Environmental monitoring beyond plain text files

Étienne Wodey

geo-Q CTP
Hannover, 17 August 2016

HITec

RTG 1729

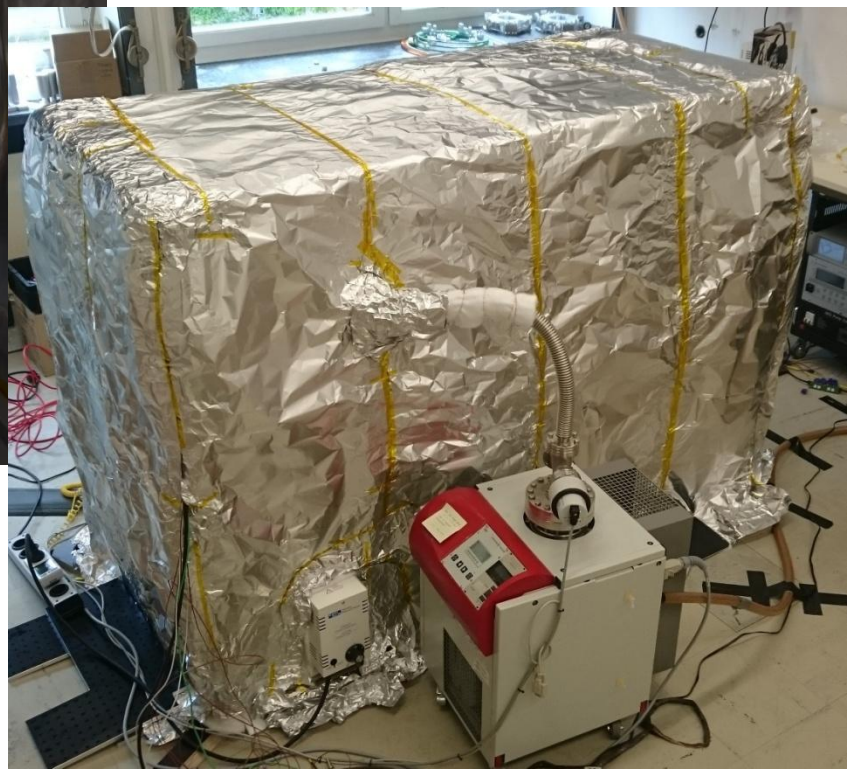
DFG Deutsche Forschungsgemeinschaft

Institute of Quantum Optics
University of Hannover

quest

The task

Monitor the bakeout of our UHV chamber (sensor head for VLBAI)

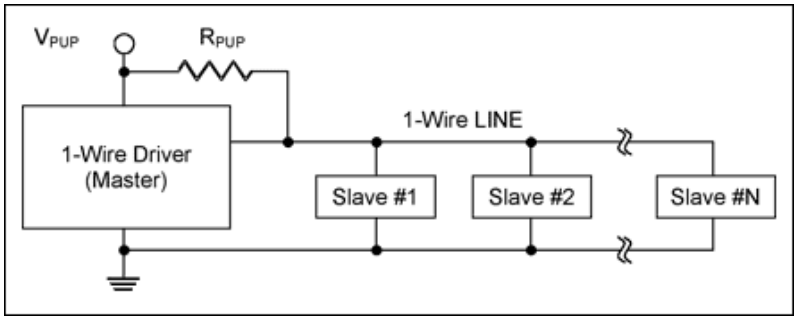


- 2 turbo pumping stations
- 3 ion pumps
- 2 NEG (passive getter) pumps
- 3 pressure gauges
- 24 temperature sensors (type E thermocouples)
- room temperature is also interesting

Getting everything digital: thermocouple readout

Bottom line: unless someone gives me a good reason not to, you should use the **1-Wire® bus** for scalable, non time-critical applications

