Ad Hoc and Sensor Networks Exercise 4

Sensor Network programming
TinyOS exercise

- Sensor network programming in a nutshell
  - Read a tutorial
  - Solve two (simple) tasks on real hardware
- Lab-style exercise
  - Teams of two to three students are ideal
  - Two parallel lab working places are available
  - Reservation system on the course website
- Expected time needed for all tasks: 3-4 hours
- Mandatory to get the testat without taking the exam
Tutorial

- **Carefully** read the tutorial on the TinyOS website
  - http://www.tinyos.net/tinyos-1.x/doc/tutorial/index.html
  - Ignore instructions about setting up the system, flashing applications, simulation, and data ROM access

- The tutorial contains several exercise tasks: Think about them but **do not write code**

- We use an **Eclipse plug-in** to develop the applications which is not mentioned in the tutorial
Exercise 1

• Exchange of a sensor reading
  – Two sensor nodes are used in this task
  – One node periodically samples its light sensor and broadcasts the reading over its radio
  – The other node listens for radio messages and signals if it is getting brighter or darker
    – Brighter -> The green LED of the receiver is set
    – Darker -> The red LED of the receiver is set
    – No significant change -> The yellow LED is set

What exactly do we get from the sensor?
Exercise 2

• **Optical Communication using Morse Codes**
  
  – **Sender**
    – Sensor node connected to the pc over a serial connection
    – Controlled by a (provided) java application
    – LEDs are toggled on/off to transmit Morse signals
  
  – **Receiver**
    – Sensor node sampling its light sensor. Detects and decodes Morse signals
    – Connected to a pc over a serial connection. On the pc the Hyperterminal is used for output
Exercise 2

- **Optical Communication using Morse Codes**
  - **Sender**
    - Sensor node connected to the pc over a serial connection
    - Controlled by a (provided) java application
    - LEDs are toggled on/off to transmit Morse signals
  - **Receiver**
    - Sensor node sampling its light sensor. Detects and decodes Morse signals
    - Connected to a pc over a serial connection. On the pc the Hyperterminal is used for output
Exercise 2

- **Optical Communication using Morse Codes**
  - **Sender**
    - Sensor node connected to the pc over a serial connection
    - Controlled by a (provided) java application
    - LEDs are toggled on/off to transmit Morse signals
  
  - **Receiver**
    - Sensor node sampling its light sensor. Detects and decodes Morse signals
    - Connected to a pc over a serial connection. On the pc the Hyperterminal is used for output

![Diagram showing the setup of the sender and receiver using optical communication through Morse codes.](image_url)
General Information

• Code Skeletons for both applications are provided
  – Exercise 1 only needs very little additional programming and should be solved by all groups
  – Exercise 2 is more challenging but is also more fun

**Hint:** Exercise 1 may contain helpful code fragments
Setup

- The lab computers are preinstalled with all necessary tools and the exercise skeletons.
- An Eclipse plug-in for TinyOS development is installed and configured. Check the following website for a quick start on how to use it: http://dcg.ethz.ch/projects/tos_ide/
Compiling

- For all Exercise applications two compile commands are defined.
  - `Target tinynode` is used to compile the application. The compiler output is directed to the Eclipse console.
  - `tinynode install` is used to compile and flash the application to a sensor node.
    - A popup asks for a `bsi` number. This is the identifier of the serial port the sensor node is attached to. The identifier count starts at 0. Therefore COM1 == 0, COM2 == 1, COM3 == 2.
Final Remarks

- The lab is in the ETL building. Hardware and keys must be fetched in our office ETZ G64.1.
- If you get stuck come back to ETZ G64.1 and ask for help.
- A FAQ page will be linked from the course website and updated regularly.