Master's Thesis

# An evaluation of low power, low-rate wireless data communication technologies for battery powered sensor networks

Einar Vading
Gustav Enander


Department of Electrical and Information Technology,
Faculty of Engineering, LTH, Lund University, June 2014.

# An evaluation of low power, low-rate wireless data communication technologies for battery powered sensor networks

Einar Vading and Gustav Enander

Department of Electrical and Information Technology
Lund University

Advisors: Marcus Johansson (Axis), Jon Hansson (Axis) and Stefan Höst (EIT)

June 23, 2014

Printed in Sweden
E-huset, Lund, 2014

# Abstract

One major cost driver in sensor system installations is the cost of labor. With wireless sensors the installation cost can be lowered since no cables needs to be routed. For the total cost of ownership to be low, however, the battery life of the sensor must be long. A wireless sensor system is more susceptible to interference than a wired counterpart and since the transmission is done over the air, it is easier to intercept and possibly forge communication. For this thesis work, a wireless sensor prototype was developed with the aim of being both secure and low power. We found that it was possible to achieve a battery life of at least 5 years with a range of about 30 m and message authentication.

# Executive summary

## I. Introduction

Wireless sensors are becoming increasingly popular, mainly because of simplified installation since there is no need for laying cables. Typical wireless sensors measure temperature, humidity, atmospheric pressure or can be used as alarm sensors. Wireless sensor systems imply low installation costs, especially in large systems where many sensors need to be connected.

There are however demands on wireless sensor systems that do not exist for wired systems. Long battery lifetime is probably the most important demand – if there should be any long term benefit from the simplified installation, battery life must be long enough.

There are also security issues that do not exist for wired systems, for example the risk of someone interfering or jamming the wireless communication, be it deliberately or by mistake.

The goal of this work was to develop a sensor prototype for measuring temperature with focus on long battery lifetime, good range and secure communication.

## II. Existing technologies

There are several technologies that are suitable for wireless sensors. ZigBee, Bluetooth and Z-wave are examples of some, that can be used for this type of application. These technologies include both hardware specifications for the electronics and software for handling the wireless communication.

A common property of many wireless systems is that they utilize license free frequency bands. There are three license free frequency bands available in Europe, these are: 433 MHz, 868 MHz and 2.4 GHz (for example used by wireless internet, Wi-Fi). Another feature of the existing technologies is that as little data as possible is transferred, to minimize power consumption.

It was decided not to use any of the existing techniques for this prototype, and instead develop a new solution using our own software together with a pre-fabricated radio chip. This gave the possibility to further minimize power consumption by tailoring the hardware and software to fit the exact needs of the application.

## III. Prototype design

The prototype sensor works at 868 MHz, which is one of the license free frequency bands that is available in Europe. Similar frequency bands exist in most part of the world, for example 915 MHz that is available in the US and can be used without hardware modifications.

The choice of frequency is important as it directly affects the maximum possible communication range – higher frequency results in shorter range. The data rate is also frequency dependent, a higher frequency results in a higher possible data rate. Measurements and tests concluded that 868 MHz was the best choice for this application, mainly because of the good range. There are also regulations that limit the maximum output power and transmission time on this frequency band which minimizes the risk of interference from other systems.

To minimize power consumption the sensor spends most of its time in a low-power sleep mode, consuming only about 1.3 µA. This corresponds to the energy produced by a drop of water falling from 1 cm height! The sensor then wakes up periodically (for example every 60 s) to announce itself. It also wakes up for events, for example at a defined relative change in temperature. The time in active mode is much more power consuming than when waiting and it should therefore be kept to a minimum.

The prototype's main hardware consists of a microprocessor (MCU) and a radio chip. Both the MCU and radio chip were chosen to minimize power consumption, while still being secure and reliable. The sensor is powered by a small button cell battery (CR2032).

## IV. Verification

The manufactured sensor's performance was evaluated by measuring power consumption and maximum possible range. Range was measured in two typical office environments, one with a majority of gypsum walls and the other with a majority of thick concrete walls.

## V. Results and conclusions

Results conclude that the manufactured prototype sensor is working well and should be able to reach a battery lifetime of about 5 years.

The choice of MCU and radio chip is very important for maximizing battery lifetime. Making the software as efficient as possible in order to minimize the power consuming active time is also crucial.

Maximum indoor range is highly dependent of the environment. Tests show that concrete walls result in much higher attenuation of the radio signal and thus a much shorter achievable range. A maximum range of 20 m to 30 m is however possible in most indoor locations.

The developed security solution makes it impossible to send fake messages. Thanks to the sensor's function of announcing itself every 60 s, any attempt to jam the data communication will be discovered.

This work concludes that wireless sensors are a good alternative to wired sensors. Microcontrollers and radio chips are being further developed, which can provide even better battery lifetime. Other technologies such as energy harvesting, for example by using solar cells, could also be used for further extending battery lifetime.

# Acknowledgements

We would like to thank our supervisors Marcus Johansson and Jon Hansson at Axis Communications who both provided us with great support and a great deal of inspiration. We would also like to thank our examiner Stefan Höst at LTH who has given us new perspectives for this thesis as well as help with administrative details. A big thanks also goes out to all the employees at Axis that have helped us and answered all our questions.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1
# Introduction

## 1.1 Background

There are many different technologies for transmitting data using radio, such as Bluetooth, ZigBee, Z-Wave, Wi-Fi etc.

This thesis project will specify the requirements for a suitable data link for battery powered sensor nodes communicating to a central node. Different technologies will be evaluated with respect to the requirements and an appropriate technology chosen. A prototype using the chosen technology will then be designed, manufactured and evaluated. Focus will be on the parameters energy consumption (battery lifetime), range, cost and data integrity.

One possible use of wireless sensors could be supervision of shipping containers. By collecting temperature and/or humidity data, perishable goods could be supervised and an alarm could be raised in the event that the environment changes. In those cases it could be of interest to verify that the data is correct to make the system less susceptible to sabotage.

The end goal of this thesis is to interface the sensors to an existing Axis product and since the New Business division of Axis works extensively with the A1001 access controller it was chosen as a platform for experimentation. Transferring the results to any other Axis product that runs on the same platform should be simple and straight forward.

## 1.2 Specifications

In this section the requirements for the sensor's communication link will be specified. The specifications are not absolute but more of guidelines as to what kind of performance that is desired. The security specifications are especially loose and depend a lot on the intended use case of the sensor.

### 1.2.1 Battery lifetime

To ensure low maintenance costs it is desirable to maximize battery life. An arbitrary goal of five years is thought to be reasonable while down to three years can be acceptable. Five years implies an average current draw of 5.7 µA using a single 250 mAh coin cell battery.

