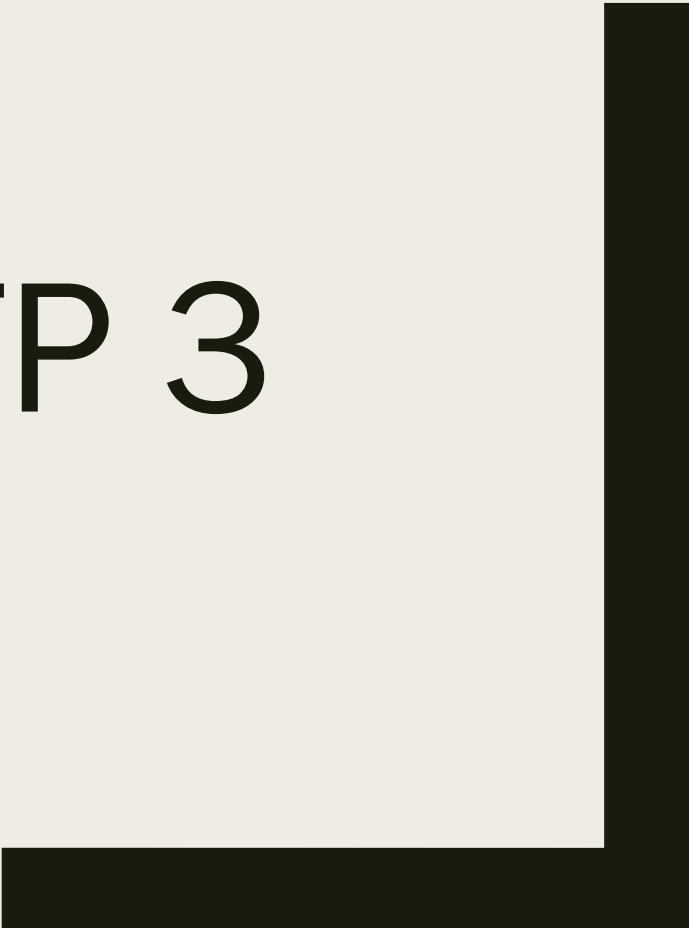




# RÉSUMÉ DU TP 3

Hands on Contiki OS and Cooja  
Simulator



# Démarrer Hello World sur un Tmote Sky node

- 1 – faire au préalable :
  - `$ cd Contiki`
  - `$ cd examples/hello-world`
  - `$ make TARGET=native`
  - `$ ./hello-world.native`
- 2 – Compile Contiki pour la plateforme sky:
  - `$ make hello-world.upload TARGET=sky`
  - `$ make TARGET=sky savetarget`
- 3 – Connecter le Tmote Sky par USB : `$ make sky-motelist`
- 4 – Upload the firmware to the node : `$ make hello-world.upload MOTE=1`
- 5 – Pour voir le programme de sortie : `$ make login MOTE=1`

# Sensors, LEDs and Button

- Contiki utilise des « protothreads » :
  - Attendre un événement : `PROCESS_WAIT_EVENT()`
  - Main process : `PROCESS(nom_process, String)`
- `AUTOSTART_PROCESSES(&nom_process)`
- `PROCESS_THREAD(nom_process, ev, data)`
- `leds_on(LEDS_ALL)` : Allume toutes les leds
- `Leds_off(x)`: Eteind les lumières
- Compteur qui permet de changer de couleur à chaque clique

```
#include "contiki.h"
#include "dev/button-sensor.h"
#include "dev/light-sensor.h"
#include "dev/leds.h"
#include <stdio.h> /* For printf() */
/*-----
PROCESS(hello_world_process, "Hello world process");
AUTOSTART_PROCESSES(&hello_world_process);
/*-----
PROCESS_THREAD(hello_world_process, ev, data) {
    PROCESS_BEGIN();
    int static compteur = 0;
    SENSORS_ACTIVATE(button_sensor);
    SENSORS_ACTIVATE(light_sensor);
    while (1){
        PROCESS_WAIT_EVENT();
        if (ev == sensors_event && data == &button_sensor){
            compteur = compteur+1;
            if (compteur % 2 == 0) {
                leds_off(LEDS_RED);
                leds_off(LEDS_BLUE);
                leds_on(LEDS_RED);
            }
            else {
                leds_off(LEDS_BLUE);
                leds_off(LEDS_RED);
                leds_on(LEDS_BLUE);
            }
            printf("Light: %u\n", light_sensor.value(0));
            char* x = "Hello";
            printf("%s\n", x);
            printf("%i\n", compteur);
        }
    }
    PROCESS_END();
}
```