POET PH, ORP, EC & Temperature over I2C

version: v1.0.0-10-g4bb3221

Sentron Europe B.V.

2024-09-30 14:15:35 +0200

Contents

POET PH, ORP, EC & Temperature over I2C. This innovative sensor brings together ISFET pH sensing, 4-electrode EC (Electrical Conductivity) measurement, and ORP analysis in a single device. The POET features a 2-wire, I2C-compatible interface that supports I2C bus speeds up to 400 khz.

1 Electrical characteristics

1.1 Pin Functions

Pin	Wire color	Description
VCC	Yellow	Supply voltage
GND	Brown	Ground
SDA	White	Serial data
SCL	Green	Serial clock

1.2 VCC

Parameter	Value
Supply voltage (min)	3.3V

Sentron Europe B.V.
Kamerlingh-Onnesstraat 5
9351 VD Leek
The Netherlands
+31 (0)50 5013 800
info@sentron.nl | www.sentron.nl

Millar Headquarters
11950 N. Spectrum Blvd
Pearland, TX, USA
+1 832 667 7000
www.millar.com

Parameter	Value
Supply voltage (recommended)	3.3V
Supply voltage (max)	5.5V
Input current (idle)	2.2 mA
Input current (measurement)	7.0 mA

1.3 SDA & SCL

The SDA and SCL signals have internal 100kOhm pull-ups to the internal 3.0V reference. For optimal performance, stronger, external pull-ups are recommended. The signals have an absolute maximum input voltage rating of 3.3V, beyond which a leakage current through the protective diodes will occur.

Parameter	Value
Low input level (max)	0.75V
High input level (min)	2.1V
High input level (max)	3.3V
Frequency (max)	400 khz
Protective diode current (max)	±2 mA
Drive strength (max)	6.0 mA

When operating at maximum drive strength the low level output voltage will only be pulled down to approximately 0.5V.

1.4 Galvanic isolation

The POET by itself provides no means of galvanic isolation. It is up to the user to provide adequate isolation from both the grid as well as other equipment placed in the same medium.

2 Device operation

The POET implements an i2c compatible slave interface using 7 bit addressing. When processing a request, further requests are ignored until the command has been handled. The device may use clock stretching during its communications. Device addresses are fixed and cannot be changed:

- **Address 0x1F** This is the main address. Used for receiving commands and transmitting the measurement data.
- **Address 0x50** This address is reserved for internal use.

2.1 Programming

The POET only starts a measurement after receiving a command byte on its main address 0x1F. When a stop condition occurs the last received command byte will be executed. Individual measurements can be selected and combined by setting their corresponding bits in the command byte:

bit(s)	Description
0	Set to 1 for temperature measurement
1	Set to 1 for ORP measurement
2	Set to 1 for pH measurement
3	Set to 1 for EC measurement
4:7	Reserved, set to 0

After the command byte has been latched by the stop condition, the following measurement times need to be observed:

Measurement	Time (ms)
base delay	100 ms
temperature	384 ms
ORP	1664 ms
Ph	384 ms
EC	256 ms

Thus, before reading the result of a command performing *all* measurements, one would have to wait `100 + 384 + 1664 + 384 + 256 = 2788 ms`. After the delay time has passed, reading from the main address (`0x1f`) results in a reply of variable length. The total length of the reply is determined by the individual measurements selected. All of the fields returned take the form of a `2s complement little endian 32 bit integer`.

Measurement	# bytes	Returns	Description
temperature	4	`temp_mC`	The temperature in milli-degrees celsius
ORP	4	`orp_uV`	The ORP value in micro-volts
Ph	4	`ugs_uV`	The pH gate-source potential in micro-volts
EC	8	`{ec_nA, ec_uV}`	Kelvin measurement of the EC, with the sensor current in nano-Amps and excitation in micro-volts

When performing multiple measurements at once, the order in which the fields are returned is the same as the above table (temperature first, EC last).

3 EC calculation from EC sensor current and excitation

The EC measurement is performed as a kelvin four-terminal measurement, thus from the above example we can calculate a resistance of `66000 uV / 66000 nA = 1000 Ohm`. Calibration in a solution of known conductivity is necessary to determine the probe specific cell constant.