### 1.2.2 Range

Since each inexpensive sensor will communicate with a more expensive base station it is desirable that each base station is within range of a "reasonable" number of sensors while still limiting sensor output power and with that, power consumption. Here, a "reasonable" number is taken to mean enough sensors as to make the overall installation cost low while still limiting the number of sensors so that the frequency band is not over utilized. Since the application is not known, this range is arbitrarily taken to be 20 m to 30 m in an indoor environment.

### 1.2.3 Security

The chosen technology should be insensitive to frequency band congestion and deliberate attacks, such as man-in-the-middle. It is desirable that a sensor node is able to announce itself to the main unit at regular intervals so that interference can be detected. The sensor node should also be able to sign any data originating from it to make it harder for a hypothetical attacker to replay captured frames and thereby being able to manipulate the main unit.

### 1.2.4 Data

The amount of data that is to be transmitted will be low. For a typical sensor only state changes and heartbeats are absolutely necessary. Low battery indication and transmitted signal level are features that can be used to improve usability of the device.

### 1.2.5 Usability

If the sensors need to be paired with the base station, the pairing should be easy and intuitive for the operator. It should not be necessary to repeat the pairing after a power outage, whether it occurs at the sensor or at the base station.

If possible an open standard is desirable, making the system easier to extend for end users.

### 1.2.6 Production effort

The chosen technology should be legal to use worldwide with no or minor modifications to the hardware. The sensor should also be inexpensive to manufacture, meaning that part count and complexity should be kept low.

# Chapter 2
# Theory

There are a large number of standards for wireless communication in existence today. This chapter will briefly describe some of the standards that may be suitable for wireless sensor networks. A common property of these standards is that they are designed for low power operation. To achieve this goal all standards use cyclic sleep schemes and send as small amounts of data as possible, thus minimizing the most current consuming awake time.

A brief explanation of channel access methods and modulation techniques used by these wireless standards is also presented. A more thorough presentation of these methods and techniques can be found in e.g. *Digital Communications* by J. Proakis [1].

## 2.1 Multiple access methods

When several devices need to share the same medium, i.e. the same frequency band, there needs to be some way for them to handle possible collisions. These techniques go by the collective name "media access control" and come in various complexities. In this section two of the more common media access control methods will be presented.

### 2.1.1 ALOHA

When using ALOHA the device sends whenever it likes without first sensing the shared medium. If no ack is received within a time-out period the device waits a random amount of time and tries again. ALOHA is hence used for lightly loaded networks where the risk of collision is low [2].

The probability $p$ of a successful transmission when using ALOHA can be calculated using

$$p = e^{-2G} \tag{2.1}$$

where $G$ is the number of transmission attempts per frame-time (i.e. the time needed to send one frame).

### 2.1.2 Carrier Sense Multiple Access

Carrier Sense Multiple Access (CSMA) is a method where the shared medium is sensed by the transmitter before attempting to use it, "listen before talk", thereby reducing the risk of collision. If the shared medium, in this case the radio channel, is occupied the device waits a random time before it tries to access the medium again. The method is suitable for networks where the shared medium is heavily loaded and the risk of collisions is high.

CSMA can be improved by adding Collision Detection (CSMA/CD) or Collision Avoidance (CSMA/CA) which further increases the chance of a successful transmission.

## 2.2 Modulation techniques

This is a brief description of some modulation techniques, i.e. methods to encode an analog radio carrier to transmit a digital signal. A more thorough presentation of these techniques can be found in *Digital Communications* by J. Proakis [1].

### 2.2.1 Frequency Shift Keying (FSK)

Frequency Shift Keying (FSK) is a technique for encoding data by modulating the frequency of the carrier while the amplitude and phase of the carrier are kept constant.

#### Binary Frequency Shift Keying (BFSK)

BFSK uses two different frequencies, one to represent 0 and one to represent 1. A frequency $f_d$ is added to or subtracted from the carrier frequency $f_c$. The resulting frequency $f_c + f_d$ will represent one of 0/1 and $f_c - f_d$ will represent the other.

#### Gaussian Frequency Shift Keying (GFSK)

GFSK uses the same principle as BFSK with the extension that it utilizes a Gaussian filter to smooth the transitions between the two different carrier frequencies. First the frequency is modulated using discrete frequency steps, then the resulting modulated frequency is low-pass filtered using a Gaussian filter. That way, the transitions between the two frequencies becomes smoother, resulting in a signal that has a narrower spectral band since the high frequency content of the discrete frequency steps is filtered out. The smooth frequency changes also consume less power and allow the use of cheaper electronics since the unwanted out of band emissions are lowered [3].

### 2.2.2 Phase Shift Keying (PSK)

The phase of the carrier can be modulated to represent two or more different data elements. The amplitude and frequency of the carrier are kept constant. The simplest PSK method is binary PSK (BPSK) where one data element is represented by 0° phase shift and the other is represented by 180° phase shift.

#### Quadrature Phase Shift Keying (QPSK)

QPSK uses two separate BPSK modulators; one in phase and the other 90° out of phase (quadrature). Two bits from the original signal are sent to the BPSK modulators and the results are added. This results in a signal that has four possible phases: 45°, −45°, 135° and −135°. Hence every phase shift represents a combination of two bits, and the baud rate can be lowered while maintaining the same bitrate.

### 2.2.3 Spread Spectrum

Spread spectrum is a method to let many users share a single channel. It can also give some protection against jamming signals, eavesdropping, and interference from other stations.

Spread spectrum expands the bandwidth of the original signal to enable more users to share the same spectrum at the same time. The expansion is achieved using pseudo-random spreading codes that are identical at the transmitting and receiving ends of each user, but where the codes of the different users are orthogonal to each other. Given the orthogonal spreading codes, two signals can share the same spectrum and then be decoded at their respective receiving ends without interfering with each other. The orthogonality and randomness of the code will make signals that are spread with different codes look like noise with equal intensity over the entire spectrum, while signals spread with the same code will look like a single well defined frequency component. There are several techniques for spreading the bandwidth, two of the more common being Direct Sequence Spread Spectrum (DSSS) and Frequency Hopping Spread Spectrum (FHSS) [2].

#### Direct Sequence Spread Spectrum (DSSS)

DSSS spreads the original signal by replacing each data bit with a sequence of $n$ bits. The sequence is determined by modulating the original signal with a pseudo-random generated spreading code. The required bandwidth of the signal will be $n$ times the original signal bandwidth [2].

