Sensor Programming Tutorial

Sensors in Tmote Sky

- The Tmote Sky mote has integrated humidity, temperature and light intensity sensors
- Sensors are managed in Contiki OS as follows
  - The library “dev/sht11/sht11-sensor.h” is used to manage the temperature humidity sensors
  - The library “dev/light-sensor.h” is used to manage the light intensity sensor
- Run the simulation by selecting “Sensor Simulation” in the IoTrain-Sim interface, or open the corresponding file in Cooja (“iotrain-sim/database/fundamental_training/single_node/sensing/simulation/sensor.csc”)

Source Code Commentary

- Print data from all sensors every two seconds
  - Source code: iotrain-sim/database/fundamental_training/single_node/sensing/simulation/sensor.c

```
#include "contiki.h"

#include "dev/light-sensor.h"
#include "dev/sht11/sht11-sensor.h"

#include <stdio.h>
#include <math.h>

/*---------------------------------------------------------------------------*/
PROCESS(sensor_acq_process,"Sensor Acquisition");
AUTOSTART_PROCESSES(&sensor_acq_process);
```

Include sensor libraries

Source Code Commentary (cont.)

```
PROCESS_THREAD(sensor_acq_process,ev,data)
{
        static struct etimer et;
        static int val;
        static float s = 0;
        static int dec;
        static float frac;
        PROCESS_BEGIN();
        printf("Starting Sensor Example.\n");
        while(1)
        {
                etimer_set(&et, CLOCK_SECOND * 2);

                SENSORS_ACTIVATE(light_sensor);
                SENSORS_ACTIVATE(sht11_sensor);

                PROCESS_WAIT_EVENT_UNTIL(etimer_expired(&et));
                val = sht11_sensor.value(SHT11_SENSOR_TEMP);
                if(val != -1)
                {
                        s= ((0.01*val) - 39.60);
                        dec = s;
                        frac = s - dec;
                        printf("\nTemperature=%d.%02u C (%d)\n", dec, (unsigned int)(frac * 100),val);
                }
```

// Set timer to repeat the iterations every 2 seconds

Activate light_sensor to measure the light intensity and sht11_sensor to measure temperature and humidity

// Get the actual sensor value

// Calibrate the sensor value by doing some calculation

Source Code Commentary (cont.)

```
                val=sht11_sensor.value(SHT11_SENSOR_HUMIDITY); if(val != -1)
                {
                        s= (((0.0405*val) - 4) + ((-2.8 * 0.000001)*(pow(val,2))));
                        dec = s;
                        frac = s - dec;
                        printf("Humidity=%d.%02u %% (%d)\n", dec, (unsigned int)(frac * 100),val);
                }
                val = light_sensor.value(LIGHT_SENSOR_TOTAL_SOLAR);
                if(val != -1)
                {
                        s = (float)(val * 0.4071);
                        dec = s;
                        frac = s - dec;
                        printf("Light=%d.%02u lux (%d)\n", dec, (unsigned int)(frac * 100),val);
                }
                etimer_reset(&et);
                SENSORS_DEACTIVATE(light_sensor);
                SENSORS_DEACTIVATE(sht11_sensor);
        } //end of while
    PROCESS_END();
}
```

Deactivate all the sensors

Exercise

- Create a sensor application which includes a button that, when pressed, makes the application display the value of the light sensor
- Verify the program by running it in Cooja and checking the console output when the button is pressed
- Hints
  - Remember to modify the Makefile by adding the new filename to “CONTIKI_PROJECT”
  - You can check the following file for a possible solution: “iotrain-sim/database/fundamental_training/single_node/sensing/simulation/button-light-sensor.c”