
openenergymonitor_thermostat*Documentation*

Release 0.0.1

Stuart Mumford

Jan 24, 2017

Contents

1	HTTP API	3
2	oemthermostat Package	5
2.1	Classes	5
2.2	Class Inheritance Diagram	7
	HTTP Routing Table	9
	Python Module Index	11

This module provides a Python API to the Open Energy Monitor [Thermostat](#). Currently it only provides enough functionality for external control of the thermostat rather than providing access to all the configuration options.

This package implements one class *oemthermostat.Thermostat* which provides properties and methods to control the device. Some simple examples are below:

```
>>> from oemthermostat import Thermostat
>>> t = thermostat('192.168.0.1')
>>> t.setpoint
21.5
>>> t.setpoint = 18.6
>>> t.state
False
>>> t.switch()
>>> t.state
True
```

Contents:

CHAPTER 1

HTTP API

The following documents the HTTP API of the thermostat / relay device. This was discovered by reading the original source code for the ESP8266 and using the dev tools in Firefox to inspect the calls in the web interface.

Note: This is incomplete, I will add more as I research it.

GET `/control/thermostat.cgi?param=state`

Get the status of the thermostat.

Parameters

- **param=state** – Request state of the thermostat.

Example request:

```
GET /control/thermostat.cgi?param=state HTTP/1.1
Host: example.com
Accept: application/json, text/javascript
```

Example response:

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "temperature": "22.81",
  "humidity": "N/A",
  "humidistat": 0,
  "relay1state": 0,
  "relay1name": "Heating",
  "state": 2,
  "manualsetpoint": 1900,
  "heat_cool": 0
}
```

Status Codes

- 200 OK – no error

POST /control/thermostat.cgi?param=thermostat_state

Set operation mode of the thermostat.

Parameters

- **param=thermostat_state** – Set thermostat operation mode.

Form 0 - off, 1 - schedule, 2 - manual

POST /control/thermostat.cgi?param=thermostat_manualsetpoint

Set target temperature of thermostat.

Form Parameters

- **int** – Temperature in 1/100 C.

POST /control/thermostat.cgi?param=thermostat_heat_cool

Form 0 - heating, 1 - cooling

POST /control/thermostat.cgi?param=thermostat_schedule

Set scheduled setpoint.

Example request:

```
POST /control/thermostat.cgi?param=thermostat_schedule HTTP/1.1
Accept: application/json

{ "mon":
  [
    { "s": 0,
      "e": 2400,
      "sp": 2100
    }
  ]
}
```

GET /control/relay.cgi?relay1=(int: state)

Change the current state of the relay.

Parameters

- **relay1** – 0 - off, 1 - on

oemthermostat Package

2.1 Classes

<i>Thermostat</i> (host[, port, username, password])	A class for interacting with the OpenEnergyMonitor Thermostat's HTTP API.
--	---

2.1.1 Thermostat

class oemthermostat.**Thermostat** (*host, port=80, username=None, password=None*)

A class for interacting with the OpenEnergyMonitor Thermostat's HTTP API.

Parameters

- **host** (*str*) – Hostname or IP address.
- **port** (*int* (optional)) – The port of the web server.
- **username** (*str* (optional)) – The username for HTTP auth.
- **password** (*str* (optional)) – The password for the HTTP auth.

Attributes Summary

<i>mode</i>	Returns the current mode of the thermostat.
<i>setpoint</i>	Current thermostat setpoint in C
<i>state</i>	Current state of the relay.
<i>temperature</i>	Current value of the temperature sensor in C

Methods Summary

<code>get(endpoint, **kwargs)</code>	Perform a GET request
<code>post(endpoint, **kwargs)</code>	Perform a POST request
<code>status()</code>	Get the status dictionary from the thermostat
<code>switch()</code>	Change the state of the relay.

Attributes Documentation

mode

Returns the current mode of the thermostat.

Returns mode – 0 for manual mode, 1 for schedule mode and 2 for manual mode.

Return type *int*

setpoint

Current thermostat setpoint in C

state

Current state of the relay.

Returns state – returns *True* if the relay is on and *False* if the relay is off.

Return type *bool*

temperature

Current value of the temperature sensor in C

Methods Documentation

get (*endpoint*, ***kwargs*)

Perform a GET request

Parameters

- **endpoint** (*str*) – The endpoint to send the request to, will have ‘cgi’ appended to it.
- **kwargs** (*dict*) – All other kwargs are passed to *requests.get*

Returns response – The result of the request.

Return type *requests.Response*

post (*endpoint*, ***kwargs*)

Perform a POST request

Parameters

- **endpoint** (*str*) – The endpoint to send the request to, will have ‘cgi’ appended to it.
- **kwargs** (*dict*) – All other kwargs are passed to *requests.post*

Returns response – The result of the request.

Return type *requests.Response*

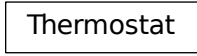
status ()

Get the status dictionary from the thermostat

switch ()

Change the state of the relay.

2.2 Class Inheritance Diagram



HTTP Routing Table

/control

```
GET /control/relay.cgi?relay1=(int:state),
    4
GET /control/thermostat.cgi?param=state,
    3
POST /control/thermostat.cgi?param=thermostat_heat_cool,
    4
POST /control/thermostat.cgi?param=thermostat_manualsetpoint,
    4
POST /control/thermostat.cgi?param=thermostat_schedule,
    4
POST /control/thermostat.cgi?param=thermostat_state,
    4
```


O

oemthermostat, 5

G

`get()` (oemthermostat.Thermostat method), 6

M

`mode` (oemthermostat.Thermostat attribute), 6

O

oemthermostat (module), 5

P

`post()` (oemthermostat.Thermostat method), 6

S

`setpoint` (oemthermostat.Thermostat attribute), 6

`state` (oemthermostat.Thermostat attribute), 6

`status()` (oemthermostat.Thermostat method), 6

`switch()` (oemthermostat.Thermostat method), 6

T

`temperature` (oemthermostat.Thermostat attribute), 6

Thermostat (class in oemthermostat), 5