#### Frequency Hopping Spread Spectrum (FHSS)

FHSS has a different approach; $n$ different carrier frequencies are used and modulated by the signal. The carriers are modulated one at a time in an order determined by a pseudo-random sequence. This way, the bandwidth of the original signal is spread to the total bandwidth of the $n$ carrier frequencies [2].

## 2.3 IEEE 802.15.4

IEEE 802.15.4 is a standard that defines the physical and MAC (media access control) layers in the OSI-model for low-rate wireless personal area networks (LR-WPANs). IEEE 802.15.4 is designed to be used for simple, reliable, low-cost communication networks with low power consumption (i.e. battery powered devices).

The physical layers support a variety of frequency bands, for example:

- 868–868.6 MHz (Europe, 1 channel, 20 kbps)
- 902–928 MHz (US, 10 channels, 30 kbps)
- 2400–2483.5 MHz (Worldwide, 16 channels, 250 kbps)

The 868 MHz and 902 MHz bands use BFSK and DSSS. The 2.4 GHz band uses QPSK and DSSS.

Networks using 802.15.4 can operate using two different topologies; star topology or peer-to-peer topology (Figure 2.1). Using star topology all nodes communicate through a central node, called a PAN (Personal Area Network) coordinator. In the peer-to-peer topology all nodes can communicate directly to another node as long as it is within range. The peer-to-peer topology can be used to create advanced network structures, such as large cluster-tree networks.

Nodes can either be Full Function Devices (FFD) or Reduced Function Devices (RFD). FFDs are typically mains powered and have the capability to route traffic between nodes.

RFDs are simple, battery powered end point devices that does not need to transfer large amounts of data, e.g. a simple switch or sensor. RFDs cannot route traffic between nodes. IEEE 802.15.4 can be configured to use CSMA/CA or ALOHA for multiple access [4].

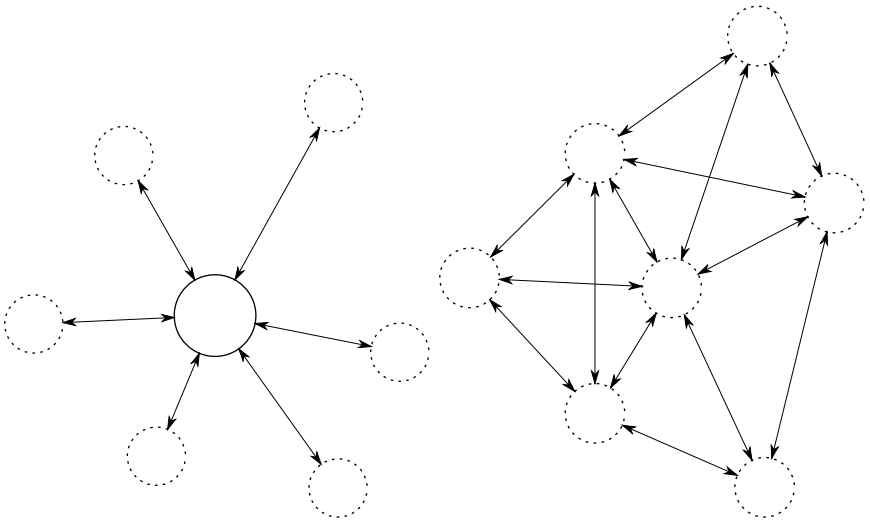

**Figure 2.1:** Network topologies in IEEE 802.15.4 with star topology on the left and peer to peer topology on the right. Dashed circles represent reduced function devices and stroked circles represent full function devices.

## 2.4 Bluetooth Low Energy

Bluetooth Low Energy (BLE) is a technology designed for low-power, short-range wireless communication. BLE uses the 2.4 GHz frequency band with 40 channels and 2 MHz channel spacing. Three of the channels are advertising channels that are used for device discovery, connection establishment and broadcast transmission. The remaining 37 channels are used by connected devices for node-to-node communication [5].

To avoid interference BLE uses adaptive Frequency Hopping Spread Spectrum (FHSS), in which a new channel is used for every new data event. Channels where there is a lot of interference can dynamically be excluded from the hopping scheme. The modulation technique used is Gaussian Frequency Shift Keying (GFSK).

A BLE node can either be a master or a slave. A network containing one master and a number of slaves is called a piconet, a type of star topology. A node can only belong to one piconet [6].

The range of BLE is usually some tens of meters. Maximum theoretical throughput is 236.7 kbps [5].

### 2.4.1 Security

BLE relies on CCM (Counter with Cipher Block Chaining Message Authentication Code) for both encryption and signing on the Link Layer [7]. CCM is proven to be secure given that the underlying block cipher is secure [8]. For BLE the underlying cipher is AES-128 which is considered safe enough to be used by U.S. American federal agencies to protect sensitive data [9], something that testifies to its security.

To exchange keys used for encryption and authentication Bluetooth devices needs to be "paired" to each other. BLE supports three different ways of pairing devices: JW (Just Works), PE (Passkey Entry) and OoB (Out of Band). Of these three so called "association models" only OoB provides protection against both eavesdropping and Man In The Middle (MITM )attacks and only if the OoB method provides protection in itself. If no attacker is present during pairing using either JW or PE, the following communication can be safely encrypted and thereby secured against eavesdropping [7].

## 2.5 ZigBee

ZigBee is a wireless communications standard that is built upon the original (2003) release of the IEEE 802.15.4 specification. It aims to be a reliable, cost-effective, low-power and secure standard [10].

The IEEE 802.15.4 standard (as of 2003, which ZigBee uses) has two physical layers: one for the 868/915 MHz frequency band and one for the 2.4 GHz frequency band. ZigBee builds upon this and provides the higher network and application layers.

One of the key features of ZigBee is the support for mesh networks. A typical ZigBee mesh network can be seen in Figure 2.2. ZigBee does not use the terms FFDs and RFDs to describe nodes. Instead the nodes can be either a ZigBee

Endpoint (ZED), a ZigBee Router (ZR) or a ZigBee Coordinator (ZC). A network can contain only one coordinator but can contain several routers and endpoints.

Coordinators are responsible for forming the network and setting up routing tables. A router's main function is simply to route traffic between nodes. Coordinators and routers needs to be awake most of the time and are because of this often mains powered.

Endpoints cannot route traffic and only communicate with their parents, i.e. a router or coordinator. They are generally low-power, due to the fact that they sleep most of the time, and often battery powered. The maximum range is about 100 m line-of-sight [6].

