Sensor package provides drivers for various sensors.

Temperature

Summary

Temperature interface provides functions to read temperature.

Supported devices

- DS18* (OneWireTemp.html) (1-Wire)
- TMPXXX (TMPXXX.html) (I2C) : TMP75, TMP102, TMP275
- BMP085 (BMP085.html) (I2C)

Methods list

getKelvin()

Returns the temperature in kelvin.

REST API : GET /devices/**name**/sensor/temperature/k

- name : device name from configuration file

getCelsius()

Returns the temperature in celsius.

REST API : GET /devices/**name**/sensor/temperature/c

- name : device name from configuration file

¶ getFahrenheit()

Returns the temperature in fahrenheit.

REST API : GET /devices/**name**/sensor/temperature/f

- name : device name from configuration file

Python example

```
from webiopi.devices.analog import TMP102
tmp = TMP102(...)           # setup a TMP102 I2C Temperature sensor
# or
from webiopi import deviceInstance
tmp = deviceInstance("tmp") # retrieve device named "tmp" in configuration file

tmp.getKelvin()
tmp.getCelsius()
tmp.getFahrenheit()
```

REST example

```
HTTP GET /devices/tmp/sensor/temperature/k
HTTP GET /devices/tmp/sensor/temperature/c
HTTP GET /devices/tmp/sensor/temperature/f
```

Pressure

Summary

Pressure interface provides functions to read atmospheric pressure.

Supported devices

- BMP085 (BMP085.html) (I2C)

Methods list

getPascal()

Returns the pressure in pascal.

REST API : GET /devices/**name**/sensor/pressure/pa

- name : device name from configuration file

getHectoPascal()

Returns the pressure in hecto pascal.

REST API : GET /devices/**name**/sensor/pressure/hpa

- name : device name from configuration file

getPascalAtSea()

Returns the pressure at sea level in pascal.

REST API : GET /devices/**name**/sensor/pressure/sea/pa

- name : device name from configuration file

getHectoPascalAtSea()

Returns the pressure at sea level in hecto pascal.

REST API : GET /devices/**name**/sensor/pressure/sea/hpa

- name : device name from configuration file

Python example

```
from webiopi.devices.analog import BMP085
bmp = BMP085(...)           # setup a BMP085 I2C Pressure sensor
# or
from webiopi import deviceInstance
bmp = deviceInstance("bmp") # retrieve device named "bmp" in configuration file

bmp.getHectoPascal()
bmp.getHectoPascalAtSea()
```

REST example

```
HTTP GET /devices/bmp/sensor/pressure/hpa
HTTP GET /devices/bmp/sensor/pressure/sea/hpa
```

Luminosity

Summary

Luminosity interface provides functions to measure light.

Supported devices

- TSL2561 (TSL2561.html) (I2C)
- TSL4531 (TSL4531.html) (I2C)
- VCNL4000 (VCNL4000.html) (I2C)

Methods list

getLux()

Returns the luminosity in lux.

REST API : GET /devices/**name**/sensor/luminosity/lx

- name : device name from configuration file

Python example

```
from webiopi.devices.analog import TSL2561
tsl = TSL2561(...)          # setup a TSL2561 I2C Luminosity sensor
# or
from webiopi import deviceInstance
tsl = deviceInstance("tsl") # retrieve device named "tsl" in configuration file

tsl.getLux()
```

REST example

```
HTTP GET /devices/tmp/sensor/luminosity/lx
```

Distance

Summary

Distance interface provides functions to measure distances.

Supported devices

- VCNL4000 (VCNL4000.html) (I2C)

Methods list

getMillimeter()

Returns the distance in millimeter.

REST API : GET /devices/**name**/sensor/distance/mm

- name : device name from configuration file

getCentimeter()

Returns the distance in millimeter.

REST API : GET /devices/**name**/sensor/distance/cm

- name : device name from configuration file

getInch()

Returns the distance in millimeter.

REST API : GET /devices/**name**/sensor/distance/in

- name : device name from configuration file

Python example

```
from webiopi.devices.analog import TSL2561
vcn = VCNL4000(...)         # setup a VCNL4000 I2C Luminosity sensor
# or
from webiopi import deviceInstance
vcn = deviceInstance("vcn") # retrieve device named "vcn" in configuration file

vcn.getMillimeter()
```

REST example

```
HTTP GET /devices/tmp/sensor/distance/mm
```