Contrôle des chambres RPC avec le microcontrôleur Raspberry pico et l'IoT

C.Combaret ,L.Mirabito

Institut de Physique des 2 Infinis de Lyon

June 8, 2023

Contôle des chambres RPC à Lyon

Contrôles et mesures

L'opération des chambres RPC nécessite:

- La haute tension ($<$ 10 kV) fournie par des modules HV dans des controlleurs CAEN (SY1527) ou Wiener
- La basse tension (6 V) fournie par des alimentations Lambda Genesys ou Zup
- Des debitmètres massiques Brooks pour les 3 gaz utilisés (TFE,CO2/IsoButane,SF6)
- Des capteurs environnementaux de pression et température Bosch (BMP283,BME280)
- Des capteurs d'humidité Honeywell (HIH8000)
- Des balances pour les bouteilles de gaz

Système pré-existant

Lecture

- HV: CAEN → Socket TCP/IP, Wiener → SNMP , depuis un PC
- Zup et Genesys → RS232 sur raspberry Pi
- Debitmètre Brooks → Soft windows non connecté fourni par ServInstrumentation
- BMP283,BME280 → SPI/I2C depuis raspberry PI
- HIH8000 → I2C depuis raspberry PI
- Balance ADAMS non lue

Système pré-existant

Software

- Service linux dédié pour chaque hardware, interfacé par des services REST (START,STOP,STATUS,COMMAND)
- Un service d'interface centrale
  - Interface utilisateur (via des scripts python)
  - Stockage MongoDB, propagagtion vers GRAFANA
- Un monitoring dans GRAFANA

Système pré-existant

Contraintes

- Configuration centralisée peu flexible si plusieurs tests sont menés en parallèle
- Pas de watchdog si un des service hardware est bloqué dans un accès.
- Pas de redémarrage lors d'un arrêt sauvage du PC (coupure secteur)
- Tout les contrôles dépendent d'ordinateurs avec OS (mise à jour, sécurité)

Solution

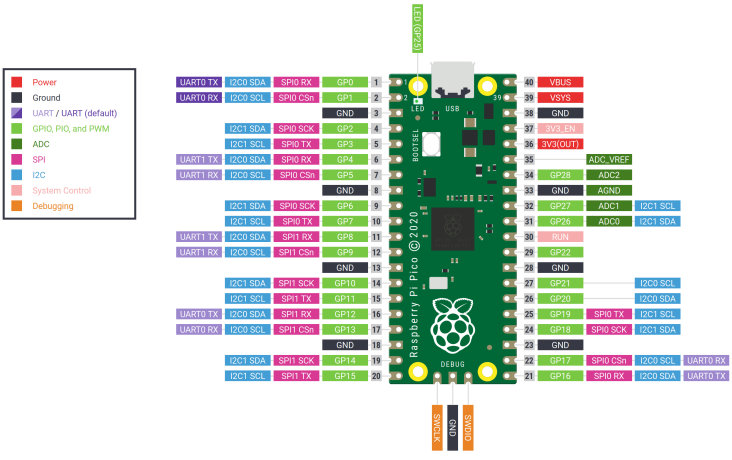
- Utilisation d'un microcontrôleur $\rightarrow$ Raspberry pico
- Internet des Objects $\rightarrow$ MQTT

Le Raspberry PICO

Le Raspberry PICO est une carte microcontrôleur développé par la fondation Raspberry:

- RP2040 double coeur à 133 MHz, Ram 264 Ko , 2 Mo de memoire flash
- Bas coût
- Programmable en C et en python
- Large communauté de développeurs (hardware et software)

Connectique



- 2 SPI, 2 I2C, 2 UART, GPIO, ADC, Timer
- μ RP2040 avec eeprom embarquée (0.2 Euros)
- USB, interface wifi ou ou TCP/UDP (wiznet W5500)