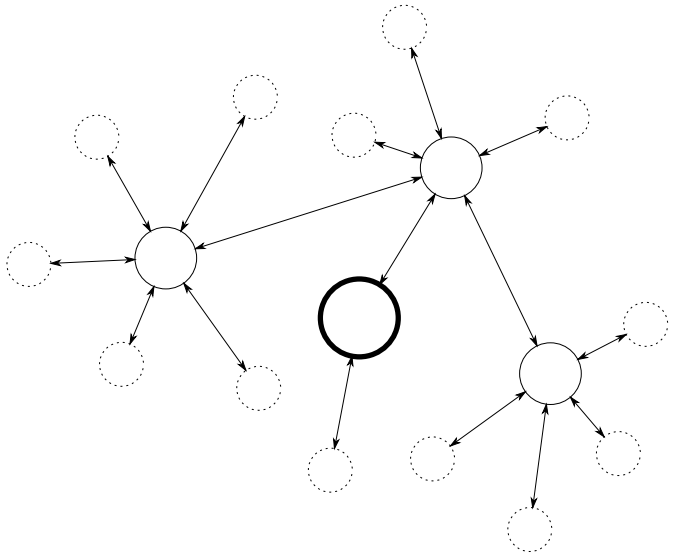

**Figure 2.2:** ZigBee mesh network with ZigBee Endpoints, dashed circles, ZigBee Routers, smaller unbroken circles, and ZigBee Coordinator, larger thick circle.

### 2.5.1 Security

ZigBee uses a similar scheme for encryption to Bluetooth called CCM*. CCM* is an extension to CCM that besides supporting authentication with possible encryption, like CCM, it also supports encryption only. ZigBee encryption and authentication takes place on the Network Layer instead of, as for Bluetooth, on the Link Layer. Both ZigBee and Bluetooth architectures diverge from the well-known OSI-model however and it is therefore not possible to precisely detail what the equivalent OSI layer is. Similar to Bluetooth the underlying block cipher is AES-128 which, for the chosen cipher at least, gives the same level of safety.

ZigBee has capabilities to distribute keys used for encryption and signing of data transmissions. Depending on the chosen method for key distribution the connection can be more or less secure. Again, like Bluetooth, ZigBee has the

option to distribute keys OoB making its transfers as secure as the chosen OoB method permits [10].

## 2.6 Z-Wave

Z-Wave is a wireless communications protocol specification operating at 868 MHz (Europe) and 908 MHz (US). To avoid interference CSMA/CA is used. The used modulation technique is Binary Frequency Shift Keying (BFSK). Allowed data rates are 9.6 kbps and 40 kbps.

Devices can be of two types; controllers and slaves. A Z-Wave network is typically a mesh network with up to 4 hops, containing one controller and several slaves. The controller keeps a routing table for the nodes in the network. Slaves may act as routers but generally do not if they are battery powered. The protocol has support for automatic re-routing if a node should disappear from the network [11], [12].

### 2.6.1 Security

Unlike Bluetooth and ZigBee, Z-Wave is a proprietary technology. This makes it harder to analyze the security any further but one report outlines a method to exploit a vulnerability in the implementation of the Z-Wave protocol in a commercially available door lock [11]. Implementation vulnerabilities can affect any technology though and should be a concern with all the other evaluated technologies as well.

## 2.7 DASH7

The DASH7 Alliance protocol (D7A) is an evolving standard for wireless communication in the 433 MHz band. It is based on the IS0/EIC 18000-7 standard and uses CSMA/CA for channel access and BFSK for modulation.

Four device types are defined in the D7A standard; blinker, endpoint, gateway and subcontroller. A blinker is the simplest type of device that can only transmit data and does not use a receiver. Endpoint devices can both receive and transmit data. They are typically battery operated low power devices that periodically wake up to send and receive data. Gateways are the only type of device that has a receiver that always is on. They have support for all D7A features and are generally used to connect networks. A subcontroller also supports all D7A features, but is not always awake. It wakes up periodically to send and receive data [13].

Thanks to the low frequency used, D7A devices can reach a range of up to 1 km. The data rate is limited to 28 kbps because of the low bandwidth in the 433 MHz band [14].

### 2.7.1 Security

The DASH7 Protocol Specification does not mention anything more about security other than that security features will be implemented in future revisions of the specification [13].

## 2.8 Proprietary

There exist several proprietary RF technologies from vendors like Texas Instruments, Nordic Semiconductor, Microchip and others. To do a full survey of all the different RF technologies from the various vendors is not feasible within the scope of this thesis project, therefore "proprietary" is taken to mean any technology that is developed in-house at Axis Communications.

# Chapter 3
# Evaluation

To gain a better understanding of which technologies that should be incorporated in the prototype, an effort was made to characterize the different frequency bands that are available. Both with respect to legal issues and with respect to properties such as range and penetration capability. In this chapter these issues will be investigated.

## 3.1 Range

The two most important parameters that affect range are frequency and transmitted power. Friis transmission formula (eq. 3.1) expresses the relation between received power ($P_R$), transmitted power ($P_T$), gain of the antennas ($G_T$ and $G_R$), wavelength ($\lambda$) and range ($r$) [15]. Eq. 3.1 assumes that the antennas are matched and have aligned polarizations.

$$P_R = \frac{P_T G_T G_R \lambda^2}{(4\pi r)^2} \tag{3.1}$$

Eq. 3.1 shows that longer wavelengths, i.e. lower frequencies, results in increased received power and thus longer range. In fact, a twice as high frequency will result in approximately half the range. The equation also shows that a 6 dB increase in transmitted power ($P_T$) will result in double the range.

## 3.2 Frequency bands

There are a number of license free frequency bands available that may be utilized for wireless sensor networks. The following are the ones believed to be most suitable and some of their most important properties.

### 3.2.1 433 MHz

Many license free radio transmitters in Europe operate in the 433 MHz band with use cases being remote light switches and remote keyless entry systems [16]. This leads to congestion in the narrow frequency band with possible interference as a result. It is not unusual for RF equipment in this band to periodically stop working, or stop working in certain places as a result of this interference [17].

Also, 433 MHz is not available under the same conditions worldwide. While approximately the same frequency band can be used over most of the world there are many different requirements on duty cycle and output power [18].

Even though the open air range is longer for 433 MHz than for both 868 MHz and 2.4 GHz, according to Friis formula (Equation 3.1), it is harder to construct a good antenna since $\frac{\lambda}{4}$ at 433 MHz is longer, resulting in a larger antenna. In addition the possible data rate is lower than for 868 MHz.

### 3.2.2 868 MHz

ETSI standards, which are followed by most European countries, regulate the use of the 868.0–870.0 MHz frequency band. The frequency band is further divided into blocks with different requirements. Common for all frequency blocks are requirements on duty cycle, listen-before-talk, maximum radiated power and channel spacing [19].

