Project Report: Challenge 3 - IOT

Alberto Latino - **10600138**, *Lorenzo Mainetti* - **10622242**

Goal:
Simulate a TinyOs device able to perform the division by three of a specific number (person code). Based on the remainder of this division, specific leds were turned on or off. In particular, remainder equal to zero toggles Led0 and so on.

1. **ThingSpeak setup (**https://thingspeak.com/channels/1726996**)**
   We first set up the ThingSpeak channel, making it public and adding the requested dashboards: one chart and one lamp per field (field1, field2, field3).
   The ThingSpeak channel stores data sent from the Node-RED application through MQTT. Hence, we created a MQTT Device with our credentials.

2. **TinyOS App implementation**
   a. **ChallengC.nc**
      We used as components Leds, Boot and a Timer.
      At each step the division is performed, the remainder is computed and the result is forwarded to the next step. Each step is started by the Timer, which has a period of 60 seconds. The first iteration starts at the boot of the application. All the leds are initialized at [0,0,0] and at each step are toggled respectively. When the dividend becomes equal to zero, the timer is stopped. The code has memory of the state of the three leds and forwards to the Node-RED application the status of the leds. At each step the status depends on the previous one and on the value of the remainder.

   b. **ChallengeAppC.nc**
      Here we defined all the components implementing the application and we wired them.

3. **Node-RED application**
   We built a flow that meets the challenge goals and requirements. Below the nodes that we used and a description on how we used them. We created a flow that takes messages coming from the TinyOS simulator (Cooja) and forwards them to the ThingSpeak platform.

   TCP input Node
   - The output generated by the TinyOS application simulated in Cooja is sent through the 60001 port and tcp connection and intercepted by the tcp node at localhost.

   Function Node
   - Takes messages coming from the TCP connection established with the TinyOS simulator. Each message represents the status of the three leds and inserts it in the payload of the msg.

   MQTT out Node:
   - This node allowed us to connect and send data through MQTT to the ThingSpeak platform. At each step it forwards the status of the three leds to ThingSpeak, where the charts are generated and the leds are toggled.

We also added a **Debug Node** to print the msg.payload and to check that everything worked as expected. They were useful to understand what was going on in the flow and helped to fix initial errors.