Raspberry PICO and IoT for RPC chamber slow control

C.Combaret ,L.Mirabito

Institut de Physique des 2 Infinis de Lyon

June 18, 2024

Slow control of RPC chambers in Lyon

Controls and measures

Our RPC chambers operation requires:

- High Voltage (< 10 kV) provided by HV modules plugged in CAEN (SY1527) or Wiener crates
- Low Voltage (6 V) provided by Lambda Genesys or Zup power supplies
- Brooks Mass flowmeters for the 3 used gas (TFE,CO2/IsoButane,SF6)
- Bosch sensors (BMP283,BME280) for Pressure and Temperature monitoring
- Honeywell humidity sensor (HIH8000)
- Weighing machine (ADAM CPW+) for gas bottle monitoring

Previous framework

Readout

- HV: CAEN → Socket TCP/IP, Wiener → SNMP , from a linux PC
- LV: Zup and Genesys → RS232 on raspberry Pi
- Brooks flowmeters → Standalone Windows or micro-controller soft provided by resellers
- P,T: BMP283,BME280 → SPI/I2C on raspberry PI
- H,T: HIH8000 → I2C on raspberry PI
- Weighing machine ADAMS not read

Software

- Dedicated Linux daemon for each hardware, interfaced with REST Services for command (START,STOP,STATUS,COMMAND)
- One central Linux service for user interface (via python scripts), MongoDB storage and feeding of GRAFANA data point
- GRAFANA monitoring display

Motivations to change

Constraints

Hardware

- No watchdog if one readout is stuck.
- No service configuration in case of power supply cut or network short failure for example
- Some hardware not accessible remotely

Software

- Central configuration frozen, not really flexible in case of multiple tests
- All controls are based on computer with OS (update, security)
- Some software non-free

Solution

hardware → Use a micro-controller Raspberry PICO wherever it's possible

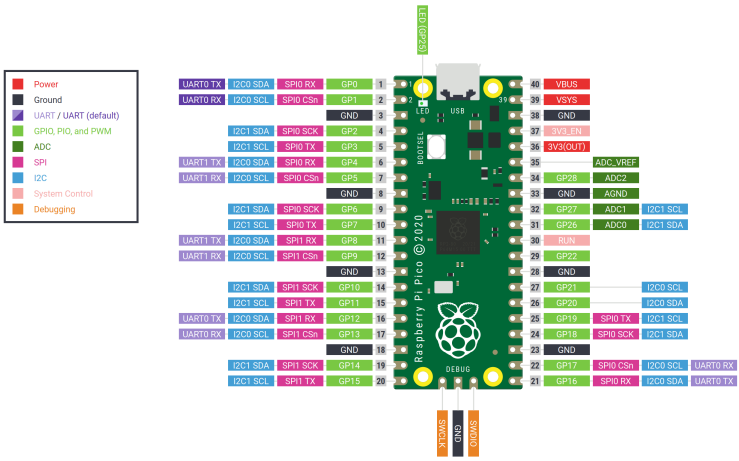
software Internet Of Things → MQTT

The Raspberry PICO

The Raspberry PICO is a micro-controller board developed by the Raspberry foundation:

- RP2040 double core at 133 MHz, Ram 264 Ko , 2 Mo of flash memory
- Low cost
- Programmable in C or python
- Large community of developers (hardware and software)

Connectivity



- 2 SPI, 2 I2C, 2 UART, GPIO, ADC, Timer
- μ RP2040 with embedded eeprom (0.2 Euros)
- USB, WiFi or Ethernet RJ45 TCP/UDP (wiznet W5500)

