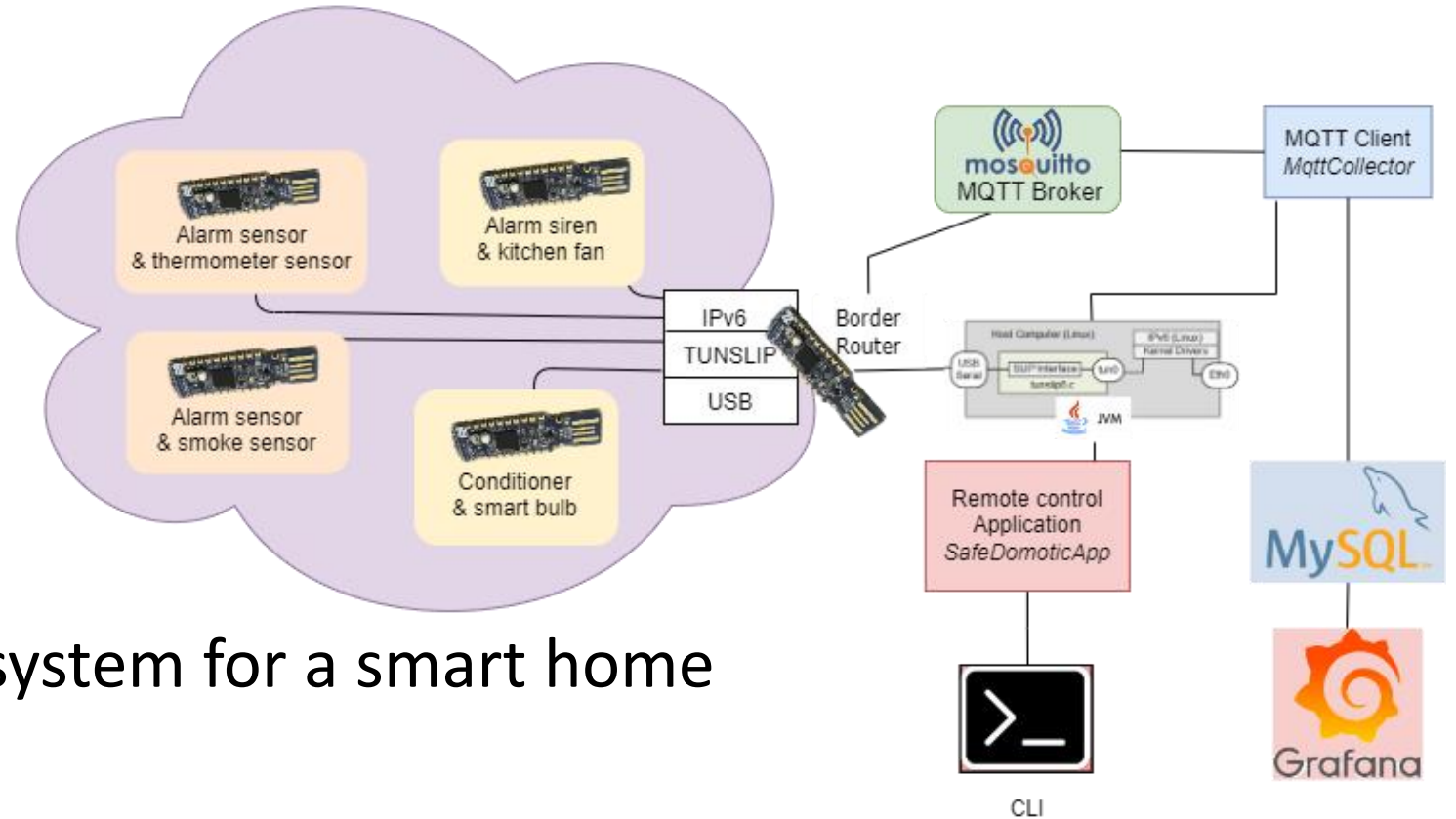


SafeDomoticHome

Project for Internet of Things course, 2022-23 ed.

