POLITECNICO
MILANO 1863

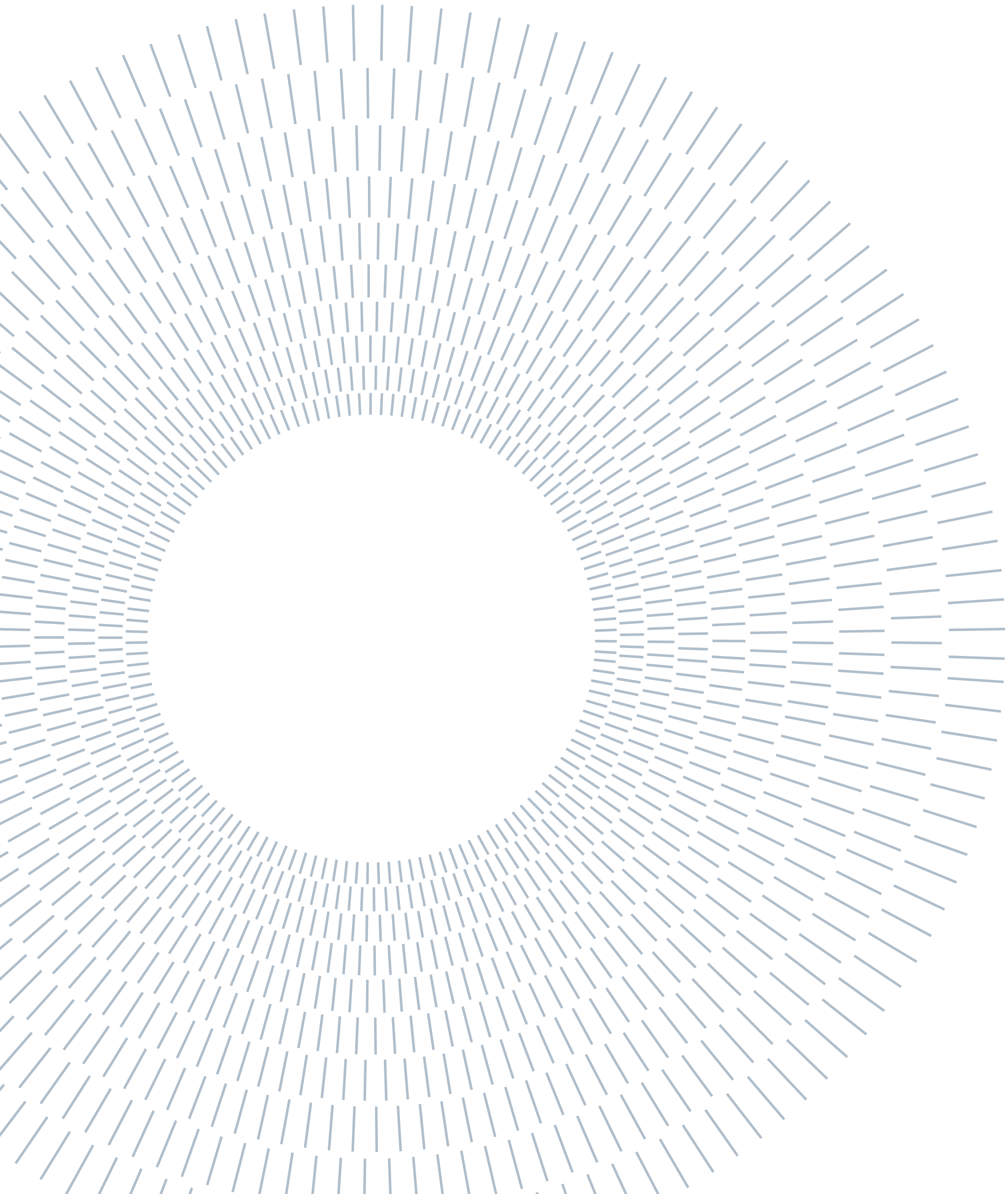
SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