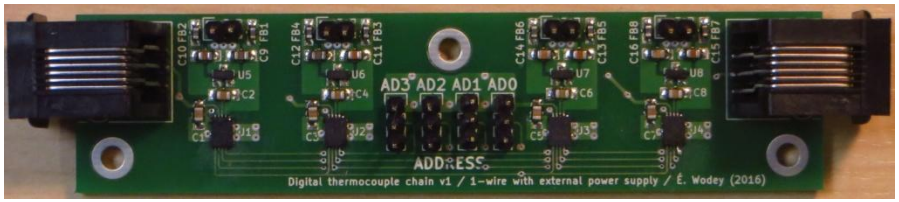
Good reasons:
- O(1) in line numbers (compared to SPI which is O(number of sensors))
- no clock (very long distances)
- many device families supported by the linux kernel (see `drivers/w1/slaves`)



https://www.maximintegrated.com

Drawbacks:
- no clock (rather slow)
- device address (serial number) factory set
- in principle proprietary (Dallas Semiconductor Corp. / Maxim Integrated)

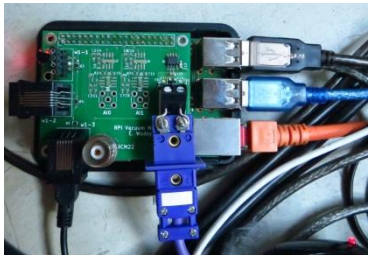
Home-made 4-fold MAX31850 thermocouple to digital board on telephone cable line (RJ12/6P6C)



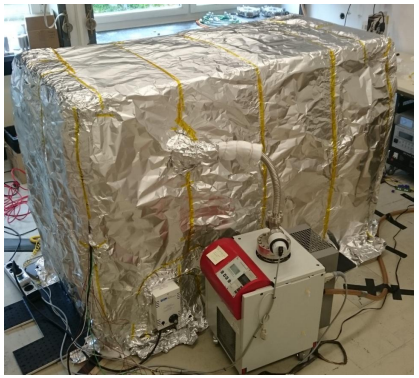
The setup

(x3)

serial / RS232



HV



...
x6
...

HV

(x2)

1-wire

serial / RS232

UDP

IQ network

Horia Varlan / Flickr

database

Tim Morgan / Flickr

webserver

RRZEicons / wikimedia

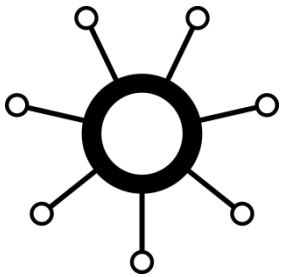
Dashboard


Vacuum monitor
Zoom Out
Last 3 hours
Refresh every 10s
Pressure in main chamber
1.2e-9 mbar
TPS1 pressure
2e-8 mbar
Pfeiffer cube pressure
3e-8 mbar
Yb oven IGP pressure
7e-9 mbar
Main IGP pressure
1.4e-9 mbar
Mean temperature
72 °C
Temperature dispersion
4.7 °C
Viewports
160 °C
140 °C
120 °C
100 °C
80 °C
60 °C
16:00
16:30
17:00
17:30
18:00
18:30
Top viewport
Bottom viewport
Transverse cooling, south viewport
Transverse cooling, north viewport
2D-MOT, north viewport
Diag. MOT viewport, top-west
MOT south viewport
Bottom detection, north-west viewpo
Top detection, south-west viewport
Top detection, north-east viewport
2D-MOT, south viewport
2D pusher viewport
Diag. MOT viewport, bottom-east
Horiz. MOT viewport, north
Chambers & others
95 °C
90 °C
85 °C
80 °C
75 °C
70 °C
65 °C
60 °C
16:00
16:30
17:00
17:30
18:00
18:30
Pumping cross
Offset coil mount
Oven IGP flange
CF16 gate valve, ZS side
Diff. pumping stage
Main IGP flange
Capacitorr flange
Main chamber, south-west
Main chamber, north-east
IMG-300 - Pressure in main chamber
5e-9 mbar
4e-9 mbar
3e-9 mbar
Pfeiffer cube pressure
8e-8 mbar
7e-8 mbar
6e-8 mbar
5e-8 mbar
TPS1 pressure
4.5e-8 mbar
4e-8 mbar
3.5e-8 mbar