Lorenzo Ceccanti

Introduction



- IoT telemetry and control system for a smart home
- Home equipped with:
 - Alarm sensor
 - Temperature sensor
 - VOC sensor to measure % of cooking fumes
 - Conditioner, alarm siren and smart bulb as actuators
 - Fan on the kitchen hob automatically triggered

nrf52840 deployment

1. Flash BR code onto the nrf52840 dongle (if necessary). In a new terminal (n.1):

```
osboxes@osboxes:~$ docker start -ai infallible_spence
user@osboxes:~/contiki-ng$ cd IoTSafeDomoticHome-main/
user@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ ./flashBR.sh
```

2. Deploy the BR dongle as first USB device

```
user@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ ./deployBR.sh
```

3. Flash MQTT and CoAP C source code on the other motes (if necessary). In a new terminal (n.2):

```
osboxes@osboxes:~$ docker start -ai eager_wescoff
user@osboxes:~/contiki-ng$ cd IotSafeDomoticHome-main/
user@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ ./flashTemperature.sh
user@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ ./flashVoc.sh
user@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ ./flashConditionerLight.sh
user@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ ./flashSirenFan.sh
```

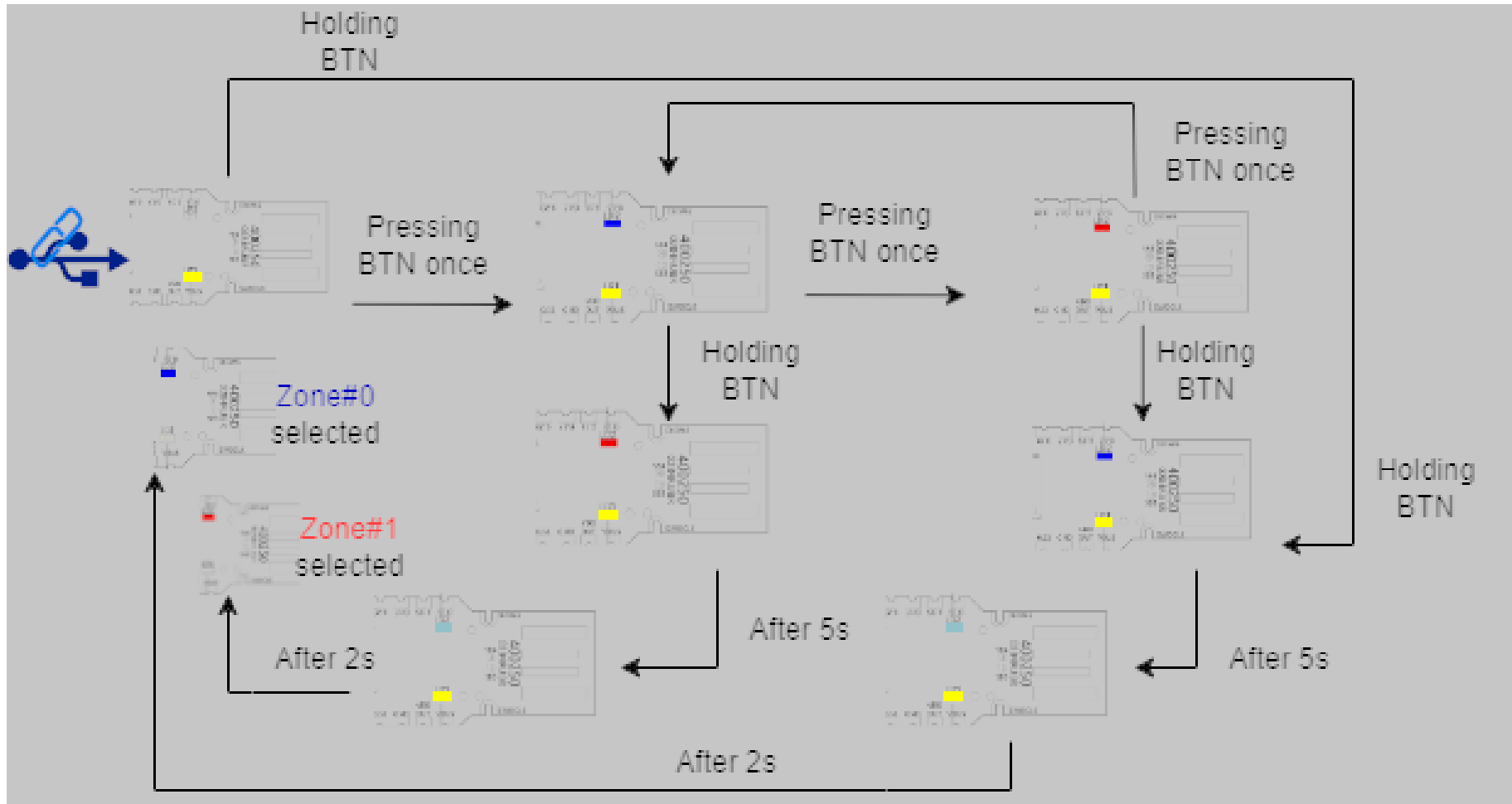
nrf52840 deployment (2)

