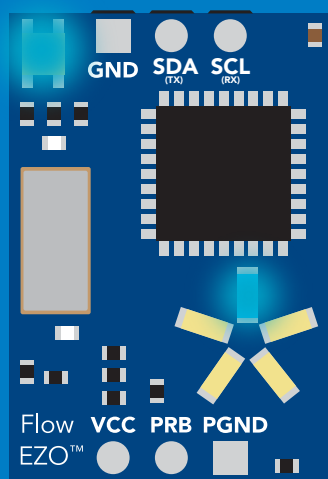
L,1



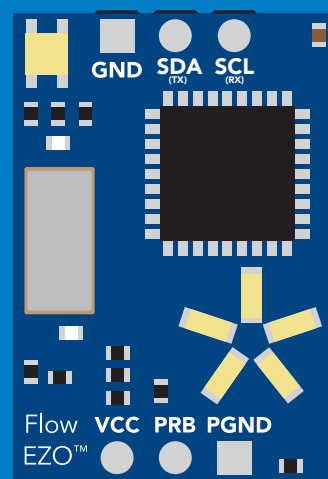
L,0



L,?



L,1



L,0

# Find

300ms  processing delay

## Command syntax

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

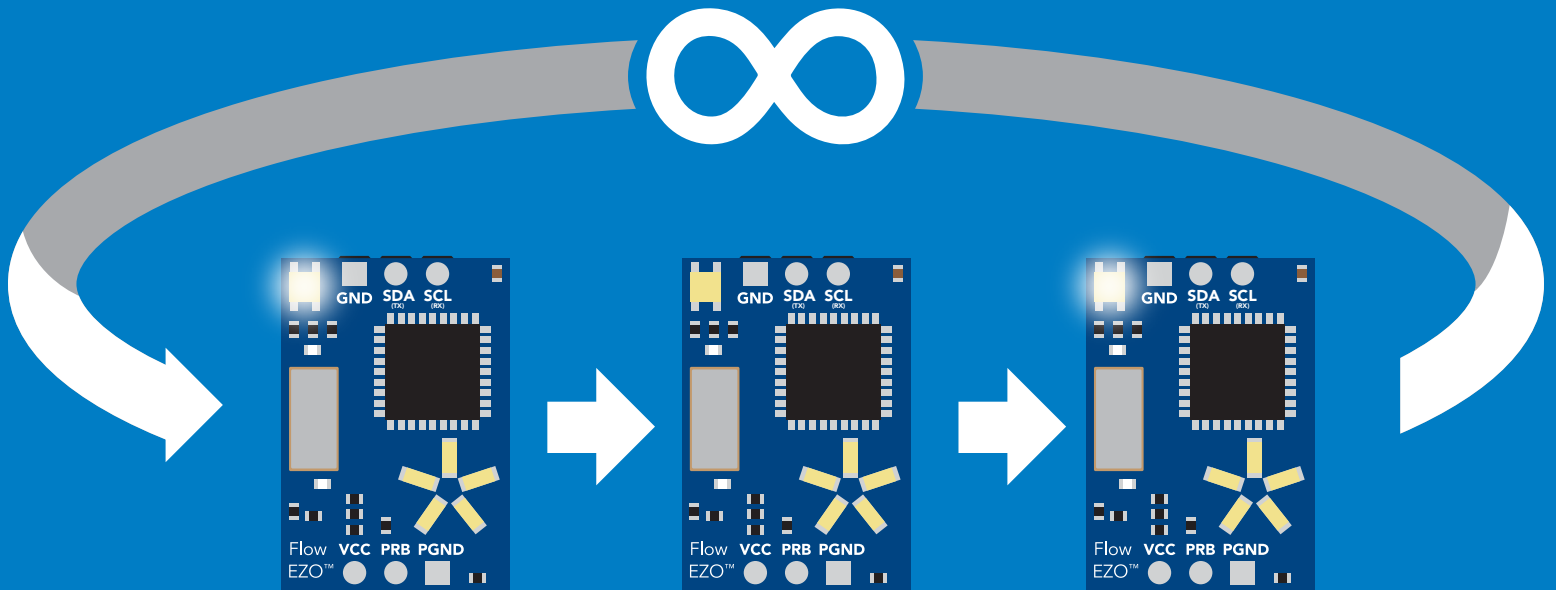
Find LED rapidly blinks white, used to help find device

## Example

## Response

Find <cr>

 Wait 300ms  
1 Dec 0 Null



# Taking reading

## Command syntax

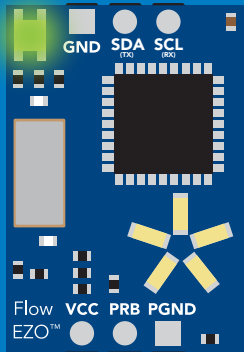
300ms  processing delay

R return 1 reading

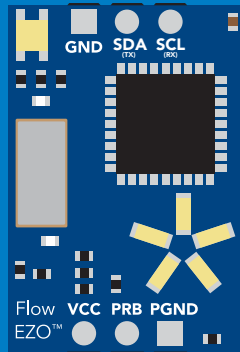
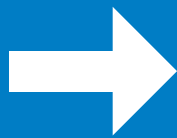
## Example

## Response

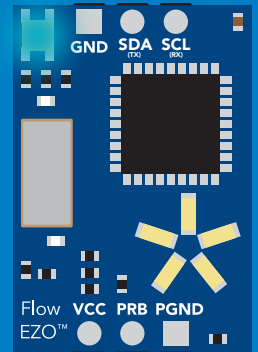
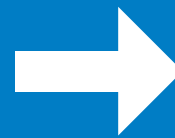
R  **1** **25.10** **0**  
Wait 300ms Dec ASCII Null



**Green**  
Taking reading



Transmitting



**Blue**  
Standby

# Clearing the total volume

Command syntax

300ms  processing delay

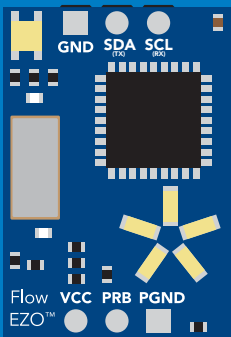
**Clear** clears the total volume, resets counter to 0.00

Example

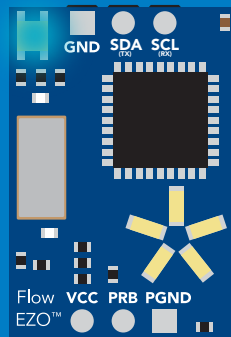
Response

Clear

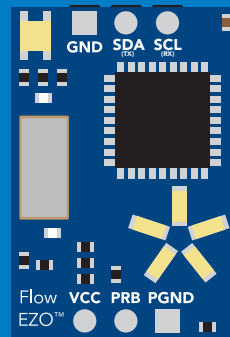
 Wait 300ms    **1** Dec    **0** Null



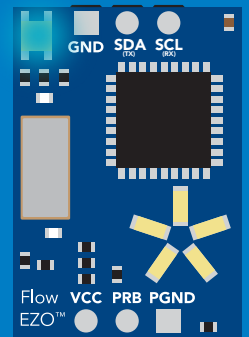
R



534.26



Clear



0.00

# Change flow rate display

This command changes the time base of the flow rate.