Similar requirements are in place for the 915 MHz band used in the US. A license free frequency band in the region 868-930 MHz exists in most countries [20].

#### LTE interference

LTE uses three blocks in the 800 MHz band, each with a bandwidth of 10 MHz (Table 3.1). The 800 MHz band is mainly used for the data uplink and mostly in rural areas. Due to strong Out-Of-Band (OOB) emissions from the LTE User Equipment (LTE UE), LTE handsets may cause interference in the 800 MHz band.

**Table 3.1:** LTE frequency blocks in the 800 MHz band

| Block | Frequency range |
|---|---|
| A | 832–842 MHz |
| B | 842–852 MHz |
| C | 852–862 MHz |

According to [21] the interference caused by nearby LTE User Equipment (UE) might affect the performance of Short-Range Devices (SRD) that operates in the 868 MHz band.

Furthermore, [21] states that there is a risk of interference when LTE UE is used within a range of up to several meters of a SRD. The interference may result in a reduction of range of the SRD, or even a total loss of function. It is also stated that the risk of interference is highest for SRDs operating close to the 863 MHz border.

ERA Technology has presented a study on the effect of LTE UE interference on a social alarm system operating at 869.215 MHz. The report shows that the alarm system is affected by interference from the LTE UE under worst-case conditions, i.e. with the LTE UE placed 2 m from the social alarm unit and the LTE UE operating at maximum allowed output power (23 dBm).

Under more realistic conditions, i.e. the LTE UE operating 5 dB and 10 dB below the maximum allowed output power, the social alarm was well functioning and not affected by LTE UE interference [22].

#### 868 MHz range test

Range and penetration in the 868 MHz frequency band were tested using two Anaren AIR Module CC110L BoosterPacks. The test was conducted in an indoors office environment using output power ranging from 0 dBm to +10 dBm, using 4.8 kbps or 38 kbps bitrate.

Using +10 dBm output power the range was approximated to 30 m, ranging over ±1 floors. 0 dBm output power resulted in a range of approximately 20 m. The range was slightly (approximately 10 m) better using 4.8 kbps as a result of increased receiver sensitivity at lower data rates. Penetration was deemed to be better than for the equivalent test at 2.4 GHz.

### 3.2.3 2.4 GHz

The 2.4 GHz band is a globally available ISM band that is heavily utilized for Wi-Fi, Bluetooth, ZigBee and 3-G cellular data, among others. On top of this, microwave ovens often radiate around 2.4 GHz, further restricting the usable bandwidth.

The available bandwidth is approximately 100 MHz which does provide some room for coexistence and several of the technologies mentioned uses some sort of multiple access or spread spectrum scheme to reduce interference and/or increase robustness. Furthermore, a $\frac{\lambda}{4}$ antenna for 2.4 GHz is only about 3 cm allowing for a small and effective antenna implementation.

#### 2.4 GHz range test

A simple 2.4 GHz range test was conducted using two nRF24L01P radio modules controlled by Arduino microcontrollers. The modules were equipped with printed PCB antennas and configured for an output of 0 dBm (1 mW).

The test was conducted in an indoors office environment, without free line-of-sight between the radio modules and an outside environment with free line-of-sight between the radio modules. The maximum range was estimated to 10 m indoors and 100 m outdoors. It was noted that the penetration of walls was a lot worse for the 2.4 GHz radio than for the 868 MHz radio.

## 3.3 Evaluation results

Evaluation and tests concluded that 868 MHz was the best choice of frequency band. It provides good indoor range with acceptable data bitrate and antenna size.

The 868 MHz frequency band restrictions on duty cycle and transmitted power minimize the risk of interference from other transmitters. Similar license free frequency bands exists in most countries and many radio chips can operate in

these bands with only software or minor hardware modifications. LTE interference may be a future problem and should be further investigated. It should however be noticed that similar interference problems also exist in other license free frequency bands.

Range could be extended by utilizing mesh networking that some standards support. This is, however, power consuming and requires some nodes (e.g. routers) to be mains powered. Mesh networks may also result in significant delay when transmitting a data packet through many nodes, which is highly undesirable in an alarm application.

It was decided that ALOHA should be used and that CSMA was not needed. This was motivated by the fact that the each sensor transmits at a low duty cycle, which implies that the network will be lightly loaded and the risk of collision is low.

The probability of a successful transmission when using ALOHA can be calculated using Equation 2.1. A reasonable approximation is 50 sensors that transmits 120 bits and receives 120 bits every 60 s at 38.4 kbps. Under these conditions the number of transmissions per second is

$$\frac{50 \text{ transmissions}}{60 \text{ s}} = 0.83 \text{ transmissions/s}$$

The time needed to transmit one frame of 120 bits and receive one frame of 120 bits is

$$\frac{240 \text{ bits}}{38400 \text{ bps}} = 6.25 \text{ ms}$$

Transmission attempts per second, $G$, becomes

$$G = 0.83 \cdot 0.00625 = 0.0052$$

Using equation 2.1 the probability for a successful transmission can be calculated:

$$p = e^{-(2 \cdot 0.0052)} = 98.9\%$$

For a plot of how the collision probability varies with the number of sensors, see Figure 3.1. A plot using a large amount of sensors can been seen in Figure 3.2, where the exponential behaviour of the collision probability is clearly visible.

The above listed facts led to the decision not to use an existing standard and instead develop a simple proprietary protocol. The protocol should only contain the absolute necessary functions and could hence be very lightweight with focus on low power. Security should be implemented, either in software or hardware, using AES-128 based CBC-MAC.

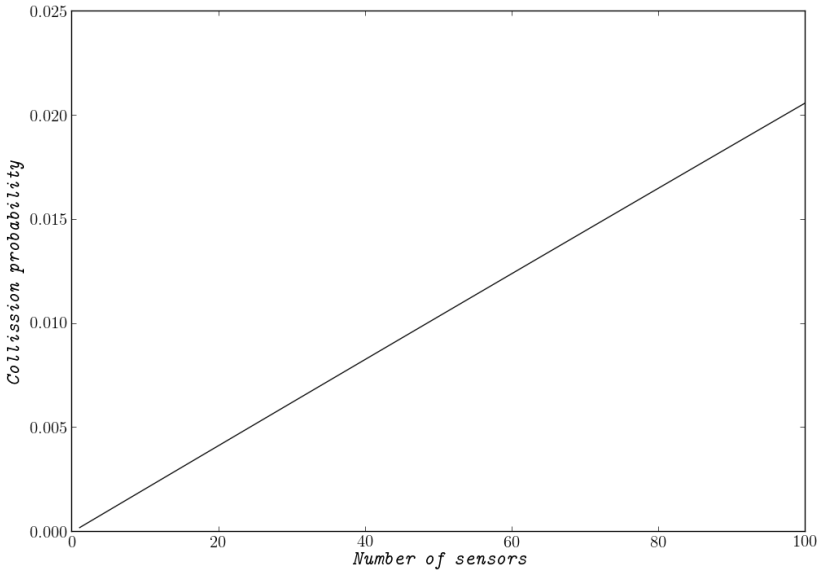


**Figure 3.1:** Probability of collision during a communication event as a function of the number of sensors, 0–100 sensors.