4. **If the database does not exists, execute the SQL script.** After that, execute the Java Application *statical-coap-discovery* (if necessary). In a new nerminal (n.3):

```
osboxes@osboxes:~$ cd contiki-ng/IoTSafeDomoticHome-main/  
osboxes@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ ./startStaticCoap.sh
```

5. Connect the flashed nrf52840 dongles to USB hub

Alarm zone selection

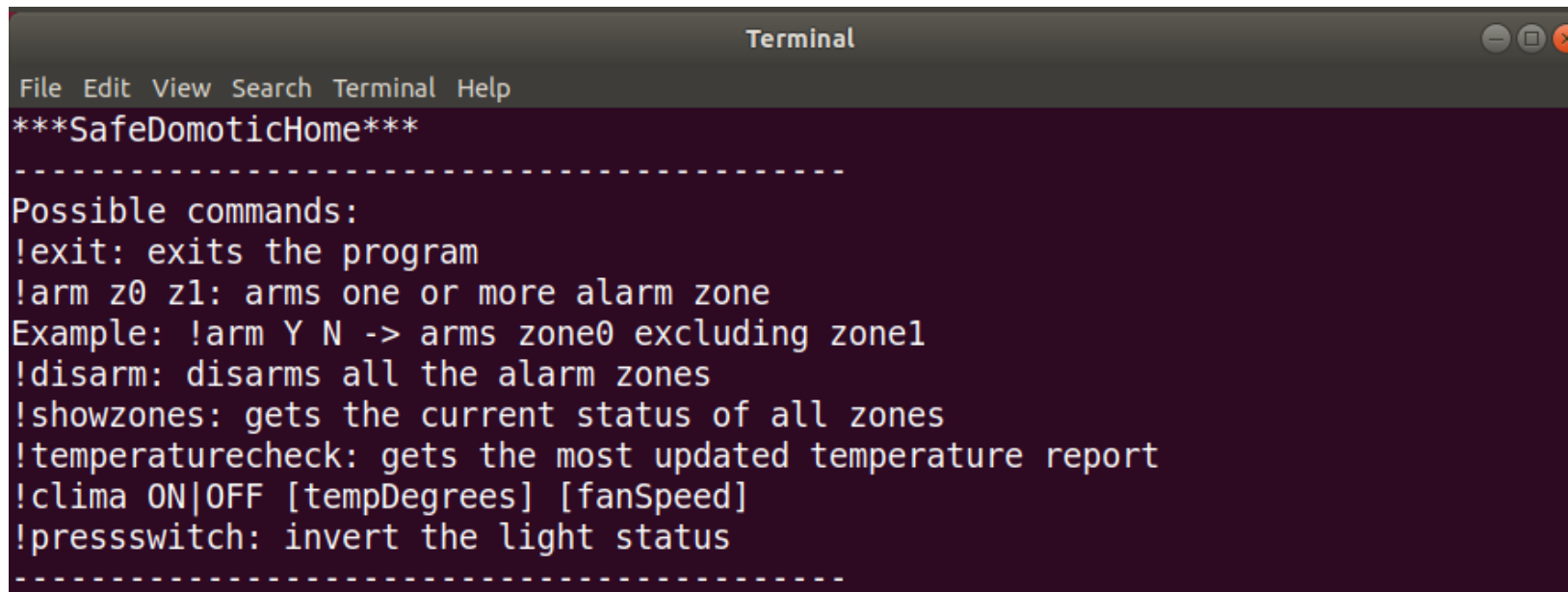


Launching *SafeDomoticApp*

To compile and run the Java Application, type:

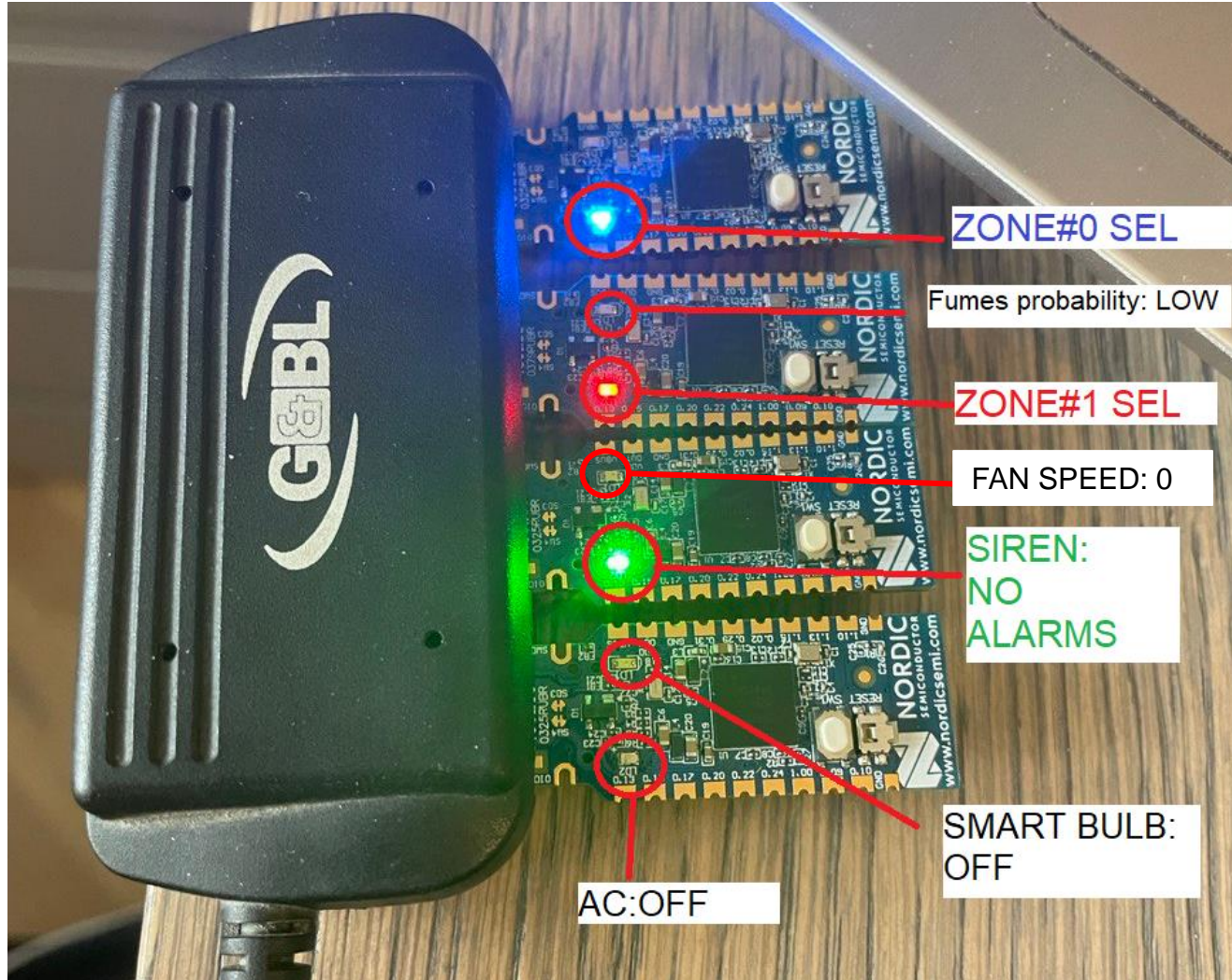
```
osboxes@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ ./startApp.sh
```