Total volume / flow rate



**Command syntax**

300ms  processing delay

Frp,s calculate flow rate per second

Frp,m calculate flow rate per minute

Frp,h calculate flow rate per hour

Frp,? calculate flow rate per?

## Example

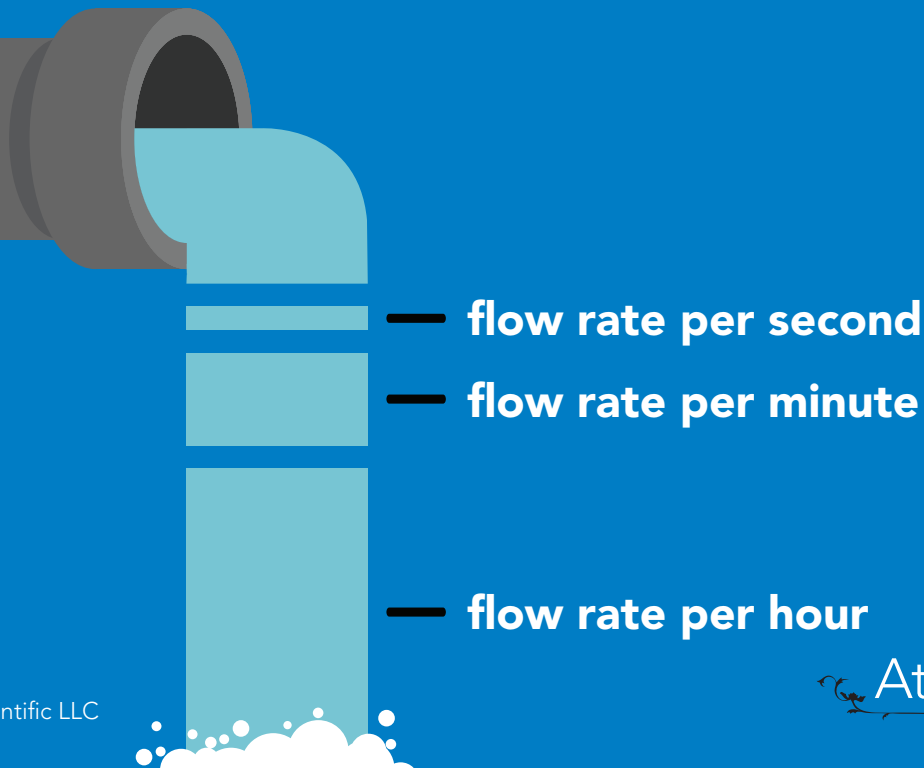
## Response

Frp,h

 Wait 300ms  
1 0  
Dec Null

Frp,?

 Wait 300ms  
1 ?Frp,h 0  
Dec ASCII Null





# Conversion factor

By default all readings are in L/LPM. The Conversion factor command lets you convert the readings to a different measurement.

Conversion factor range= 0.001 - 1,000,000

Liters x CF = converted reading

Example conversion factors:

Liters to milliliters =1,000

Liters to gallon = 0.264

300ms  processing delay

## Command syntax

default conversion factor = 1

CF,n set conversion factor  
CF,? show conversion factor

## Example Response

R  Wait 300ms **1** Dec **5.74 (liters)** ASCII **0** Null

CF, 0.264  Wait 300ms **1** Dec **0** Null

R  Wait 300ms **1** Dec **1.51 (gallons)** ASCII **0** Null

CF, ?  Wait 300ms **1** Dec **?CF,0.264** ASCII **0** Null

# Enable/disable parameters from output string

## Command syntax

300ms  processing delay

O, [parameter],[1,0]

enable or disable output parameter

O,?

enabled parameter?

## Example

## Response

O,TV,1 / O,TV,0



**1** **0**  
Dec Null

enable / disable total volume

O,FR,1 / O,FR,0



**1** **0**  
Dec Null

enable / disable flow rate

O,?



**1** **?O,TV,FR** **0**  
Dec ASCII Null

if both are enabled

### Parameters

TV total volume  
FR flow rate

### 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,



1 0  
Dec Null

name has been cleared

Name,zzt



1 0  
Dec Null

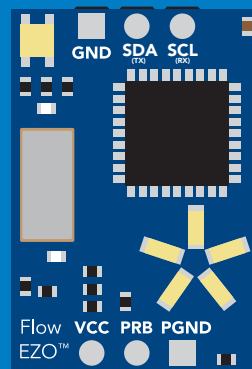
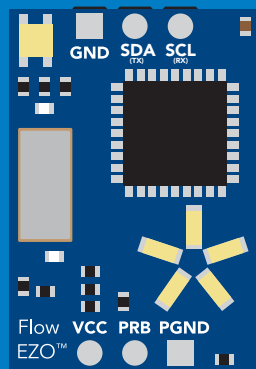
Name,?



1 ?Name,zzt 0  
Dec ASCII Null

Name,zzt

Name,?



1 0

1 ?Name,zzt 0

# Device information

Command syntax

300ms  processing delay

i device information

## Example

## Response

i



Wait 300ms

1

Dec

?i,FLO,2.00

ASCII

0

Null

## Response breakdown

?i, FLO, 2.00  
↑     ↑  
Device Firmware

# Reading device status

Command syntax

300ms  processing delay

Status voltage at Vcc pin and reason for last restart

## Example

## Response

Status

 **1** **?Status,P,5.038** **0**  
Wait 300ms Dec ASCII Null

## Response breakdown

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

### Restart codes

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

# Sleep mode/low power

## Command syntax

Sleep enter sleep mode/low power

Send any character or command to awaken device.