Two models

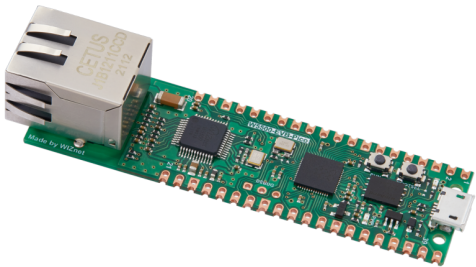


Figure: Ethernet 17 Euros

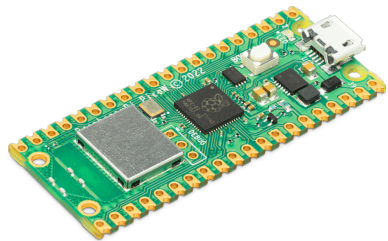


Figure: WiFi 8 Euros

Adaptation Board

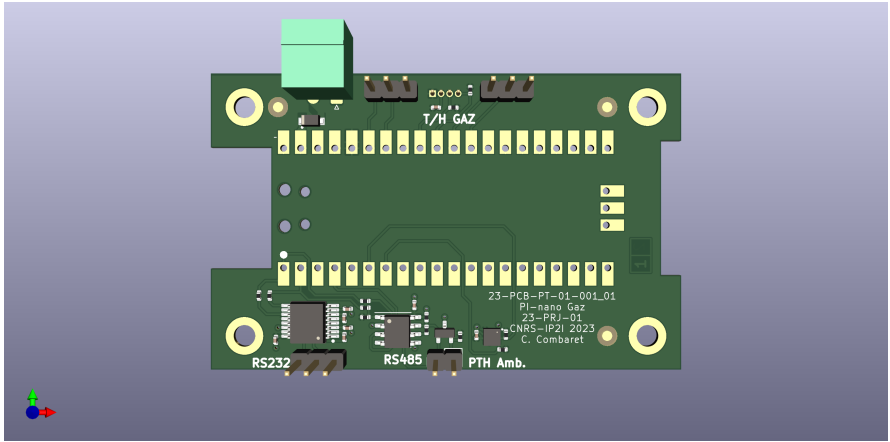


Figure: Versatile interface Board to our sensors. An additional LCD screen can be connected

Readout organization

hardware	Bus	Measures	device
Genesys / Zup	RS232	V,I	pico board
Brooks SLA5800	RS485	flow	pico board
BME280	SPI	P,T,H	pico board
HIH8000	I2C	H,T (Gas)	pico board
ADAMS CW+	RS232	Mass	pico board
Wiener HV	SNMP	V,I	Linux service on PC
SY1527 HV	TCP socket	V,I	Linux service on PC

P/T/H and LV

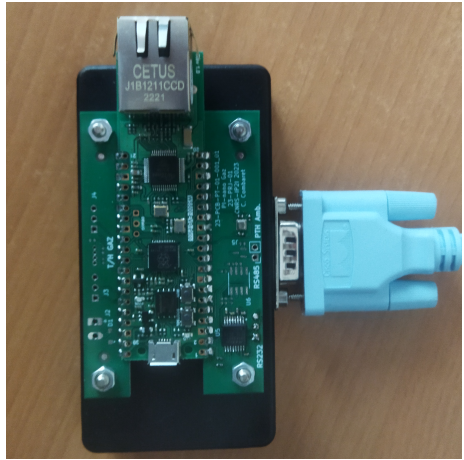
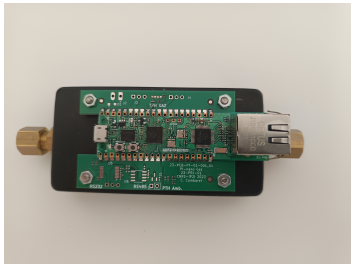


Figure: Adaptation board with an RS232 interface connected towards LV or weighing machine. An additional P/T/H BME280 sensor is connected to measure ambient pressure variation.

Gas relative humidity sensor

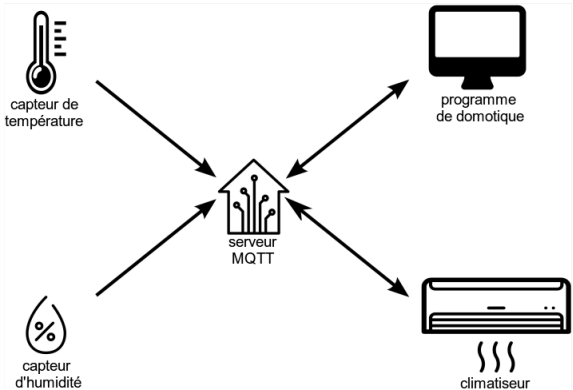
We used the HIH8000 sensor from Honeywell:



- The **HIH8000** is connected via a flat cable on top of the board.
- It is glued in a plastique tube inside the box.
- The box is then filled with epoxy to ensure the gas tightness.

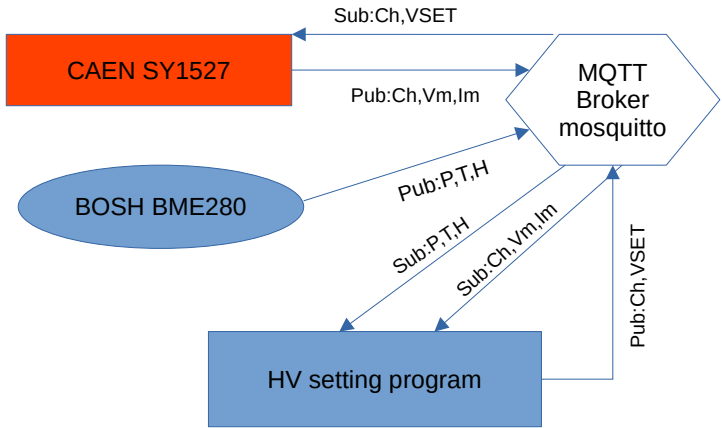
MQTT

Protocol of the Internet Of Things for home automation:



