LED Programming Tutorial

LED Simulation Example

- The simplest way to open the simulation is to select “LED Simulation” in the IoTrain-Sim interface
- Alternatively, you can open it manually as follows
  - Open Cooja
  - Click File > Open simulation > Browse…
  - Go to the folder “iotrain-sim/database/fundamental_training/single_node/actuation_control/led/simulation”
  - Select the file “led.csc”
  - Click Open
- Once the simulation control window appears, click the “Start” button to begin the simulation
  - The logical running time for this simulation is set to 10 minutes

Source Code Commentary

- Use a Tmote Sky node that prints a message and turns on the red LED
  - Source code: iotrain-sim/database/fundamental_training/single_node/actuation_control/led/simulation/led.c

```
#include “contiki.h”
#include “dev/leds.h” ❶
#include <stdio.h>
/*---------------------------------------------------------------------------*/
PROCESS(led_process, "LED process");
AUTOSTART_PROCESSES(&led_process);
/*---------------------------------------------------------------------------*/
PROCESS_THREAD(led_process, ev, data)
{
  PROCESS_BEGIN();
  printf(“Turn ON the red LED\n”);
  leds_on(LEDS_RED); ❷
  PROCESS_END();
}
/*---------------------------------------------------------------------------*/
```

Source Code Commentary (cont.)

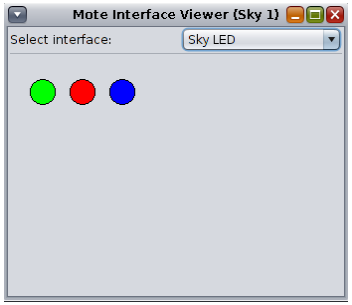
❶ We add “dev/leds.h”, which is used to manage the LEDs (for details check the file “contiki/core/dev/leds.h”)

- Available LED commands
  - unsigned char leds_get(void);
  - void leds_set(unsigned char leds);
  - void leds_on(unsigned char leds);
  - void leds_off(unsigned char leds);
  - void leds_toggle(unsigned char leds);
- Available LED constants
  - LEDS_GREEN 1
  - LEDS_BLUE 2
  - LEDS_RED 4
  - LEDS_ALL 7

❷ We turn ON only the red LED

Exercise 1

- Create a new application that turns on all the LEDs, as shown in the figure
- Verify the application by running it in Cooja and checking the LEDs status

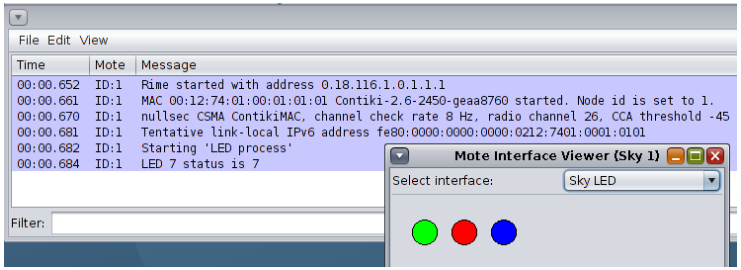


- Hints
  - You can use the function leds_on() and the constant LEDS_ALL
  - Remember to modify the Makefile by adding the new file name to “CONTIKI_PROJECT”

For a possible solution, see the file “iotrain-sim/database/fundamental_training/single_node/actuation_control/led/simulation/solution_led1.c”

Exercise 2

- Create a new application that gets the status for all the LEDs and prints it to the console



- Hints
  - Use the function leds_get() to get the LED status
  - You can use printf() to display information about what is happening in your application
  - If an LED is on, its corresponding bit in the return value will be set to ”1”
  - Remember to modify the Makefile by adding the new file name to “CONTIKI_PROJECT”

For a possible solution, see the file “iotrain-sim/database/fundamental_training/single_node/actuation_control/led/simulation/solution_led2.c”

Exercise 3

- Modify the code below to see what number you get when you turn on more than one LED

```
#include "contiki.h"
#include "dev/leds.h"
#include <stdio.h>
/*---------------------------------------------------------------------*/
PROCESS(led_process, ”LED process");
AUTOSTART_PROCESSES(&led_process);
/*---------------------------------------------------------------------*/
PROCESS_THREAD(led_process, ev, data)
{
  PROCESS_BEGIN();
  leds_on(LEDS_RED);
  printf("LED %u status is %u\n", LEDS_RED, leds_get());
  PROCESS_END();
}
```