Contiki Project - NES 2016/2017

Consider a **Smart Home** context where there are three wireless sensor and actuator nodes (Tmote **Sky**):

- a **Central Unit** (**CU**) node with Rime address **3.0**. It is placed in the living room, and it is the principal node of the WSAN. Imagine this node to be connected in output to a serial monitor. The user mainly interacts with this node, giving it the commands for the smart home and reading the feedbacks on the serial monitor;

- a node (**Node1**) with Rime address **1.0**. It is placed in the entrance hall of the house, close to the door;

- a node (**Node2**) with Rime address **2.0**. It is placed in the garden of the house, close to the gate.

There exist **5 possible commands** that the user may give to the CU. Each command corresponds to a **number N**. The user decides the command N by consecutively **pressing N times the button** of the CU. The command is actually determined when **4 seconds** have elapsed since the last button press. After that, the CU is **ready to receive a new command** from the user. Every time the CU is ready to receive a new command, it will have to show on the monitor the set of possible commands with the associated number N:

1. **Activate/Deactivate the alarm signal** - when the alarm signal is activated, **all the LEDs** of Node1 and Node2 start blinking with a period of **2 seconds**. When and only when the alarm is deactivated (the user gives again command 1), the LEDs of both Node1 and Node2 **return to their previous state** (the one before the alarm activation). Besides, **when the alarm is active, all the other commands are disabled**;

2. **Lock/Unlock the gate** - when the gate is locked, the **green LED** of Node2 is switched off, while the **red** one is switched on. Vice versa, when the gate is unlocked (the user gives again command 2), the green LED of Node2 is switched on, while the red one is switched off;

3. **Open (and automatically close) both the door and the gate in order to let a guest enter** - When the command is received by Node1 and Node2, their **blue LEDs have to blink**. The blinking must have a period of **2 seconds** and must last for **16 seconds**. The blue LED

of Node2 immediately starts blinking, whereas the blue LED of **Node1 starts blinking only after 14 seconds** (so, 2 seconds before the blue LED of Node2 stops blinking). The 16 seconds represent the time required for the gate/door to open and then close. The 14 seconds represent the time required for the guest to reach the entrance hall by crossing the garden;

4. **Obtain the average of the last 5 temperature values measured by Node1**. Node1 **continuously measures** temperature with a period of **10 seconds**;
5. **Obtain the external light value measured by Node2**.

Finally, the user also has the possibility to **switch on and switch off the lights in the garden**. This is done by directly pressing the **button of Node1**. The garden lights are on when the **green LED** of Node1 is on, and the **red** one is off. Vice versa, the garden lights are off when the red LED is on, and the green one is off.

N	COMMAND
1	Activate/Deactivate the alarm signal
2	Lock/Unlock the gate
3	Open (and close) the door and the gate
4	Average of the last 5 internal temperature values
5	External light value

NOTA BENE - **extend** the project as you like it, by adding **some more motes and/or commands**.

Evaluation

- All the firmwares **must compile and respect the specifications** when simulated in Cooja (these are the minimum requirements);
- Try to **apply all the knowledge and concepts** that you have learnt during the lessons. This would increase the overall quality of the work;
- The project has to be **discussed and simulated step by step** during the examination. You will have to show a **deep comprehension** of the topics in general and that the project is **your own work**.