### Example

### Response

Sleep

no response

Do not read status byte after issuing sleep command.

Any command

wakes up device

5V

STANDBY

16.9 mA

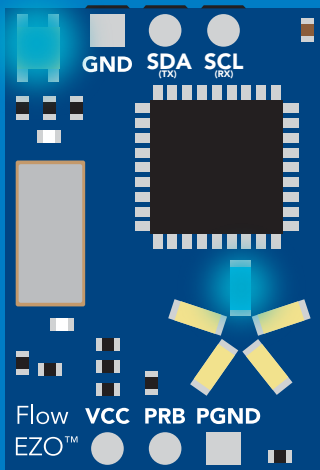
SLEEP

300µA

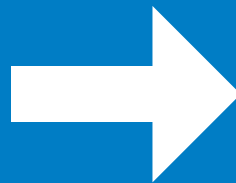
3.3V

13.2 mA

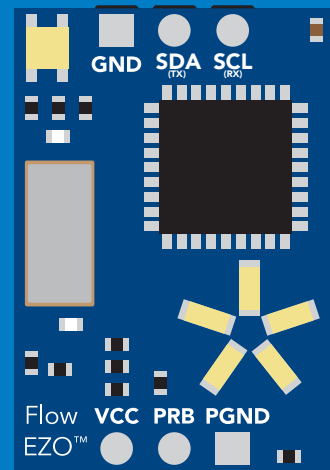
131µA



Standby



Sleep



Sleep

# Protocol lock

## Command syntax

300ms  processing delay

Plock,1 enable Plock

Plock,0 disable Plock

Plock,? Plock on/off?

default

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

## Example

## Response

Plock,1

  
Wait 300ms


1	0
Dec	Null

Plock,0

  
Wait 300ms

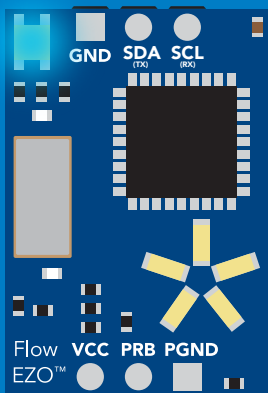
1	0
Dec	Null

Plock,?

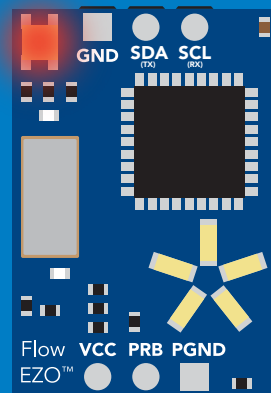
  
Wait 300ms

1	?Plock,1	0
Dec	ASCII	Null

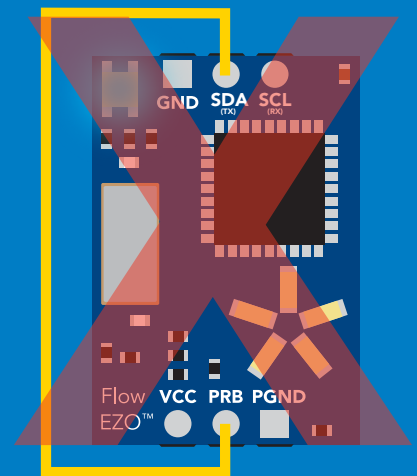
Plock,1



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

Response

I2C,100

device reboot  
(no response given)

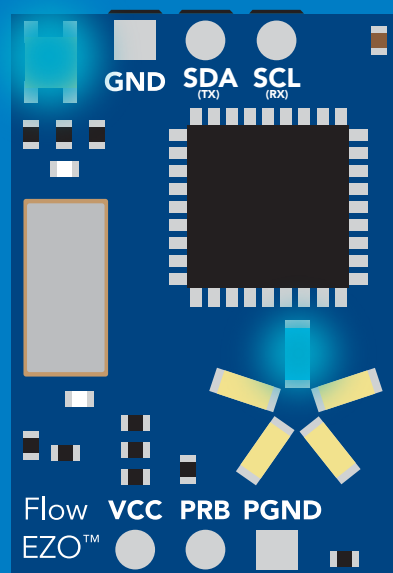
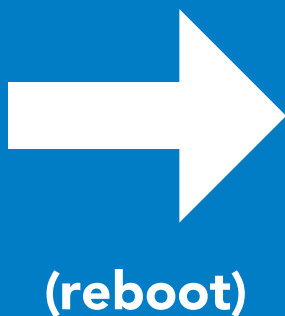
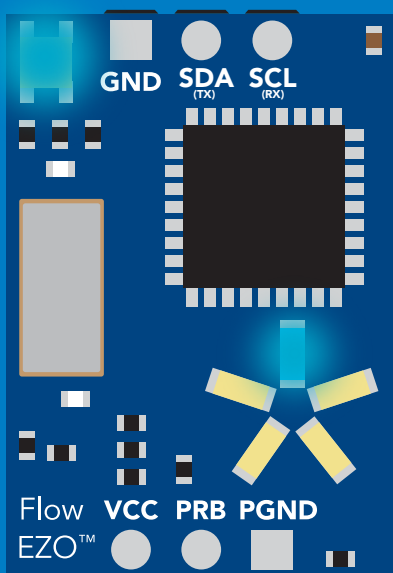
## Warning!

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

Default I<sup>2</sup>C address is 104 (0x68).

n = any number 1 – 127

I2C,100





# 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

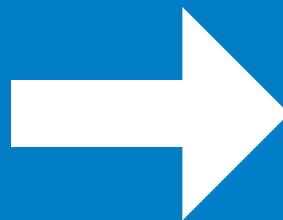
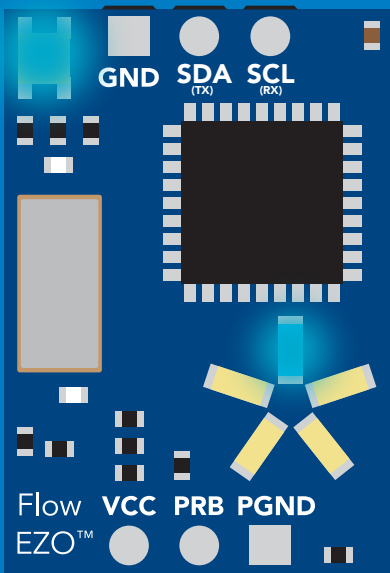
## Response