- The **broker** centralizes messages and redistributes them
- Sensors publish their data
- Air conditioner publish its status and subscribes to commands
- The home automation program subscribes to sensors data and publishes commands to the air conditioner

Application to HV control



Software framework

MQTT

- Library available in C,C++,python,JavaScript...
- Local Broker on linux → `mosquitto` deployed in `docker`

Code organization

- C++ for HV crates (TCPIP), python on micro-controllers
- Each client publish periodically its measurements and if appropriate, subscribes to commands (LVON,HVSET,FLOWSET....)
- One central linux daemon, subscribing to the status messages, is kept to save data in the MongDB database

Programming environment

μ python

- Python implementation adapted to micro-controllers
- Libraries for each type of micro-controllers and buses (Ethernet,MQTT,SPI,I2C,UART,ADC...)
- Easy deployment and maintenance (Code copy in the Flash memory)

Commands

- Line command scripts : `mosquito_pub` `mosquitto_sub`
- Python script (Library Eclipse Paho)
- Web pages with MQTT JavaScript library

Monitoring

- Debug: MQTT explorer
- Visualization: plugin MQTT in GRAFANA

MQTT explorer

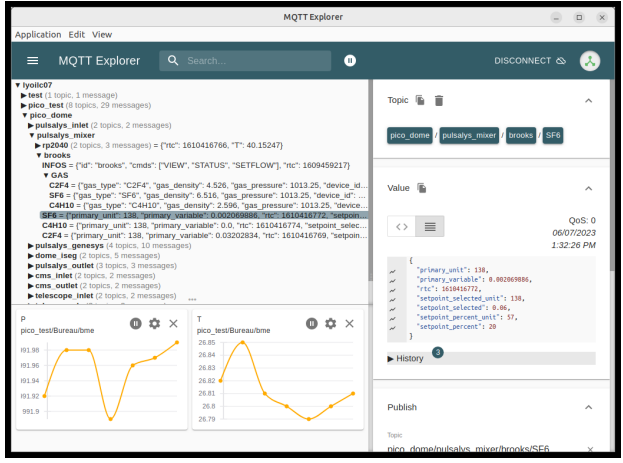
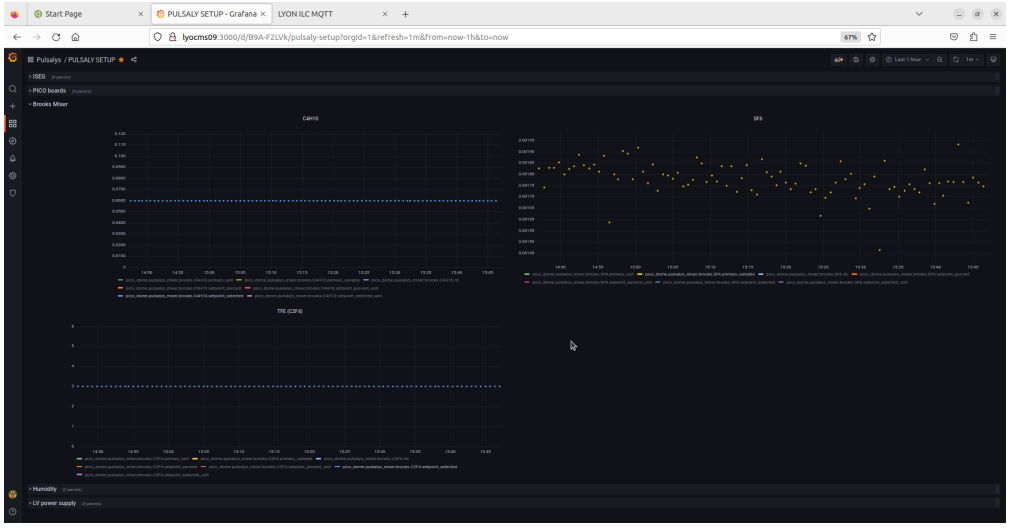
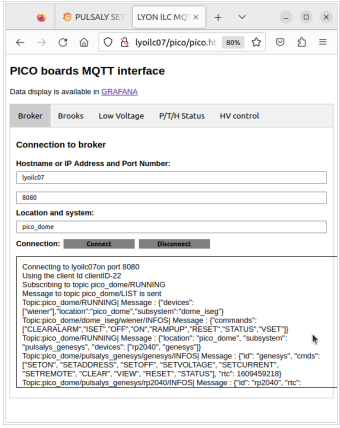