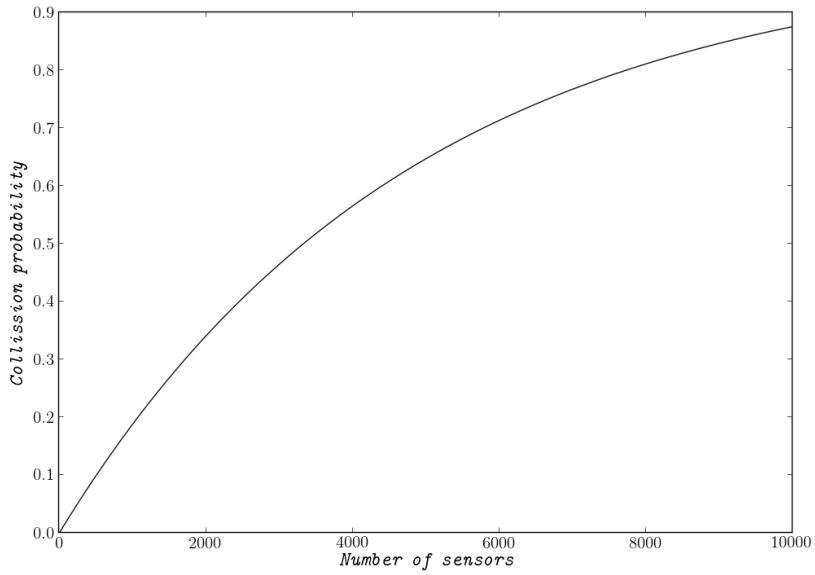


**Figure 3.2:** Probability of collision during a communication event as a function of the number of sensors, 0–10000 sensors.

Chapter 4

# Prototype design

The prototype design process was initially focused on getting an overview of how the system would be put together. The A1001 access controller uses RS-485 to connect to peripheral devices. Depending on the choice of radio chip, the accompanying microcontroller must have a compatible communication interface. A simple overview of the system can be seen in Figure 4.1.

The system consist of a radio hub that communicates with the A1001 using RS-485. The radio hub is powered from the A1001 and has hence no hard power requirements. It consists of a software development kit together with a radio module.

For the battery powered sensors a prototype using a designed PCB was constructed.

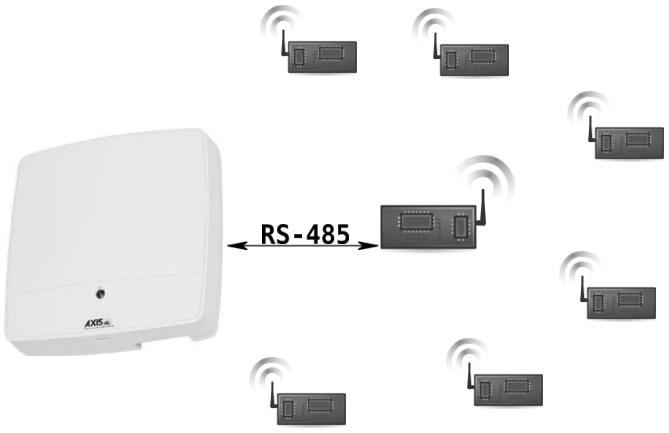


**Figure 4.1:** Overview of the complete system with A1001 access controller on the far left. The A1001 is communicating with a microcontroller which in turn handles the communication with all the sensors via radio.

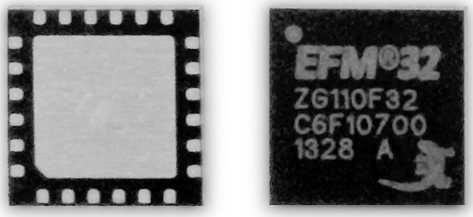


**Figure 4.2:** Silicon Labs EFM32 Zero Gecko, bottom (left) and top (right).

## 4.1 Hardware design

The sensor's hardware consists of only a few components, all mounted on a PCB. Apart from microcontroller (MCU), radio chip, sensor and battery there are only some small passives such as decoupling capacitors. No enclosure for the prototype was designed or manufactured.

### 4.1.1 Microcontroller

The most important requirement for the MCU is the current consumption in sleep mode. Since the processor will spend most of its time in sleep mode, with a real time counter (RTC) measuring the duration between two heartbeats, the sleep current must be kept to a minimum.

Other important requirements are:

- Support for hardware AES encryption (unless the chosen radio chip can do hardware AES)
- Communication interface to the radio chip, depending on which chip is used
- Reasonable cost

The chosen MCU (Silicon Labs EFM32 Zero Gecko) is a 32-bit ARM Cortex M0+, designed specifically for ultra-low power applications. The EFM32 Zero Gecko comes in different versions where the main differences are the amount of flash, RAM and GPIO pins. The chosen version, EFM32ZG110F32 (Figure 4.2), has 32 kB flash, 4 kB RAM, USART interface with SPI and I2C support along with support for hardware accelerated AES-128 which can be used to implement CBC-MAC. It's data sheet states a current consumption of 500 nA in sleep mode with 1 kHz RTC enabled [23]. The EFM32ZG also have indirect support for RS-485 since it supports USART. The signal levels of RS-485 and its differential nature does however mandate some sort of line driver. Here, a TI SN65HVD50 full duplex driver/receiver was chosen for no other reason than that it fulfills the specifications.

### 4.1.2 Comparison of 868 MHz transceiver chips

A number of suitable radio chips operating in the 868 MHz frequency band were compared and their most important performance parameters summarized into table 4.1.