Factory

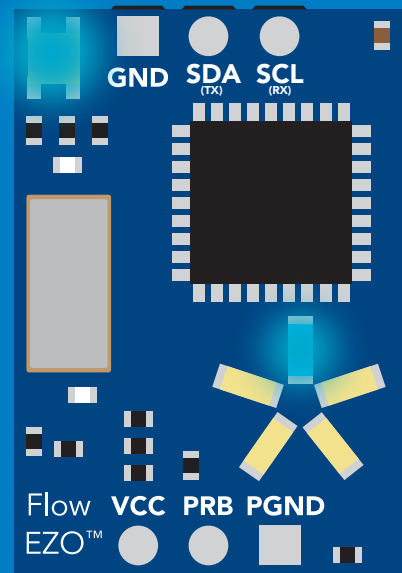
device reboot  
(no response given)

Clears all flow meter settings  
Total volume  
LED on

## Factory



(reboot)



# Change to UART mode

## Command syntax

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

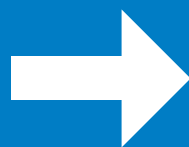
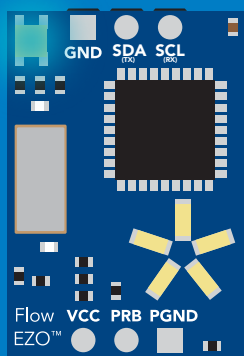
### Example

Baud,9600

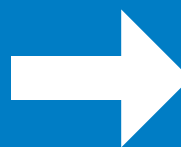
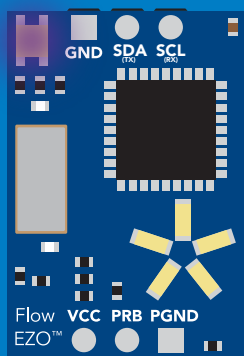
### Response

reboot in UART mode  
(no response given)

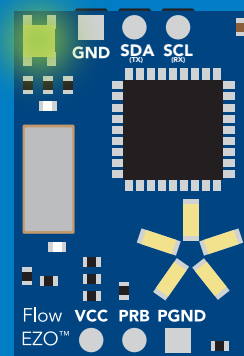
n = [ 300  
1200  
2400  
9600  
19200  
38400  
57600  
115200



Baud,9600



(reboot)



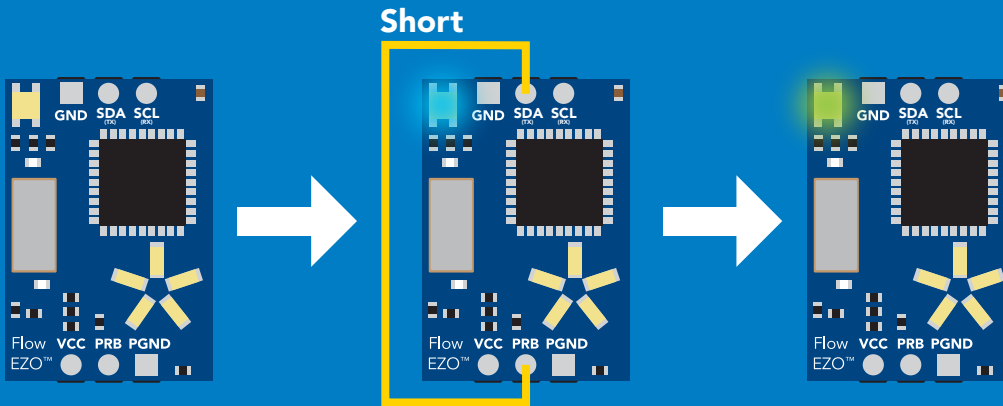
Changing to  
UART mode

# Manual switching to UART

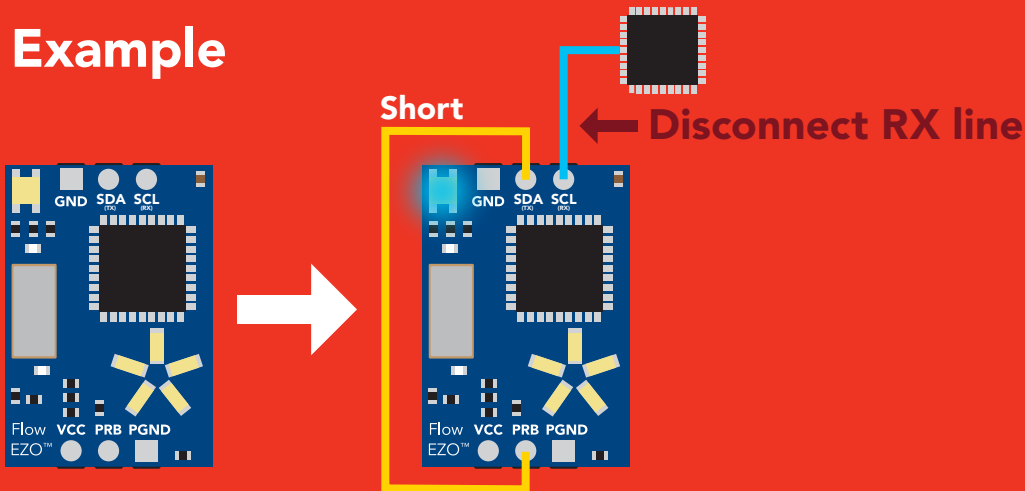
- Disconnect ground (power off)
- Disconnect TX and RX
- Connect TX to PRB
- Confirm RX is disconnected
- Connect ground (power on)
- Wait for LED to change from Blue to Green
- Disconnect ground (power off)
- Reconnect all data and power