Assuming the previous measurement was performed in a `1.41mS/cm@25°C` `0.01M` KCl solution would therefore result in a cell constant of `(1000 Ohm) * (1.41mS/cm)) = 1.41 / cm`.

And going the other way, a resistance of `1000 Ohm` and a cell constant of `1.41 /cm` would give us a conductivity of: `(1.41 / cm) / (1000 Ohm) = 1.41 mS/cm`.

4 pH calculation from Ugs

The pH sensors (`Ugs`) has a sensitivity of approximately `52 mV/pH`. To determine the sensors offset (`mV`), a measurement in a known buffer solution needs to be performed. The pH value of an actual measurement can then be calculated as follows:

```
sample_pH = buffer_pH + (sample_Ugs_mV - buffer_Ugs_mV) / 52
```

- ***note*** The exact offset and sensitivity are unique to each probe and may drift over time.

- ***note*** A second calibration point can be acquired to determine the sensitivity.
- ***note*** Even more calibration points can be used to perform either piecewise linear interpolation, or a more advanced method.

5 USB adapter

One can easily interact with the POET through a personal computer and the python programming language by utilizing an (optionally provided) FT232H based USB adapter cable.

5.1 requirements and setup

- First make sure python3 is available on your system. https://www.python.org/
  - Make sure it is added to the PATH during the setup.
- on windows change the ftdi adapters driver to `libusbk` with the zadig tool. https://zadig.akeo.ie/
- add the pyftdi package to your python installation https://eblot.github.io/pyftdi/

5.1.1 bash

For the initial setup execute the following:

```
# setup and activate a virtual environment
python -m venv .venv
source ./.venv/bin/activate
# install requirements in the virtual env
pip install pyftdi
# run with default arguments
python ./poet.py
```

Only the activation of the virtual environment and actual running of the file are necessary the next time:

```
source ./.venv/bin/activate
python ./poet.py
```

5.1.2 powershell

For the initial setup execute the following:

```
# setup and activate a virtual environment
python -m venv .venv
. ./.venv/Scripts/Activate.ps1
# install requirements in the virtual env
pip install pyftdi
# run with default arguments
python ./poet.py
```

If there is an error activating the environment saying `running scripts is disabled on this system`, then make sure running scripts is allowed by setting either:

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser
```

or

```
Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser
```

Only the activation of the virtual environment and actual running of the file are necessary the next time:

```
. ./.venv/Scripts/Activate.ps1
python ./poet.py
```

5.2 poet.py

```
import struct
from dataclasses import dataclass
from pyftdi.i2c import I2cController
import time

@dataclass
class POETResult:
  temp_mC : int
  orp_uV  : int
  ugs_uV  : int
  ec_nA   : int
  ec_uV   : int

i2c = I2cController()
# get the first ft232h enumerated by the system
i2c.configure("ftdi:///")

# if the Sentron cable with isolation transformer is used,
# an active low enable switch is connected to the ADBUS3 pin
gpio = i2c.get_gpio()
# set ADBUS3 as an output
gpio.set_direction(1 << 3, 1 << 3)
# and set it low
gpio.write(0)
# End of the sentron cable specific code

poet = i2c.get_port(0x1F)
# perform all 4 measurements:
poet.write([0b00001111])
# wait for all measurements to complete
time.sleep(2.788)
# get the 5 int32_t fields from the POET
rawReply = poet.read(20)

result = POETResult(*struct.unpack("<lllll", rawReply))

# if the sentron cable is used, cut the power after a measurement
gpio.write(1 << 3)
# End of the sentron cable specific code

print(result)
```

When everything has been set up correctly, a measurement in air at room temperature should return values close to the following result:

```
python ./poet.py
POETResult(temp_mC=20803, orp_uV=-3000000, ugs_uV=2999908, ec_nA=2, ec_uV=1499954)
```

Sentron Europe B.V.
Kamerlingh-Onnesstraat 5
9351 VD Leek
The Netherlands
+31 (0)50 5013 800
info@sentron.nl | www.sentron.nl

Millar Headquarters
11950 N. Spectrum Blvd
Pearland, TX, USA
+1 832 667 7000
www.millar.com