IoT Challenge #2, Exercise Application Layer

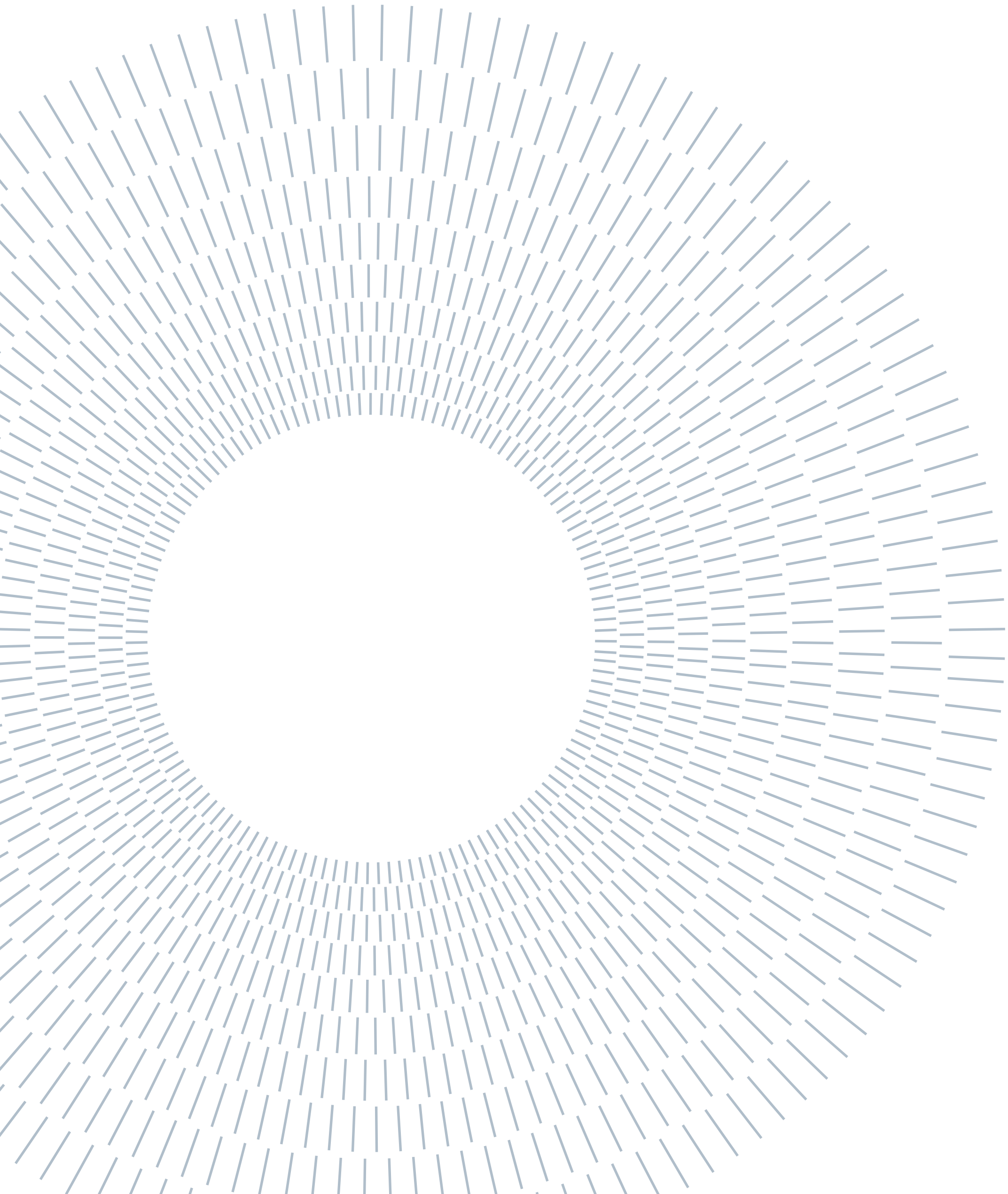
Internet of Things

Authors: **Kevin Ziroldi - 10764177**
Matteo Volpari - 10773593

Professors: Alessandro Redondi, Fabio Palmese
Academic Year: 2024-2025
Version: 1.0
Release date: 6-4-2025



Contents



1 | Exercise temperature sensor and valve

1.1. Data

The data of the exercise is reported here.

- $T_{sensor\_reading} = 5$ minutes
- $T_{average\_computation} = 30$ minutes
- $L_{resource} = L_{topic} = 10$ Bytes
- $L_{payload} = 8$ Bytes
- $E_{TX} = 50$ nJ/bit
- $E_{RX} = 58$ nJ/bit
- Ideal Wi-Fi network
- $E_C = 2.4$ mJ

The message sizes of the two protocols are reported in the following tables.