The components

Requirements:
- open-source
- lightweight and easy to deploy
- rather mainstream
- packaged in the Arch Linux distribution
- when relevant: Python scripting / plug-in interface

Data collection: `collectd` (https://collectd.org)



Data storage: `influxdb` (https://influxdata.com/time-series-platform/influxdb/)

Data visualization: `grafana` (http://grafana.org/)

The components

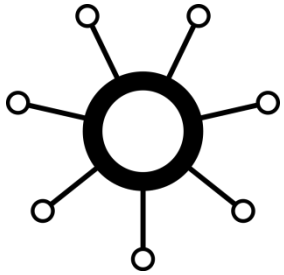
	collectd	influxdb	grafana
Latest release	5.5.2 (2016-07-26)	0.13.0 (2016-06-17)	3.1.0 (2016-07-12)
License	GPL	MIT	Apache
Availability (package)	Archlinux: yes Debian: yes	Archlinux: yes (AUR) Debian: yes (testing)	Archlinux: yes Debian: yes (unstable)
Python friendliness	**collectd-python** (plugin)	**influxdb** on PyPI	Not relevant?
Alternatives (among others)	statsd, telegraf (1)	elasticsearch, graphite (whisper) (2)	graphite-web, kibana, chronograph (3)

(1) See also: https://wiki.archlinux.org/index.php/List_of_applications#System_monitoring
http://graphite.readthedocs.io/en/latest/tools.html
(2) See also: https://en.wikipedia.org/wiki/Time_series_database
(3) See also: http://dashboarddude.com/blog/2013/01/23/dashboards-for-graphite/

Note: not all combinations of the alternatives work easily together

Some more details

… what was found to be good with this combination



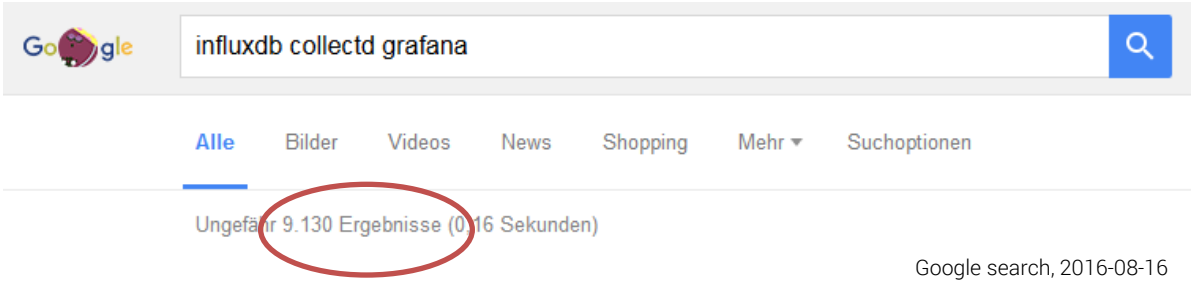
- Interfacing sensors with Python (`pyserial`) is usually simple and flexible (but can be tedious)
- `collectd-python` allows to give the collection and transfer responsibilities to a reliable 3rd party
Interface with `influxdb` is supported

```
def read_logic(self, win):
    return True if self.ask_raw(win, 10) == "1" else False

def read_alpha(self, win):
    return self.ask_raw(win, 19)

def build_message(self, win, cmd, data=None):
    if cmd not in [self.READ, self.WRITE]:
        raise ValueError("Command must be READ or WRITE")

    if data is None:
        data = b""

    if cmd == self.READ and data:
        raise ValueError("READ commands do not take data")

    if cmd == self.WRITE and not data:
        raise ValueError("WRITE commands need non-empty data")

    msg = bytearray(self.STX + self.addr)
    msg.extend(self._format_window(win))
    msg.extend(cmd)
    msg.extend(data)
    msg.extend(self.ETX)
    msg.extend(self.compute_crc(msg))

    return msg

@classmethod
def _format_window(cls, win):
    try:
        num = padleft(str(win), "0", 3)
    except ValueError:
        raise ValueError("Invalid window: '{}'".format(win))

    return num.encode("ascii")
```