Connecting TX to PRB only works for the EZO-RTD™ and the EZO-FLO™ circuits

## Example



## Wrong Example



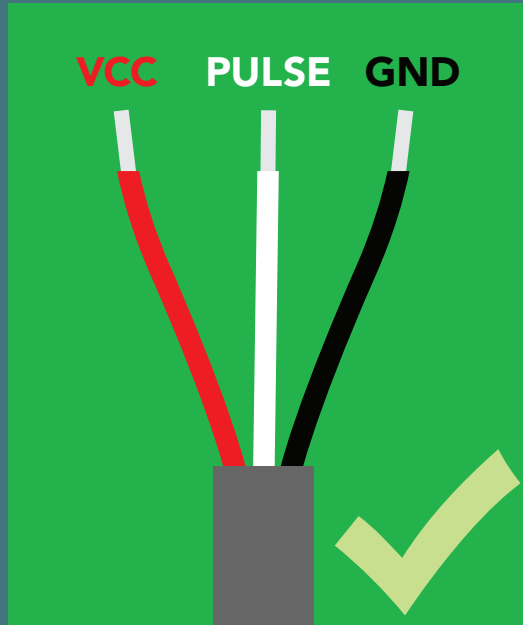
# Using your own flow meter

# Compatibility

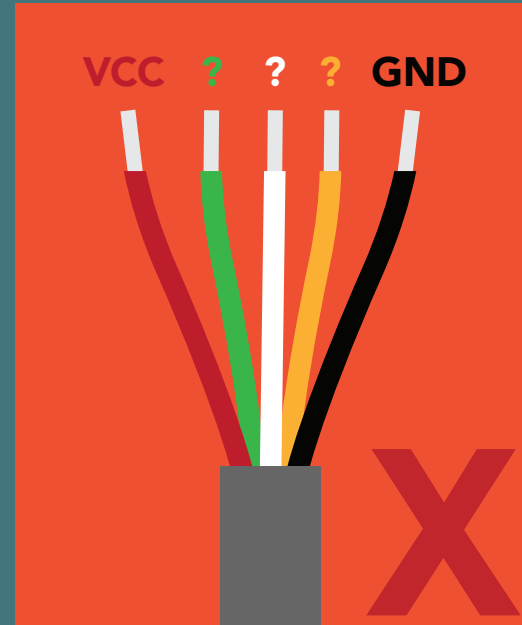
To be sure that your flow meter is compatible with the EZO-FLO™ it must meet **ALL** of the compatibility requirements listed below.

## Number of leads

3 leads

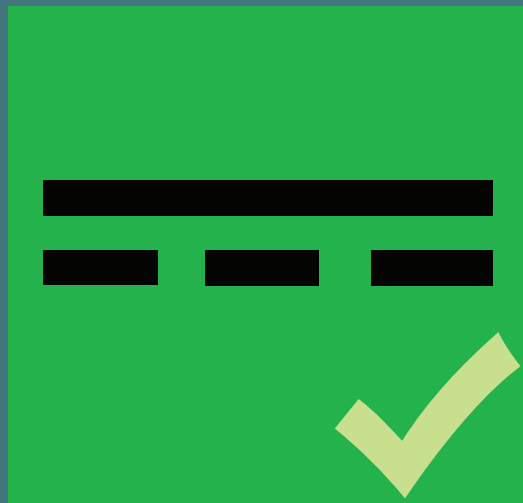


> 3 leads



## Operating voltage

3.3 – 5 VDC



AC voltage  
DC voltage > 5V



## Data output

Pulsed DC square wave only



Rising Edge = OK  
Falling Edge = OK

4–20mA

Mod Bus

RS-485



## Square wave frequency

0Hz – 8KHz



8+ KHz



## K Factor

Your flow meter must have at least 1 K factor, but no more than 16 K factors.

