

# Contiki-NG

## Lab Exercises (1/2)

Luca Mottola  
`luca.mottola@polimi.it`

# Practical Suggestions

- Use the Hello World example as a template
  - Duplicate the folder and start from there
  - If you change the name of the `.c` file, remember to change the **Makefile** too!
  - Weird compilation errors are likely due to compiler left-overs... run **make clean** if in doubt



# Exercise 1

- Write a Contiki-NG program that implements the classical **producer-consumer** pattern
  - A shared queue of a given (fixed) size of generic data items exists, say integers
  - At random intervals, one protothread pushes items into the queue as long as the queue is not full
  - At random intervals, one protothread pulls items from the queue as long as the queue is not empty
  - When a protothread cannot proceed because of the state of the queue
    - It must be suspended until that condition does not hold anymore
    - **You cannot use timers**
  - Use platform **native** to test



# Exercise 2

- Change the code of `timers` so that the `ctimer` and the `rtimer` expire at the same time
  - Absolute precision hard to achieve
  - ...but you can get pretty close if you design it carefully... ;)
  - Test in COOJA with MSPSim!