Message	Size [Byte]
GET Request	60
GET Response	55
PUT Request	77
PUT Response	58
Empty ACK	14

Table 1.1: Message sizes CoAP

Message	Size [Byte]
Subscribe	58
Sub ACK	52
Publish	68
Pub Ack	51
Connect	54
Connect Ack	47
Ping Req	52
Ping Resp	48

Table 1.2: Message sizes MQTT

1.2. EQ1.a Energy consumed using CoAP

We start by computing the energy consumed by the two devices when they communicate using CoAP in the most efficient configuration energy-wise. The temperature sensor acts as a CoAP Server, while the valve as a CoAP Client.
In order to save energy, we use CoAP Observation and Non-confirmable requests. The valve (Client) sends a GET Request with Observe to the temperature sensor (Server). The sensor sends the requested value to the Client every 5 minutes, with a GET Response. Moreover, the valve computes the average every 30 minutes.
We start by computing the number of sensor readings, N_{sensor_read}, and the number of computations of the average, N_{avg}.

$$N_{sensor\_read} = (24 \text{ h} \cdot 60 \text{ min/h})/5 \text{ min} = 288$$

$$N_{avg} = (24 \text{ h} \cdot 60 \text{ min/h})/30 \text{ min} = 48$$

We can compute the energy consumed by the two devices as:

$$E_{CoAP\_valve} = E_{TX} \cdot L_{GET\_Req} + N_{sensor\_read} \cdot E_{RX} \cdot L_{GET\_Resp} + N_{avg} \cdot E_C = 122.574 \text{ mJ}$$

$$E_{CoAP\_sensor} = E_{RX} \cdot L_{GET\_Req} + N_{sensor\_read} \cdot E_{TX} \cdot L_{GET\_Resp} = 6.364 \text{ mJ}$$

$$E_{CoAP\_total} = E_{CoAP\_valve} + E_{CoAP\_sensor} = 128.938 \text{ mJ}$$

1.3. EQ1.b Energy consumed using MQTT

In this section, we compute the energy consumption if the devices communicate through MQTT. In this scenario, the Raspberry PI acts as MQTT Broker, the sensor is a Publisher and the valve is a Subscriber. In order to minimize energy consumption, all messages are exchanged with QoS 0 and without the Keep Alive mechanism, i.e. with Keep Alive 0. The sensor (Publisher) connects to the Broker and starts publishing temperature values every 5 minutes. The valve (Subscriber) connects to the Broker, subscribes to the topic and starts receiving temperature values.

We can compute the energy consumed by the two devices as:

$$E_{MQTT\_valve} = E_{TX} \cdot L_{CONNECT} + E_{TX} \cdot L_{SUBSCRIBE} + N_{sensor\_read} \cdot E_{RX} \cdot L_{PUBLISH} + N_{avg} \cdot E_C =$$

$$= 124.332 \text{ mJ}$$

$$E_{MQTT\_sensor} = E_{TX} \cdot L_{CONNECT} + N_{sensor\_read} \cdot E_{TX} \cdot L_{PUBLISH} = 7.855 \text{ mJ}$$

$$E_{MQTT\_total} = E_{MQTT\_valve} + E_{MQTT\_sensor} = 132.187 \text{ mJ}$$

1.4. EQ2 Improvements

In this section, we propose two ways to decrease the energy consumed by the two device while using the Raspberry PI as a broker.

1.4.1. Using MQTT-SN

The first improvement consists in using MQTT-SN instead of MQTT. Using MQTT-SN introduces a trade-off:

- Disadvantage: the Publisher needs to send a REGISTER message to the broker, represented by the Raspberry PI, before being able to send PUBLISH messages.
- Advantage: PUBLISH messages have a reduced size, since they have a 2 Bytes long topic, instead of the 10 Bytes long one of MQTT messages.

On the long run, using MQTT-SN is beneficial, since we send more messages and compensate the initial cost of the REGISTER message with the lowered cost of PUBLISH messages. We need to compute the energy consumption in our time horizon of 24 hours and see if it is long enough to consume less energy.

// TODO calcoli

We can see that the new energy consumption is smaller than the one computed using MQTT.

// TODO potrebbe essere che inviando ogni 10 minuti, non conviene più MQTT-SN rispetto a MQTT.