**Objectif de ce labo :**

Le but de ce document est de vous familiariser avec l’installation et l’utilisation d’un réseaxu de capteurs sans fil (appelé en anglais Wireless Sensor Network ou encore WSN). Les composants hardware d’un réseau sans fil sont les capteurs (sensors), les nœuds (motes), les gateways et les ordinateurs, connectés comme dans l’image ci-dessous.



Les capteurs sont branchés sur les nœuds. Les nœuds, que nous appellerons aussi Motes, servent à alimenter les capteurs et à en recueillir les résultats des mesures. Les nœuds sont alimentés avec 2xAA piles chacun. En utilisant des antennes filaires, les nœuds transmettent leurs données au nœud-puit. Ce nœud-puit est identique aux autres nœuds, mais, à différence des autres, il est branché sur un Gateway. Le Gateway sert à établir une connexion avec un ordinateur par son port USB.

Nous utiliserons le système d’exploitation Tinyos pour notre WSN.

Le text qui suit est donné en anglaise. Comme vous le savez, cest très souvent dans cette langue que les informations techniques sont échangées.

**Tiny-OS Installation and Hello World app**

1. Install an Ubuntu virtual machine.
2. Tell *apt* about the TinyProd Signing Key.

 $ wget -O - http://tinyprod.net/repos/debian/tinyprod.key | sudo apt-key add -

1. Add the following lines to /etc/apt/sources.list.d/tinyprod-debian.list:

 deb http://tinyprod.net/repos/debian stretch main

 deb http://tinyprod.net/repos/debian msp430-46 main

 $ sudo -s

 $ cd /etc/apt/sources.list.d

 $ echo "deb http://tinyprod.net/repos/debian stretch main" >> tinyprod-debian.list

 $ echo "deb http://tinyprod.net/repos/debian msp430-46 main" >> tinyprod-debian.list

 $ exit

1. Install the new packages:

 $ sudo apt-get update

 $ sudo apt-get install nesc tinyos-tools-devel msp430-46 avr-tinyos

1. Download the tinyos-main from:

<https://github.com/tinyos/tinyos-main>

1. Edit .bashrc file

export TOSROOT="<local-tinyos-path>"

export TOSDIR="$TOSROOT/tos"

export CLASSPATH=$CLASSPATH:$TOSROOT/support/sdk/java:$TOSROOT/support/sdk/java/tinyos.jar:.

export MAKERULES="$TOSROOT/support/make/Makerules"

export PYTHONPATH=$PYTHONPATH:$TOSROOT/support/sdk/python

1. In order to program the motes, you will need to access the serial ports. This can be done by joining the group that grants this privilege.

 sudo gpasswd -a <your-user> dialout

1. The environment should be ready now. To test it, open a new console and enter the following commands:

$TOSROOT/apps/Blink

cd $TOSROOT/apps/Blink/

make telosb

1. Restart the machine to let the “user privilege” go on.
2. Update to the latest version of pySerial
3. Modify tos-bsl as follows:

Remove:

self.serialport.setBaudrate(baudrate)

Add:

#handle pyserial 2.x and 3.0.1

try:

 self.serialport.setBaudrate(baudrate)

except AttributeError:

 self.serialport.baudrate = baudrate

Try as follows to install your app on device:

make telosb install.1 bsl,/dev/ttyUSB0

Check if your Blink application works properly and show the final outcome to the lab instructor.

In the next step, try to analyze the Blink application code and get familiar with the way it works. In the next lab, we will get more familiar with the structure of the code.