

Flow Meter API

Revision 1 - Software v2023.2.2

irw

Summary

The following includes a description of the available methods used to interact with the flow meter via Python. See the official TSI/Copley documentation for the command set.

Contents

TSI class	2
connect(self, port = None)	2
close(self)	2
set_output_dir(self, dir)	2
read(self)	2
query_connection(self, message = "?\r")	2
query_flow_set(self)	2
query_volume(self)	3
convert(self, byte = None)	3
setup_single(self)	3
convert_single(self)	3
convert_volume(self, byte = None)	3
save_flow_set(self, data)	3

TSI class

Instantiate a flow meter of the TSI class with the following:

```
meter = TSI(port = 'COM1') # Windows  
meter = TSI(port = '/dev/USB0') # Linux
```

The available methods are listed below, including an example and description of each.

connect(self, port = None)

```
meter.connect()
```

Connect to the flow meter at the specified port. If no port is specified, attempt to connect on the default port supplied when creating the TSI object.

If successful, sets the 'dev' instance variable from the serial connection.

close(self)

```
meter.close()
```

Closes the connection to the flow meter.

set_output_dir(self, dir)

```
meter.set_output_dir(r'./data')
```

Sets the path for saving data. Creates the directory if it does not exist.

read(self)

```
meter.read()
```

Reads and returns a single line from the serial connection to the flow meter.

query_connection(self, message = "?\r")

```
meter.query_connection()
```

Default behavior: tests the serial connection to the flow meter.

If a message is supplied, sends the message to the flow meter.

Returns the flow meter response after reading all lines.

query_flow_set(self)

```
meter.query_flow_set()
```

Requests read of 1000 flow data points in binary format. Waits for the confirmation response and then reads, converts, and saves the return values as a CSV.

query_volume(self)

`meter.query_volume()`

Requests a volume measurement over 1000 data points.

Implicitly prints the measured volume.

convert(self, byte = None)

`meter.convert(self.dev.read(1))`

Internal

Converts the bytes received from the flow meter to values of flow rate in the set units (typically SLPM).

setup_single(self)

`meter.setup_single()`

Internal

Requests read of 1000 flow points. Returns **True** if request successful.

convert_single(self)

`meter.convert_single()`

Internal

Used for live plotting.

convert_volume(self, byte = None)

`meter.convert_volume()`

Internal

Processes the request from `query_volume()`

save_flow_set(self, data)

`meter.save_flow_set(results)`

Internal

Saves supplied data to a timestamp-named csv file in the configured output directory.

The saved format is of a Pandas dataframe, using the `to_csv()` method.