K Factor

$\geq 1$  &  $\leq 16$



K Factor

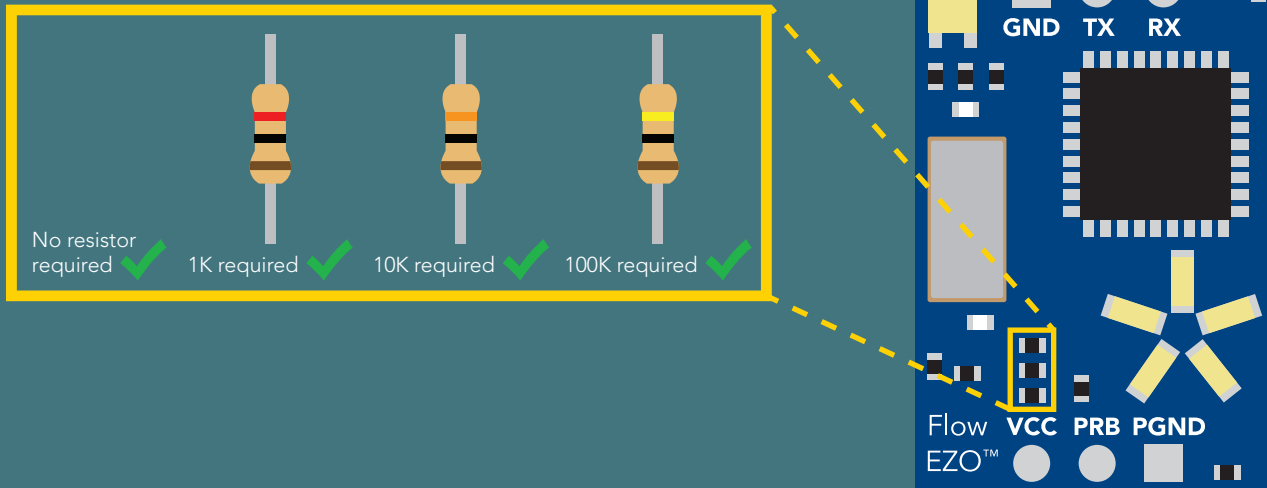
0 or  $> 16$



# Take notice

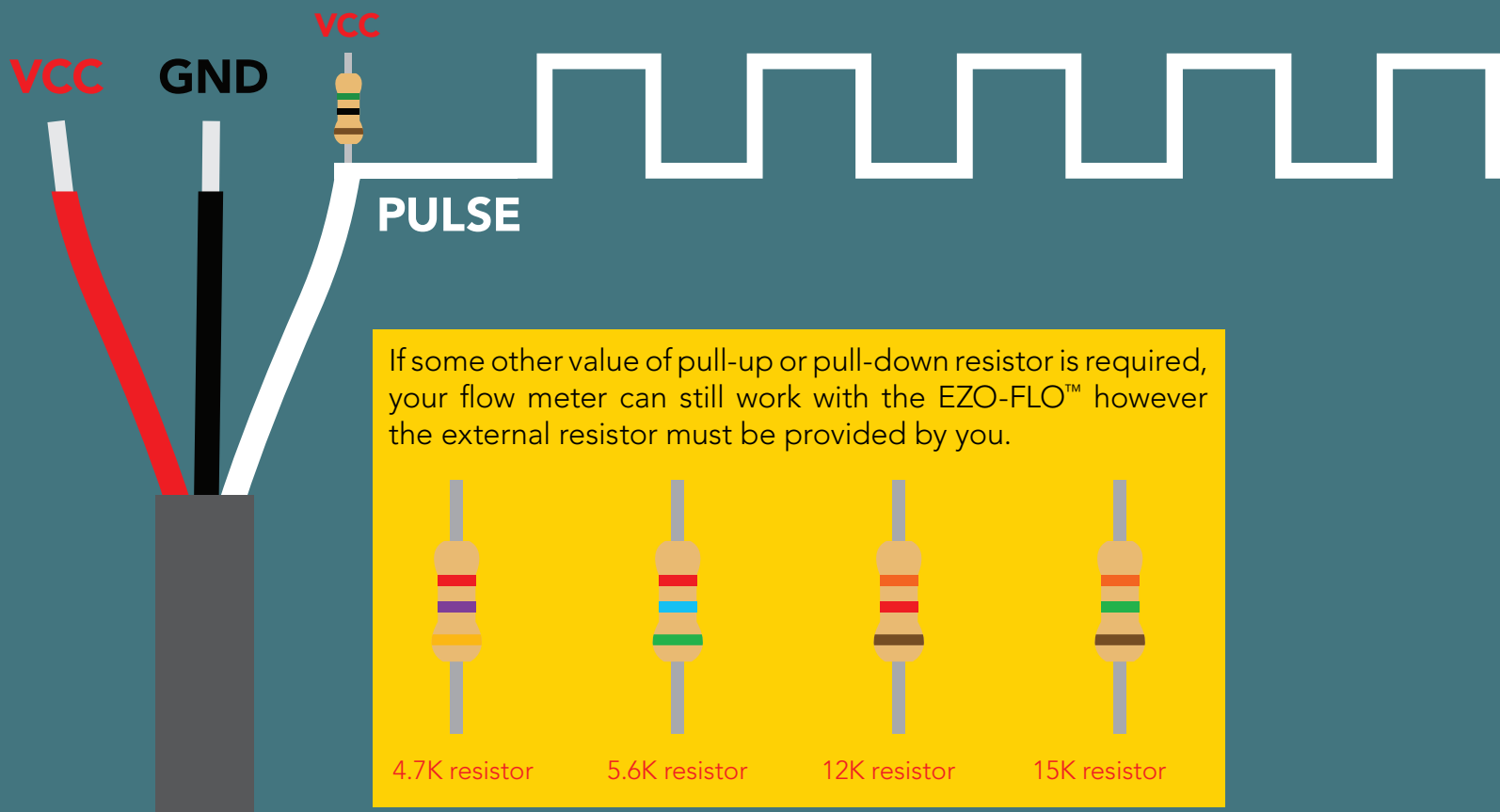
Some flow meters will require an external pull-up or pull-down resistor on the pulse lead. The EZO-FLO™ has 3 on-board pull-up or pull-down resistors available

## On-board resistors



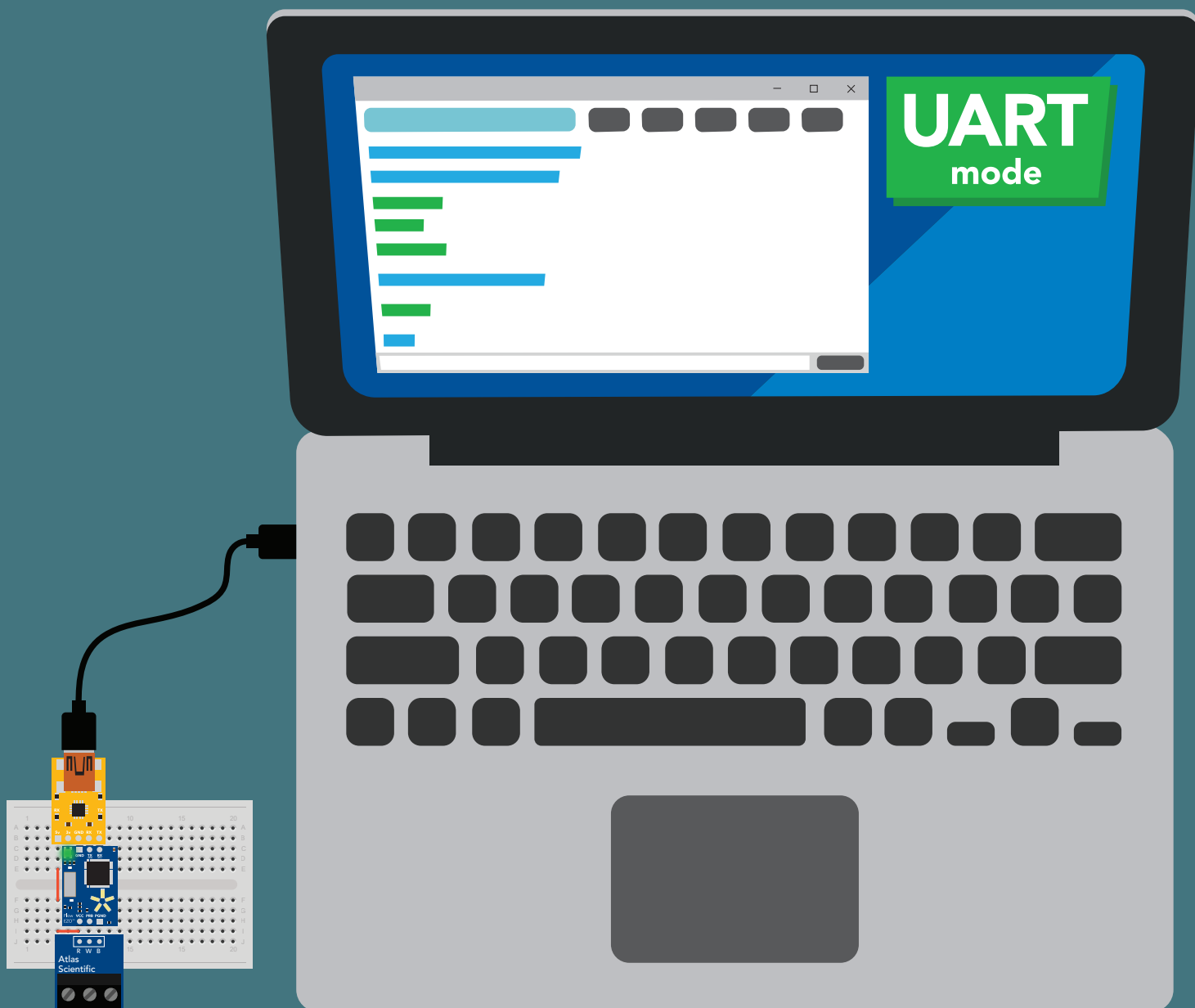
## External pull-up / pull-down resistor

Does your flow meter require a pull-up or pull-down resistor on the pulse lead?

