Dynamic LPL for wireless sensor networks

Ciaran McGirl. Supervisor: Ciaran McGoldrick

Introduction

With increased research in wireless sensor networks in recent years, various strides towards improving the technology have been made. However, the main obstacle over the years has been the power consumption of the nodes.

In its early stages, TinyOS considered this problem, and originally a long preamble was used to notify the receiver of an message about to be sent. (Shown Below)

This project looks to improve this preamble by allowing messages of different types to dictate how long the node goes to sleep, essentially allowing for the sending of a lot of data over a short period before returning to sleep.

Low Power Listening (LPL)

Developed within TinyOS and involved the sender using a long preamble before the message so that the receiver knew the transmitter was sending.

This preamble was shortened and the repeated sending of short messages proved more efficient.

The main steps of LPL are:
1. Receiver remains asleep for most of the time, occasionally waking to determine if the channel is idle or busy.
2. Sender wishes to send a message to the receiver, it attaches a preamble to the message (a wake up signal). After the preamble signal, the sender sends the message.
3. When the receiver wakes up, it samples the channel:
   i. If the channel is idle, the receiver sets the next wake-up time and sleeps again.
   ii. If the channel is busy (preamble detected), the receiver stays on until the message is received. After transmission, the receiver sets its next wake-up time and sleeps again.

Wireless Sensor Networks

The above diagram gives a sense of a typical sensor network and how it operates. Information is sensed from the environment and passed to the base station. Originally, wireless sensor networks were used in military applications in remote regions that were difficult to access. They have since spread into the commercial world, but still are used in similar conditions. Deployment in remote regions means that once a sensor network is laid out, it may not be disturbed for a long period of time. Power consumption within the nodes is, therefore, a priority and always an area to improve.

TinyOS was used in this project is a component based OS written in NesC, it is specifically designed for wireless sensor networks with limited hardware.

Typical Wireless Sensor Network

M.Sc. in Computer Science

(Networks and Distributed Systems)