**SensorNet**

*Wireless Home Automation and Sensor Network*

Matthew Hengeveld

Computing & Computer Electronics

Directed Research Project

Nora Znotinas

Fall 2014/Winter 2015

Overview

SensorNet is a wireless network designed to interface sensors and automation controls to a central control system, accessible from local networks and the World Wide Web. Using custom, open source code running on multiple microcontroller architectures, sensors and automation controls, called nodes, can interact with their environment, and many different electronic devices in the home. Monitoring and control of nodes is facilitated by the central control system, called the root, through which several interfaces are available. Inexpensive wireless transceivers allow for a low cost per node, as well as a low-latency network that utilizes the same frequency band as WiFi, but does not interfere.

Background

Currently, there are several systems on the market that offer similar features as SensorNet provides. Companies such as Belkin, D-Link, Insteon, Nest, Philips, and Skylink each have product lines designed to offer remote access to sensors and automation controls in the home, all in limited forms. Products like the Nest Learning Thermostat and the Philips Hue are both specialized systems, and are limited to one function. Other product lines are offered by companies such as Belkin, Insteon and SkylinkHome that offer a broader set of features. These product lines typically offer only controllable mains voltage plugs, light dimmers and thermostats, as well as a few other specialized products. As well, most of these products rely on each device having its own WiFi connection to the home network. This causes issues with access points not able to handle a high number of connections, excess network traffic, and wireless interference for the whole wireless network.

The 802.11 standard (WiFi) was designed to make high data-rate transfers between computers and network infrastructure. WiFi is expensive to implement, and adds significant cost to low data-rate devices without the requirements of the advanced features that the protocol provides. Integrating WiFi with microcontrollers is expensive – current solutions available include:

|  |  |
| --- | --- |
| Arduino WiFi shield | $80 |
| Arduino Yun | $70 |
| XBee | $35 |
| Intel Galileo | $75 |

Since the beginning of this project, a new WiFi module with serial communication has come on to the market, and currently can be found for close to $4. While this is a significant reduction in cost compared to the previously mentioned solutions, it is still close to twice the cost of the wireless transceiver used in SensorNet. As well, this module does not address the latency, interference and increased number of connections associated with WiFi.

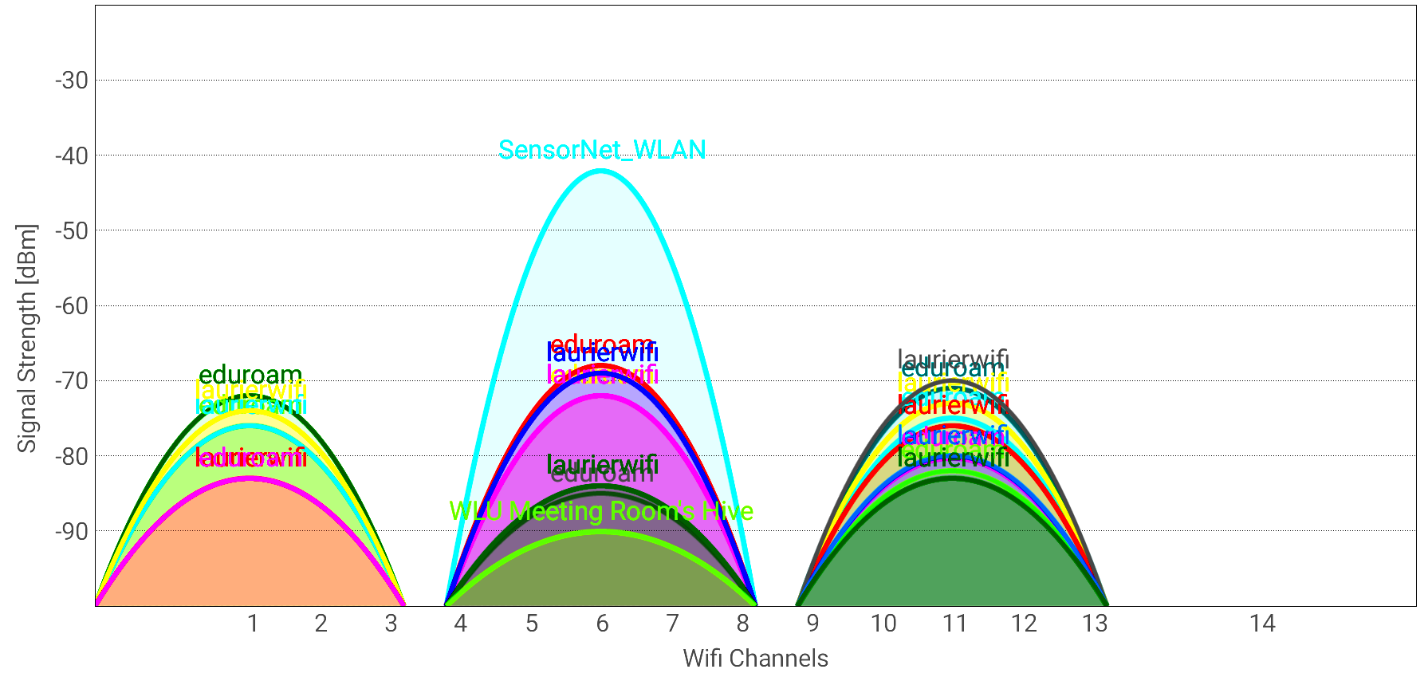
WiFi was created for transfer of large amounts of data, and comes with advanced, complicated protocols and the significant overhead that accompanies them. Sensors and automation controls have inherently small data requirements. Sensors, such as temperature, humidity, light and hall sensors, typically output from a single bit (on/off) to 24 bits (RGB format colour) of data. Similarly, automation controls typically require from a single bit (on/off) to 24 bits (PWM control of a RGB light). With SensorNet using two bytes for addressing, and one byte for error detection – for a total of about 4-6 bytes –the 802.11 standard has a typical packet size 250 - 375 times greater than that used in SesnorNet.[[1]](#footnote-1)