- Very powerful and intuitive, provides quickly decent graphs
- Supports a variety of datasources

Grafana

Popular = external support possibilities

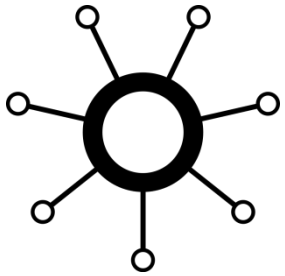


Google search, 2016-08-16

Some more details

Leibniz Universität Hannover

… selection of things that should be done better



- Debugging in **`collectd-python`** is annoying (because of the execution context)
- 1-wire thermal sensor family support in Linux kernel is limited (in particular if one wants to read many sensors)

```
def read_logic(self, win):
    return True if self.ask_raw(win, 10) == "1" else False

def read_alpha(self, win):
    return self.ask_raw(win, 19)

def build_message(self, win, cmd, data=None):
    if cmd not in [self.READ, self.WRITE]:
        raise ValueError("Command must be READ or WRITE")

    if data is None:
        data = b""

    if cmd == self.READ and data:
        raise ValueError("READ commands do not take data")

    if cmd == self.WRITE and not data:
        raise ValueError("WRITE commands need non-empty data")

    msg = bytearray(self.STX + self.addr)
    msg.extend(self._format_window(win))
    msg.extend(cmd)
    msg.extend(data)
    msg.extend(self.ETX)
    msg.extend(self.compute_crc(msg))

    return msg

@classmethod
def _format_window(cls, win):
    try:
        num = padleft(str(win), "0", 3)
    except ValueError:
        raise ValueError("Invalid window: '{}'".format(win))

    return num.encode("ascii")
```

influxdb

- Investigate **transport security** and proper authentication
- Try **other bakends** (in particular **elasticsearch**) to see if they can be better suited for certain applications

- Some quirks (in particular with scientific notation) require probably more in-depth configuration
- Maintaining a **secure webservice** is demanding (but integration in IQ central authentication seems feasible)

Grafana

Future steps

Monitoring the bakeout was only a pretext

Interesting things to monitor in our labs:
- temperatures (room, crystals,…), humidity
- frequencies / fixed synthesizer settings
- wavelengths
- heartbeats
- magnetic fields
- probably many more…

Any suggestions for the above mentioned quantities?

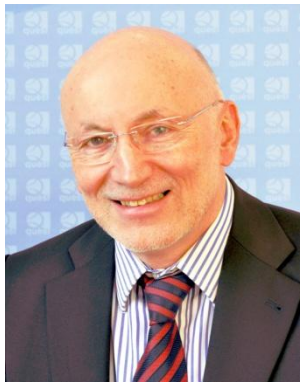
Exists for many devices (usually RS232)

This requires some work, how about merging the effort?

Steps to take:
0. find suitable sensor (if relevant)
1. convert sensor data to digital
2. write `collectd` plugin to read the sensor
3. done

Side note: this has nothing to do with realtime (fast) monitoring during an experimental cycle and is intrinsically limited (by the DB) to 1s resolution. Also, this other kind of monitoring should be decoupled from the environmental one.

Acknowledgments



W. Ertmer

E. M. Rasel

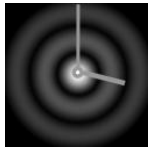
D. Schlippert

C. Schubert

D. Nath

C. Meiners

HITec



RTG 1729

DFG Deutsche Forschungsgemeinschaft

IQ Institute of Quantum Optics University of Hannover

QUEST

Code* is available on request:

wodey@iqo.uni-hannover.de

(maybe future publication on e.g. github but need to clarify the legal side: any input on this?)

Questions? Comments? Similar setups?

Thank you for your attention

*Dirty and minimal