**Table 4.1:** Comparison of 868 MHz transceiver chips

| **Chip** | **Sleep current (nA)** | **TX current at +10 dBm (mA)** | **RX current (mA)** | **RX sensitivity at ~40 kbps (dBm)** | **AES** | **MCU** |
|---|---|---|---|---|---|---|
| TI CC110L [24] | 200 | 30 | 17.7 | -104 | no | - |
| ST Spirit1 [25] | 600 | 21 (41) | 9.7 (17.6) | -106 (-109) | yes | - |
| Freescale MKW01Z128 [26] | 200 | 30 | 17 | -104 | yes | - |
| Atmel AT86RF212B [27] | 200 | 26.5 | 9.2 | -100 (at 20 kbps) | yes | - |
| Freescale MC12311 [28] | 1200 | 33 | 16 | -105 | yes | HCS08 |
| Silicon Labs Si106x [29] | 760 | 22.1 | 17.8 | -110 | no | CIP-51 |
| TI CC430 [30] | 1000 | 33 | 16 | -102 | yes | MSP430 |


**Figure 4.3:** Anaren radio modules.

### 4.1.3 Radio chip

To simplify the prototype design it was decided to not include design of an antenna. Design and matching of a PCB antenna is a difficult procedure and may require many iterations before a proper matching is achieved.

This consideration led to the choice of using a prefabricated radio module with built in PCB antenna. A suitable choice was the Anaren A110LR09 radio module (Figure 4.3) which is built upon the TI CC110L radio chip [31].

The TI CC110L has good electrical characteristics (Table 4.1) and supports a frequency range of 779–928 MHz which makes it usable in most countries. It has hardware packet handling support that adds preamble bytes, sync word and CRC checksum to the packet. Preamble/sync word detection and CRC check is taken care of by hardware in receive mode. Communication with the radio chip is takes place over a 4-wire SPI interface.

The module comes in two versions; with built in PCB antenna and with U.FL connector for use with an external antenna. The two versions share the same footprint and may therefore be used on the same prototype PCB without any modifications. This way, different antennas can be tested and their performance evaluated.

### 4.1.4 Power supply

The prototype is powered by a single CR2032 lithium battery (see Figure 4.4). The CR2032 has a rated voltage of 3 V, capacity of 250 mAh and self-discharge of less than 3.5% per year [32]. No voltage regulator is needed as both the MCU and the radio module will run on supply voltages from 3.6 V down to 1.85 V. Its small dimensions, high capacity and long lifetime makes it an ideal choice for this type of application, but unfortunately only on paper.

The reason that using a single, or even multiple, CR2032 batteries might not be the best alternative for supplying this kind of wireless sensor is that its capacity lowers significantly with increasing load. According to [33] and [34] the usable capacity in CR2032 batteries might be much lower than 250 mAh when sourcing more than 0.5 mA even if only intermittently. According to their measurements the capacity might drop to as little as 150 mAh if pulsing 0.5 mA from the battery.


**Figure 4.4:** CR2032 lithium battery.

This is partly due to the internal resistance of the battery, causing a large enough voltage drop to disable the processor and/or the peripherals. An AAA or AAAA battery with a boost converter or two such batteries in series might be a better idea. Two CR2032 batteries in parallel could also be used, providing about 300 mAh even in worst case. For the purpose of making measurements, a single CR2032 will be fine though.

### 4.1.5 Passive components

Decoupling capacitors had to be added to avoid noise on the power supply rail. A reference design which provided recommended capacitor values was used [35]. The radio module has built in decoupling capacitors, but it was decided to add two additional decoupling capacitors for the radio module's power supply just to be on the safe side.

A simple debounce circuit consisting of a resistor and a capacitor (a low-pass filter) was connected to the RESETn pin. This circuit prevents unstable behavior when resetting the MCU.

The prototype has an inexpensive and simple sensor that can sense various physical quantities. When sensing a particular quantity, the sensor output is either grounded or floating depending on value. The sensor pin that can be either floating or ground is connected to one of the micro controller pins. But since the MCU pin needs to have a well-defined value, a pull-up resistor needs to be used. The problem with using a resistor to pull the pin to a known state is that even with a value as high as 40 kΩ there will be a significant current draw when the sensor pin is grounded, since by then there will be a path from 3 V via the pull-up, to ground. This results in a power draw of 75 µA when the sensor pin is grounded, or about 100 times more than the stand-by current of the MCU and radio combined. The solution to this problem is to disable the pull-up in software and then continuously poll the pin until the switch is opened again.

To be able to program and debug the MCU a standard 6 pin 2.54 mm pin header was added to the prototype.

### 4.1.6 PCB layout

For schematic capture and PCB layout the DipTrace EDA package was used (see appendix). Special care was taken to lay out a proper ground plane since the size and shape of the ground plane can affect antenna performance.

When working with high speed digital signals, it can be shown that the ground return path will be located directly underneath the signal trace, given of course

that there is an unbroken conductive path in place. If there is no short in the return path underneath the signal trace the return current will need to take another route back to ground.

The near and far H-fields at a distance $D$ created by a closed current loop are given by

$$H_{near} = I \cdot \frac{A}{4\pi D^3}$$

$$H_{far} = I \cdot \frac{\pi A}{\lambda^2 D}$$

where $I$ is the loop current, $\lambda$ is the current wavelength and $A$ is the loop area. It is clear that a larger loop surface will increase the magnetic field strength and thereby the ElectroMagnetic Interference (EMI).

A longer return to ground will also lead to higher inductance which could lead to floating circuit elements which in turn can pose a problem when well defined signal levels are needed. An unbroken ground plane is the solution to both of these problems since the loop areas and the inductances are kept to a minimum [36].

When using a $\frac{\lambda}{4}$ antenna a good ground plane is very important. This is because the $\frac{\lambda}{4}$ antenna can be approximated as a $\frac{\lambda}{2}$ antenna given the reflection of the antenna in the ground plane. Ideally, the ground plane should be perfectly conducting and of infinite size. A copper ground plane that extends "enough" beyond the antenna has to serve as an approximation of this ideal [15].

Another important aspect of PCB design when dealing with digital circuit is that decoupling capacitors need to be placed close to the supply pins of the corresponding chip. Since digital circuitry needs relatively high currents in short bursts, a high impedance trace between the decoupling capacitor and the chip pin can lead to large voltage drops that in turn can affect the performance of the circuit.

When designing the PCB all of the mentioned aspects were taken into consideration while still trying to keep the final PCB small. Since there are only two active chips and some passive components, no more than two layers were needed, even considering the recommendations above.