Bluetooth, and other RF transceivers each have their own disadvantages that make them unsuitable for home automation and sensors: signal range, lack of networking capabilities, complexity, interference and lack of noise rejection.

Network

SensorNet uses two networks: Wifi, and a custom-designed network protocol for communicating with nodes. The network is based on the Nordic nRF24L01+ 2.4GHz transceiver.

The nRF24L01+ (nRF) operates on the same license-free ISM (Industrial, Scientific, Medical) wireless band as WiFi and Bluetooth, but uses a bandwidth of only 1MHz, compared to 20/40MHz of WiFi. WiFi uses orthogonal frequency-division multiplexing (OFDM), which reduces interference with the nRFs Gaussian frequency-shift keying modulation. Over the 2.4GHz to 2.5GHz frequency range, WiFi typically operates on channels 1, 6 and 11, so that the maximum number of WiFi channels can be utilized without overlap. This leaves 40MHz of bandwidth for use by SensorNet networks (Figure 1)[[2]](#footnote-2) and means that multiple different SensorNet networks can co-exist with WiFi networks, without using the same channels, and reducing possible interference between them. Bluetooth, however, uses the same Gaussian frequency-shift keying as the nRF, but employs advanced frequency-hoping techniques to change its frequency up to 1600 times a second. This frequency-hoping allows Bluetooth to always find a clear channel.



*Figure 1. WiFi channels. Actual graph of WiFi networks on Laurier campus (Bricker Academic). Each channel below 13 has 5MHz spacing.*

The network protocol was designed in a star topology, where the root connects to each node individually. The nRF transceiver can only operate in half-duplex. Therefore, the network protocol uses a request-reply pattern. The central control, the root, sends requests to nodes. This pattern is necessary so that interference does not occur between nodes.

The root and each node have a unique 4-byte address. The protocol also has some advanced features, provided by the Nordic nRF “Enhanced Shockburst” hardware on each transceiver. These features include automatic 1-byte CRC check for each packet, dynamic packet payload length, and automatic acknowledgement of received packets. The protocol for SesnorNet was designed to reduce the on-air time for each packet sent (and thus the chance of in-air packet collision and interference), as well as insuring that each packet is recieved.

Typically, a sensor or automation control needs less than the 32 byte maximum of one packet. The makeup of a packet is as follows:

Control Software

The control software for SensorNet consists of two parts. The first part if the interface software, which consists of a webserver utilizing Flask

1. Given an Ethernet packet size of 1500 octets. http://en.wikipedia.org/wiki/IEEE\_802.11 [↑](#footnote-ref-1)
2. http://en.wikipedia.org/wiki/List\_of\_WLAN\_channels [↑](#footnote-ref-2)