After that, a CLI will show up in a new window listing all the available commands:

A screenshot of a terminal window titled "Terminal". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal content shows the output of the startApp.sh script, which displays a CLI interface for SafeDomoticHome. The output includes a header "***SafeDomoticHome***", a separator line of dashes, and a list of possible commands with their descriptions. The commands listed are !exit, !arm, !disarm, !showzones, !temperaturecheck, !clima, and !pressswitch, each followed by a brief explanation of its function.

```
Terminal
File Edit View Search Terminal Help
***SafeDomoticHome***
-----
Possible commands:
!exit: exits the program
!arm z0 z1: arms one or more alarm zone
Example: !arm Y N -> arms zone0 excluding zone1
!disarm: disarms all the alarm zones
!showzones: gets the current status of all zones
!temperaturecheck: gets the most updated temperature report
!clima ON|OFF [tempDegrees] [fanSpeed]
!pressswitch: invert the light status
-----
```


LEDs legend



- In the picture we have the LEDs status right before the launch of the Java Application

Demo: Arming/disarming the alarm

- By pressing the button on the MQTT alarm sensor, you can increase the opening probability. MIN VALUE = 0% MAX VALUE = 80%
- To see information about that, in the «jolly_rubin» container:

```
user@osboxes:~/contiki-ng/IoTSafeDomoticHome-main$ cd mqtt-network/  
user@osboxes:~/contiki-ng/IoTSafeDomoticHome-main/mqtt-network$ make login TARGET=nrf52840 BOARD=dongle PORT=/dev/ttyACM1  
rlwrap ../../tools/serial-io/serialdump -b115200 /dev/ttyACM1  
connecting to /dev/ttyACM1 [OK]  
[INFO: door_sensor] Opening probability changed to: 80  
[INFO: door_sensor] Opening probability changed to: 0  
[INFO: door_sensor] Opening probability changed to: 20  
[INFO: door_sensor] Opening probability changed to: 40  
[INFO: door_sensor] Opening probability changed to: 60  
█
```


Demo: acting with the bulb

Demo: acting with the AC

Demo: trigger the kitchen fan

- By holding the button on the MQTT kitchen sensor, you can dramatically increase the cooking fumes percentage. The correspondent yellow LED will turn on.
- The actions made by the smart fan in the kitchen could be observed in the Grafana Dashboard together with some other summary information.
- <http://localhost:3000> -> SafeDomoticHome Grafana dashboard

Demo: trigger the kitchen fan (2)

First of all, select the appropriate refresh rate



- **<15%: good**
yellow led OFF
- **between 15 % and 22%:**
fan speed = 1
- **between 22% and 30%:**
fan speed = 2
- **>30%: critical level.**
fan speed = MAX (3)