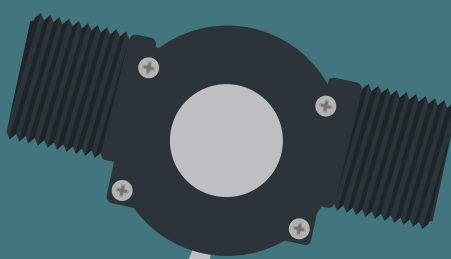


# Programming

Programming the EZO-FLO™ is easiest to do in **UART mode**, connected to a computer and programmed through a serial terminal.



*\*The EZO-FLO™ can be programmed in either UART or I2C mode, and can be re-programmed at any time.*





# Setting the K values

## Command syntax

<b>K</b> ,[volume],[per] number of pulses] <cr>	for flow meters with 1 K value
<b>K</b> ,[flow rate],[pulse rate in Hz] <cr>	for flow meters with many K values
<b>K</b> ,all <cr>	query the programmed K-value(s)
<b>K</b> ,clear <cr>	clear all programmed K-values

## Example

**K,10,1** <cr>

for flow meters with 1 K value  
(10mL / pulse)

## Response

**\*OK** <cr>

**K,0.1,13** <cr>

**\*OK** <cr>

**K,0.25,41** <cr>

**\*OK** <cr>

**K,0.5,90** <cr>

**\*OK** <cr>

...

Up to 16 in total

for flow meters with many K values  
(0.1 LPM @ 13Hz)  
(0.25 LPM @ 41Hz)  
(0.5 LPM @ 90Hz)

**K**,all <cr>

**?1:K,0.1,13** <cr>

**?2:K,0.25,41** <cr>

**?3:K,0.5,90** <cr>

**K**,clear <cr>

**\*OK** <cr>

# Setting the K values

I<sup>2</sup>C mode

## Command syntax

 300ms  processing delay

**K**,[volume],[(per) number of pulses]

for flow meters with 1 K value

**K**,[flow rate],[pulse rate in Hz]

for flow meters with many K values

**K**,n
returns the n<sup>th</sup> K value
**K**,?

returns the number of K values stored

**K**,clear

clear all programmed K-values

## Example

## Response

**K**,10,1

 for flow meters with 1 K value  
(10mL / pulse)


Wait 300ms

1

Dec

0

Null

**K**,0.1,13

 for flow meters with many K values  
(0.1 LPM @ 13Hz)


Wait 300ms

1

Dec

0

Null

**K**,1


Wait 300ms

1

Dec

?1:K,0.1,13

ASCII

0

Null

**K**,?


Wait 300ms

1

Dec

?K,1

ASCII

0

Null

**K**,clear


Wait 300ms

1

Dec

0

Null

# Setting the flow meter time base

This step is only needed for flow meters with multiple K values

In step one you programmed all the K values into the EZO-FLO. Now you have to set the time base.

0.1 LPM @ 13Hz

0.25 LPM @ 41Hz

0.5 LPM @ 90Hz

The 3 example K values above are in liters per min. The time base for these K values is in Liters per min. Use the command VP,M. If your K values were in Gallons per hour you would set the time base to VP,H.

## Command syntax

**Vp,s** <cr> set time base to volume per second

**Vp,m** <cr> set time base to volume per minute

**Vp,h** <cr> set time base to volume per hour

**Vp,?** <cr> set time base to volume per?

## Example

**Vp,h** <cr>

## Response

**\*OK** <cr>

**Vp,?** <cr>

**?Vp,h** <cr>

# Setting the flow meter time base

This step is only needed for flow meters with multiple K values

In step one you programmed all the K values into the EZO-FLO. Now you have to set the time base.

0.1 LPM @ 13Hz  
0.25 LPM @ 41Hz  
0.5 LPM @ 90Hz

The 3 example K values above are in liters per min. The time base for these K values is in Liters per min. Use the command VP,M. If your K values were in Gallons per hour you would set the time base to VP,H.

## Command syntax

300ms  processing delay

**Vp,s** set time base to volume per second

**Vp,m** set time base to volume per minute

**Vp,h** set time base to volume per hour

**Vp,?** set time base to volume per?

## Example

## Response

**Vp,h**

  
Wait 300ms

1	0
Dec	Null

**Vp,?**

  
Wait 300ms

1	?Vp,h	0
Dec	ASCII	Null

# Setting the onboard pull-up or pull-down resistors

This step is only needed if your flow meter requires an external pull-up or pull-down resistor on the pulse lead.

## Command syntax

**P,1** <cr> enable a 1K  $\Omega$  on board pull-up resistor  
**P,-1** <cr> enable a 1K  $\Omega$  on board pull-down resistor  
**P,10** <cr> enable a 10K  $\Omega$  on board pull-up resistor  
**P,-10** <cr> enable a 10K  $\Omega$  on board pull-down resistor  
**P,100** <cr> enable a 100K  $\Omega$  on board pull-up resistor  
**P,-100** <cr> enable a 100K  $\Omega$  on board pull-down resistor  
**P,0** <cr> disable the pull-up / pull-down resistor  
**P,?** <cr> query the pull-up / pull-down resistor

## Example

**P,10** <cr>

**\*OK** <cr>

**P,?** <cr>

**?P,10** <cr>

**\*OK** <cr>

# Setting the onboard pull-up or pull-down resistors

This step is only needed if your flow meter requires an external pull-up or pull-down resistor on the pulse lead.

## Command syntax

P,1	enable a 1K $\Omega$ on board pull-up resistor
P,-1	enable a 1K $\Omega$ on board pull-down resistor
P,10	enable a 10K $\Omega$ on board pull-up resistor
P,-10	enable a 10K $\Omega$ on board pull-down resistor
P,100	enable a 100K $\Omega$ on board pull-up resistor
P,-100	enable a 100K $\Omega$ on board pull-down resistor
P,0	disable the pull-up / pull-down resistor
P,?	query the pull-up / pull-down resistor

## Example

## Response

P,10



Wait 300ms

1

Dec

0

Null

P,?