Deux modèles

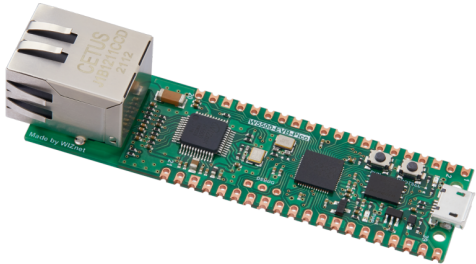


Figure: Ethernet 17 Euros

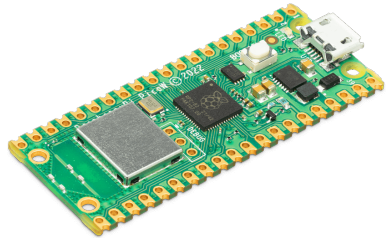


Figure: Wifi 8 Euros

Board d'adaptation

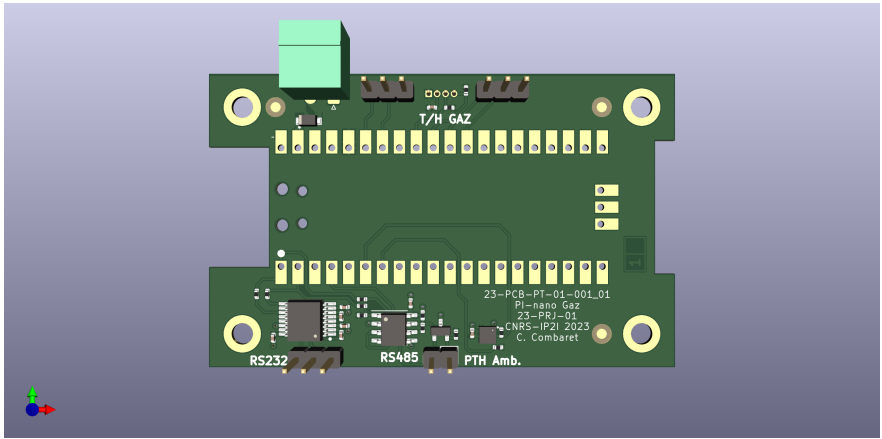


Figure: Board versatile d'interface vers nos capteurs. Un ecran LCD peut également être connecté

Organisation des contrôles

hardware	Bus	Mesures	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 (Gaz)	pico board
ADAMS CW+	RS232	Masse	pico board
Wiener HV	SNMP	V,I	Service linux sur PC
SY1527 HV	TCP socket	V,I	Service linux sur PC

Exemple

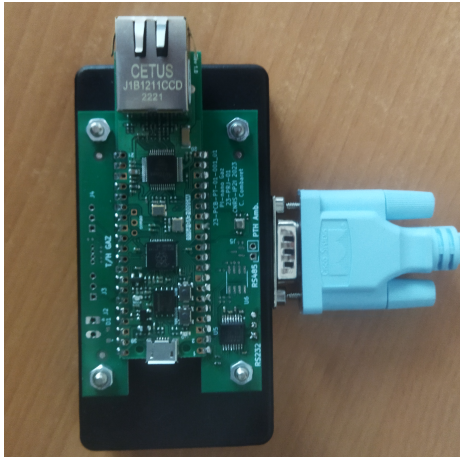
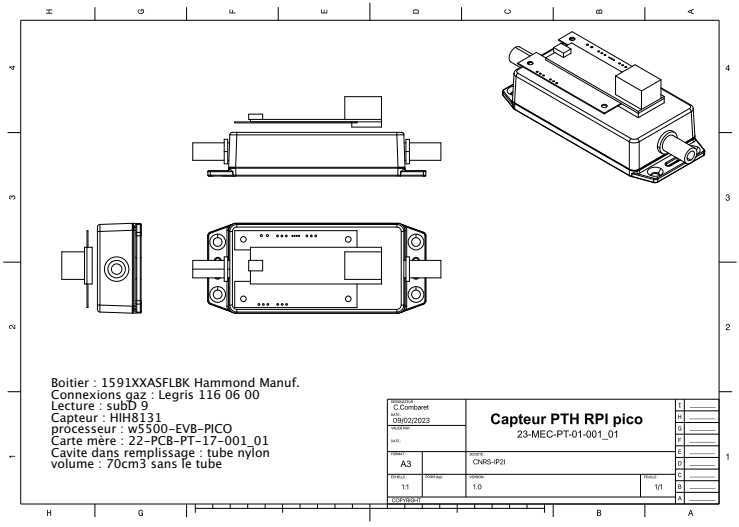


Figure: Boitier d'adaptation vers une interface RS232 (LV ou balance). Un capteur P/T/H ambiant BME280 est rajouté pour monitorer les variations de pression.

