

CS 4222  
Programming Assignment I

January 22, 2013

## Important Dates

Due: Feb 6, 23:59

Late Penalty: 10% per day

## TA Office Hours

TA: Manjunath D.

Email: [doddaven@comp.nus.edu.sg](mailto:doddaven@comp.nus.edu.sg)

Following is my office hours for consultation regarding this assignment at the Data Comm Lab 1, Basement, COM1. However, you can meet me on other days too, but please send me an e-mail before you start towards Data Comm Lab 1.

- Jan 30, 4pm to 5:30pm
- Feb 4, 4pm to 5:30pm

## Overview

The assignment aims to warm you up for more programming using TinyOS/NesC. It involves writing a simple program that turns a desired LED(s) ON/OFF for a desired duration of time. Students will mainly learn the following.

- Basic programming using TinyOS/NesC
- Compilation and downloading of TinyOS code onto the motes
- Mote to PC and PC to mote communication

## Tasks

- 1 PC-side programming — Program a client that communicates a few input parameters to a mote. The input parameters should specify one or more desired LEDS, corresponding flags indicating the desired action (ON/OFF), and time durations in seconds for which the LEDS should stay ON/OFF. Moreover, the client should also be able to display messages received from the mote.

- 2 Mote-side programming — Mote program should receive input parameters from the client and immediately send an acknowledgement packet (ACK). Then control LEDS as instructed by the input parameters.

For example, the command “./client RED, GREEN, ON, ON, 1s, 2s” indicates that RED LED must be turned ON for a duration of one second and GREEN LED must be switched ON for a duration of two seconds. If the LEDS are already ON then reset their ON duration.

## Submission

Please submit your code to IVLE workbin as a single zip/tar file. The zip/tar file should be called *CS4222\_PA1\_USERNAME.tgz*, where USERNAME is your student ID. The submitted code must satisfy the following.

- 1 The code should be compilable.
- 2 Please include a README file explaining how to compile and execute your code.

If you submit multiple versions, only the last version will be graded. If the last version is a late submission, marks will be deducted based on the date of submission.

## Grading

Points are allocated as follows:

- 25 points — correct choice of components and compilation
- 25 points — PC to mote communication
- 25 points — mote to PC communication
- 25 points — desired control of LEDS

**Weightage towards final assessment is 5%**