Wait 300ms

1

Dec

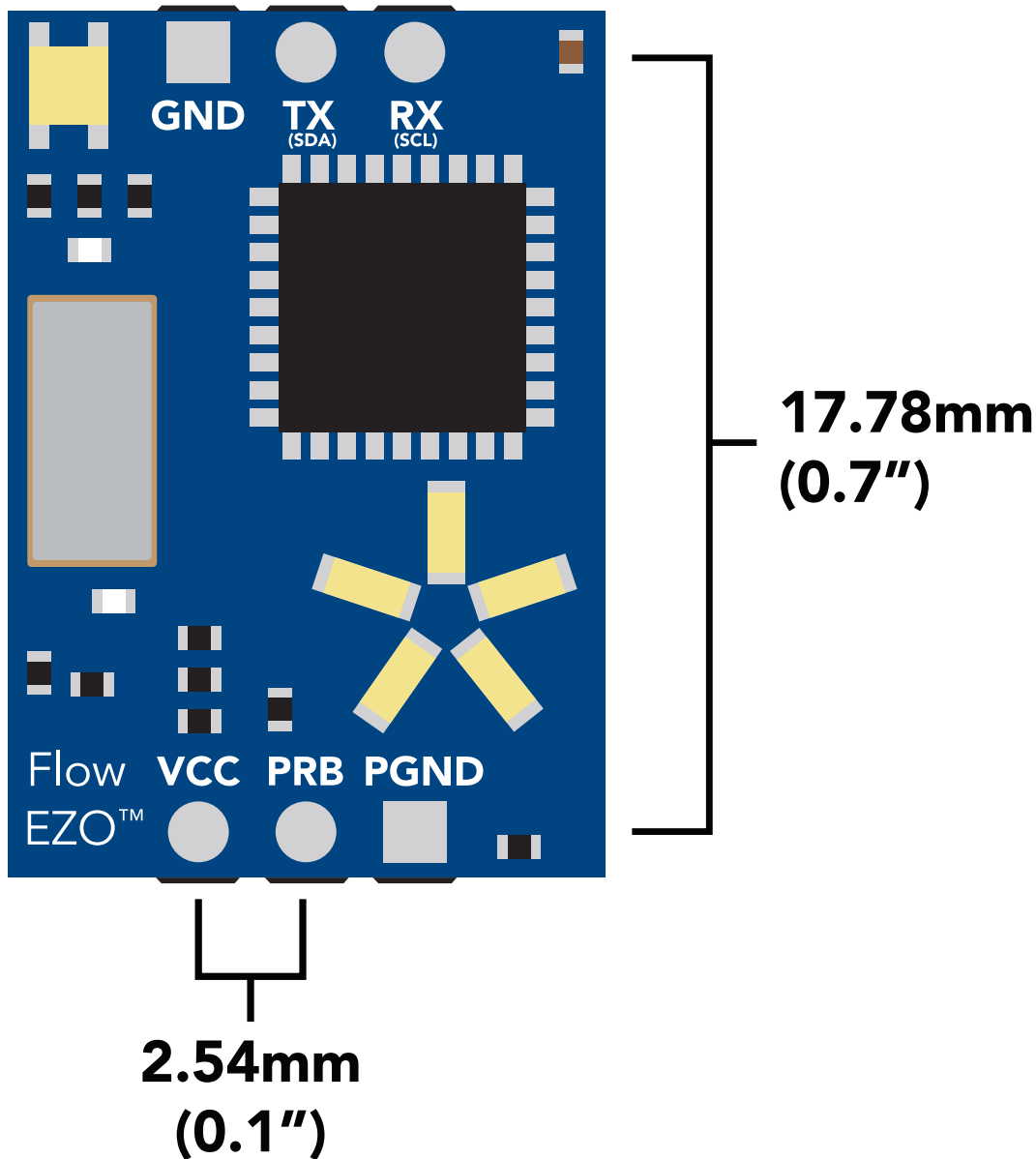
?P,10

ASCII

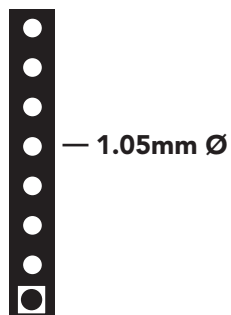
0

Null

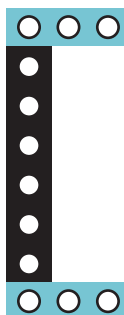
# EZO™ circuit footprint



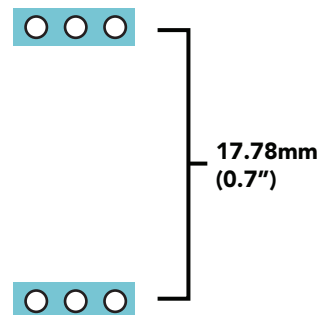
**1** In your CAD software place a 8 position header.



**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 2.6

Revised naming device info on pages 25 & 50.

## Datasheet V 2.5

Clarified default values on pages 7,11 and 36.  
Added the "conversion factor" command pages 23 and 48.

## Datasheet V 2.4

Added "Name device" command for I<sup>2</sup>C on pg 48.

## Datasheet V 2.3

Firmware update

## Datasheet V 2.2

Moved Default state to pg 8.

## Datasheet V 2.1

The 1/2" flow meter is now the default setting.

## Datasheet V 2.0

Revised entire datasheet.



# Firmware updates

v2.0 - (May 8, 2019)

- I<sup>2</sup>C mode enabled

v2.01 - (June 6, 2019)

- The 1/2" flow meter is now the default setting.
- Flow rate gets calculated every read command for better output at polling rates faster than 1 second.

v2.02 - (Nov 12, 2019)

- Changed the default pull-up resistor in 3/4" flow meter setting to 100k.

v2.03 - (Oct 8, 2020)

- Defined all readings to be in L/LMP by default.
- Added the "CF" command.
- Fixed bug where some readings only had one decimal, not two.

v2.04 - (Mar 26, 2021)

- Fixed bug where flow leds dont spin in I2C mode.

v2.05 - (Dec 12, 2021)

- Fixed bug where spinning LEDs would jump and glitch where the set,? response would erroneously contain a newline.

# Warranty

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

## The debugging phase

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

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