Mesure du taux d'humidité du gaz (HIH8000)



Software

MQTT

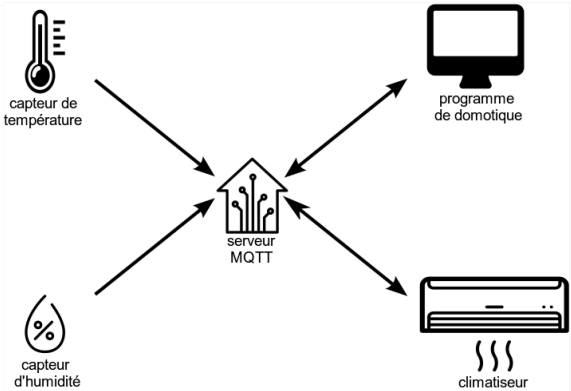
- Protocole développé pour l'IoT
- Aiguillage centralisé (BROKER) pour des messages PUBLISH/SUBSCRIBE
- Bibliothèques disponibles C,C++, python
- Broker local sous linux $\rightarrow$ mosquitto

Code dédié

- C++ pour les alimentations HV (TCPIP), python sur tous les microcontroleurs
- Chaque client publie périodiquement ses mesures et le cas échéant souscrit à des commandes (LVON,HVSET,FLOWSET....)
- On conserve un service linux unique pour le stockage dans une base de données

MQTT

Protocole pour l'internet des objects:



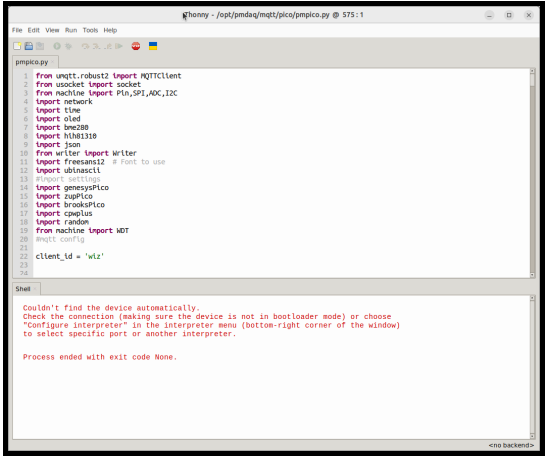
- Le **broker** centralise les messages et les redistribue
- Les capteurs publient leurs données
- Le climatiseur publie son status et souscrit à des commandes
- Le programme de domotique monitore les capteurs et publie des commandes au climatiseur

Environnment de développement

μ python

- Implémentation python adaptée aux microcontrôleurs
- Bibliothèques pour chaque type de processeurs (Ethernet,MQTT,SPI,I2C,UART,ADC...)
- Déploiement et maintenance aisée (copie du code en Flash)

Thonny IDE



- IDE Python
- Bibliothèques PICO et MQTT
- Debuggeur du microcontrôleur via USB

Figure: Editeur Python interfacé au raspberry

Interface utilisateur

Commandes

- Commande en ligne : *mosquito_pub* *mosquitto_sub*
- Scripts python (bibliothèque Eclipse Paho)
- Page web avec bibliothèque MQTT en javascript