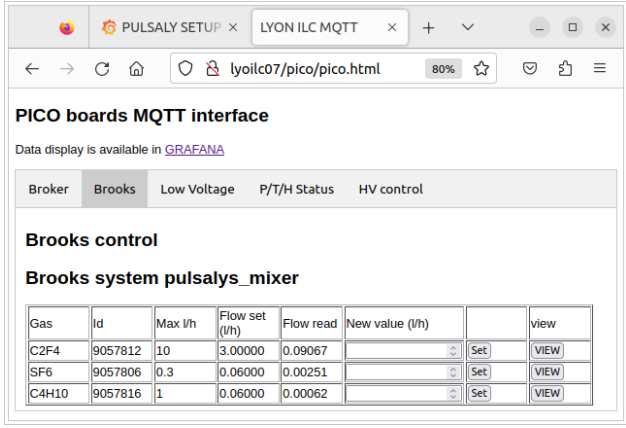
Figure: Open source debug tool for MQTT

GRAFANA Monitoring



Web pages with JavaScript interface





Conclusion

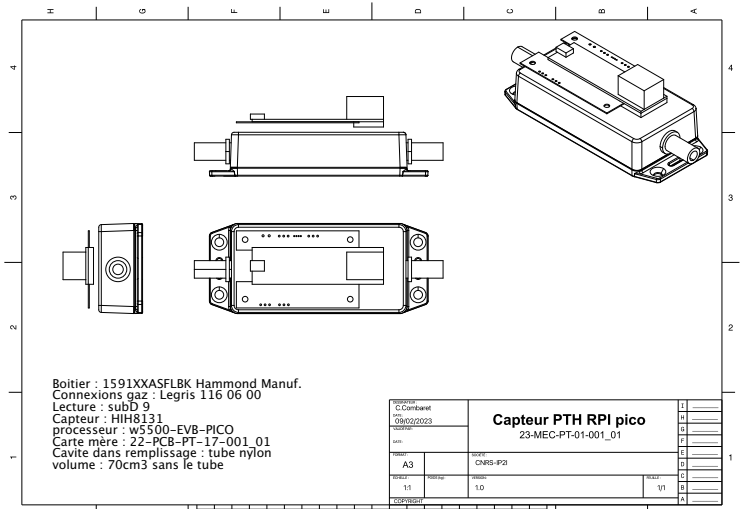
Hardware

Using the Raspberry PICO with our adaptation board gave us a real Swiss knife to control and monitor most of our systems: Gas, LV, pressure, temperatures, humidity and mass.
The cost of each system does not exceed a few tenth of Euros.

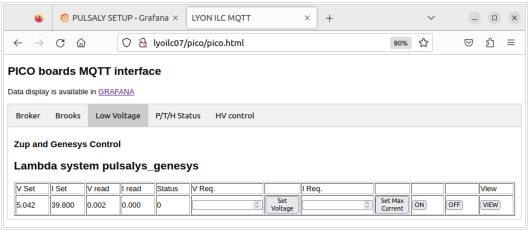
Software

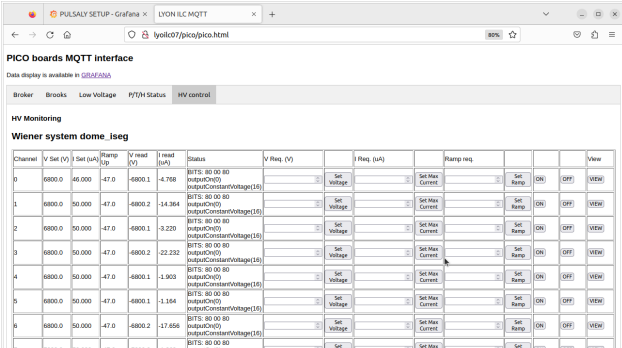
MQTT is a light,versatile, and flexible solution to manage slow control process. A large set of development and analysis tools and library are available, on a large variety of platforms and languages.
Our code is available on github together with the adaptation board Kicad project.

Backup Gas relative humidity measurement (mechanic)



Backup: Web pages with JavaScript interface





Backup: Thonny IDE

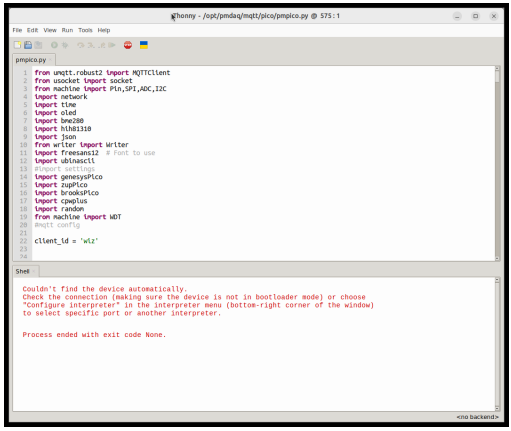


Figure: Python editor interfaced to raspberry pico

- Python IDE
- Libraries for PICO and MQTT
- Debug of the micro-controller via USB