Monitoring

- Debug: MQTT explorer
- Visualisation: plugin MQTT dans GRAFANA

MQTT explorer

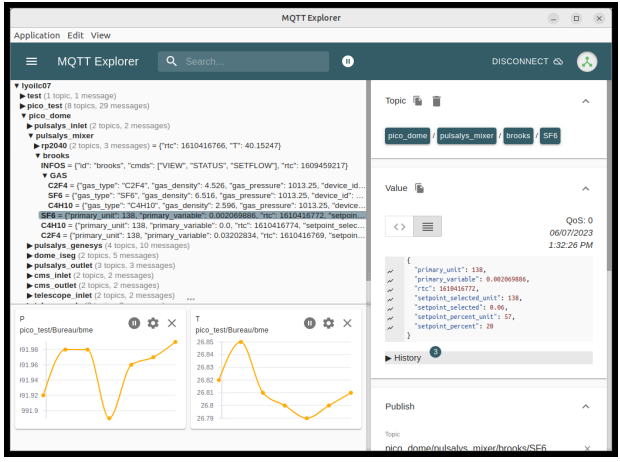
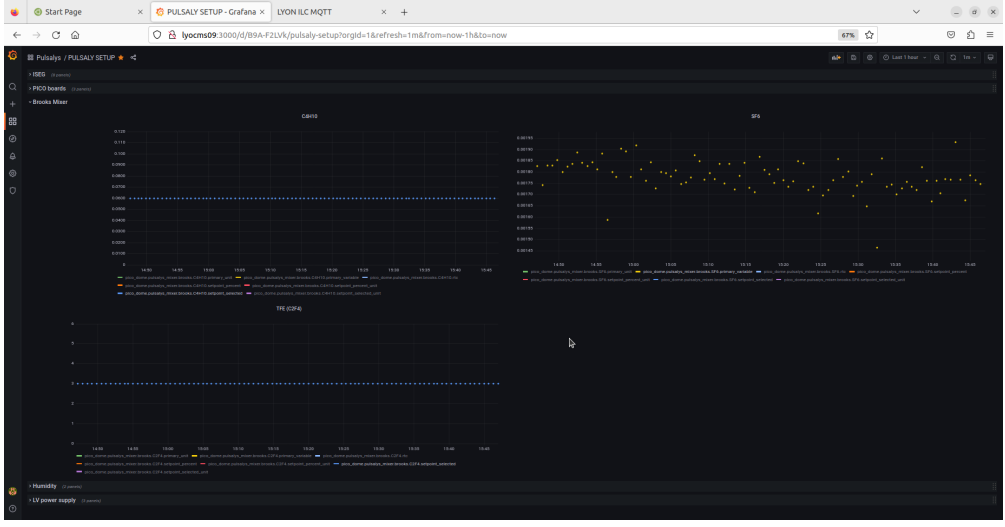
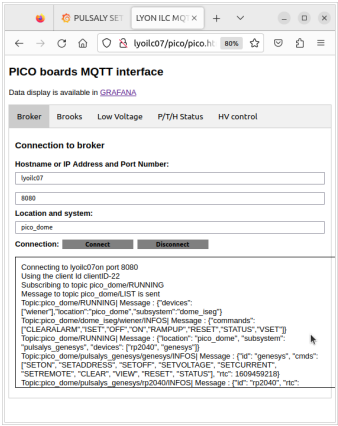


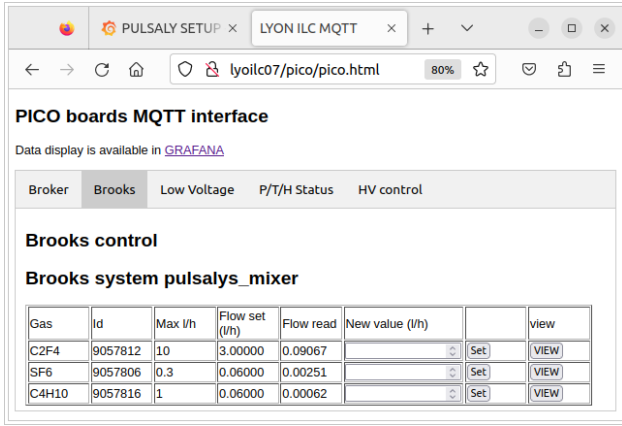
Figure: Outil de debug MQTT

Monitoring GRAFANA

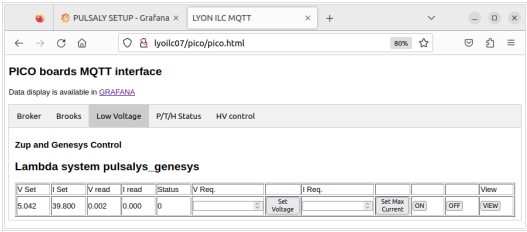


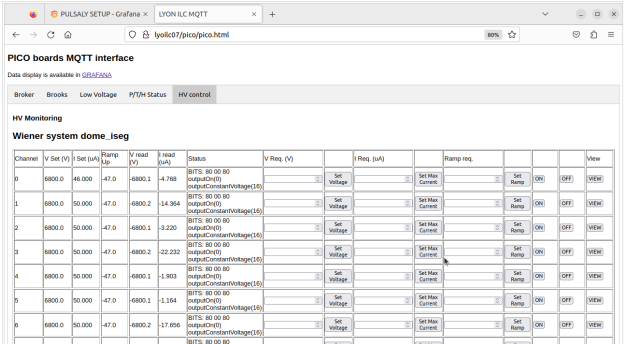
Interface JavaScript





Interface JavaScript





Conclusion

Hardware

L'utilisation du raspberry pico et le développement du PCB d'interface nous a permis d'avoir un véritable couteau suisse pour contrôler l'ensemble de nos systèmes: gaz, LV et environnement. Le coût de chaque boitier n'excède pas quelques dizaine d'Euros.

Software

MQTT est une solution légère pour gérer ces processus de slow control. Enormément d'outils de développement et d'analyse sont disponibles, sur une grande variété de platformes. L'ensemble de notre code est disponible sous github.