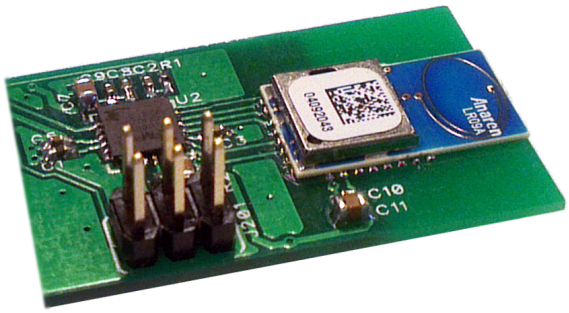


**Figure 4.5:** Sensor hardware, radio module and MCU.

## 4.2 Software design

The software for the system is split in three parts. The first part is a server that will run on the A1001. As the A1001 runs Linux the server is a Linux user space application written in C. While being an embedded system the A1001 still has plenty of resources for the purpose of this application, therefore no particular care has to be taken to minimize resource usage of the application.

The second part is a client that manages radio connections with all the sensors and communicates incoming information to the server. This piece of software runs on a bare metal ARM Cortex M0+ microcontroller and is also written in C. Due to the resource constraints on the microcontroller, care has been taken to minimize the amount of RAM and Flash used. Since this microcontroller is connected to the A1001 via cables, power is also supplied from the A1001. Since the power consumption of the microcontroller and radio is very small in comparison with the A1001, there are no particular power constraints on the hub.

The third part is the software running on the sensor itself. Here there are very high demands on both code size and power consumption, with power consumption being the more important constraint. To minimize energy usage it is vital that the microcontroller spends as much time as possible in a low energy mode and that all energy saving peripherals that can help lower energy consumption are used.

### 4.2.1 Writing software for low power applications

When developing for ultra-low power applications things that might not matter normally can have a large impact on energy performance. Take for instance the code in Listing 4.1, one might think that the generated assembly for the two functions will be the same. Looking at the code in Listing 4.2 and in Listing 4.3 reveals however that there is quite a bit of overhead in dealing with 8 bit integers on this 32 bit architecture. Even though the code in question would not be used in production, since it doesn't do anything, it illustrates that the programmer needs to be aware of how the coding affects energy performance. Counting the required clock cycles for the two functions in Listing 4.2 it is found that the int8_t function requires 13 instructions to complete while the int function requires 7 instructions, e.g. almost half as many, to complete. In a system that spends most of its time in a low power sleep mode and that only wakes up regularly to call some small functions, the choice between int8_t and int can make a big difference.

### 4.2.2 Protocol design

A packet oriented protocol has been chosen where each communication event is initiated by the sensor. In this way the sensor can be asleep whenever there is nothing to report. On the receiving side there are no hard power requirements and hence the radio can be always on, listening for packets. Since the sensor will be very simple and the data from the sensor is not assumed to be secret as long as its origin can be verified, no encryption is needed. With a bit of intelligence in the sensor, what needs to be sent is basically state changes and low battery indication, this means that each message can be short and as such consumes only small amounts of energy.

**Listing 4.1:** Simple C program to demo the difference between int8_t and int on a 32 bit architecture.

```
int8_t
myfunc1(int8_t i)
{
    return i;
}

int
myfunc2(int i)
{
    return i;
}


int main(void)
{
  while(1) {
      int8_t foo1 = 5;
      int foo2 = 5;
      int8_t bar1 = myfunc1(foo1);
      int bar2 = myfunc2(foo2);
  }

  return 0;

}
```

**Listing 4.2:** Resulting assembly when compiling the C code in Listing 4.1 using GCC with optimization O0.

```
000001d8 <myfunc1>:
 1d8:   b082            sub     sp, #8
 1da:   1c02            adds    r2, r0, #0
 1dc:   466b            mov     r3, sp
 1de:   3307            adds    r3, #7
 1e0:   701a            strb    r2, [r3, #0]
 1e2:   466b            mov     r3, sp
 1e4:   3307            adds    r3, #7
 1e6:   781b            ldrb    r3, [r3, #0]
 1e8:   b25b            sxtb    r3, r3
 1ea:   1c18            adds    r0, r3, #0
 1ec:   b002            add     sp, #8
 1ee:   4770            bx      lr

000001f0 <myfunc2>:
 1f0:   b082            sub     sp, #8
 1f2:   9001            str     r0, [sp, #4]
 1f4:   9b01            ldr     r3, [sp, #4]
 1f6:   1c18            adds    r0, r3, #0
 1f8:   b002            add     sp, #8
 1fa:   4770            bx      lr
```

**Listing 4.3:** Resulting main function when compiling the C code in Listing 4.2 using GCC with optimization O0.

```
000001fc <main>:


int main(void)
{
 1fc:   b510            push    {r4, lr}
 1fe:   b084            sub     sp, #16
  while(1) {
      int8_t foo1 = 5;
 200:   466b            mov     r3, sp
 202:   330f            adds    r3, #15
 204:   2205            movs    r2, #5
 206:   701a            strb    r2, [r3, #0]
      int foo2 = 5;
 208:   2305            movs    r3, #5
 20a:   9302            str     r3, [sp, #8]
      int8_t bar1 = myfunc1(foo1);
 20c:   466c            mov     r4, sp
 20e:   3407            adds    r4, #7
 210:   466b            mov     r3, sp
 212:   330f            adds    r3, #15
 214:   781b            ldrb    r3, [r3, #0]
 216:   b25b            sxtb    r3, r3
 218:   1c18            adds    r0, r3, #0
 21a:   f7ff ffdd       bl      1d8 <myfunc1>
 21e:   1c03            adds    r3, r0, #0
 220:   7023            strb    r3, [r4, #0]
      int bar2 = myfunc2(foo2);
 222:   9b02            ldr     r3, [sp, #8]
 224:   1c18            adds    r0, r3, #0
 226:   f7ff ffe3       bl      1f0 <myfunc2>
 22a:   1c03            adds    r3, r0, #0
 22c:   9300            str     r3, [sp, #0]
  }
 22e:   e7e7            b.n     200 <main+0x4>
```

For media access control ALOHA was chosen, partly because of the simplicity of the scheme but mostly because it works well when the medium is only lightly utilized. The random time hold off in case of collisions is seeded from the CBC-MAC, which, of cause, is not totally random but should be random enough.

For message verification a CBC-MAC based approach is chosen. That is the authentication method of Bluetooth Low energy as described in section 2.4.1. Although, where Bluetooth Low Energy uses CCM (Counter with CBC-MAC) for encryption and verification CBC-MAC will only provide authentication. To make it harder to perform replay attacks the message will be concatenated with a counter value that increments for each message. The message and the counter value will be encrypted using AES-128 and a pre-shared key after which an appropriate number of bits from the resulting cipher text will be appended to the message. The counter can be sent together with the message in plaintext since it will be extremely hard to guess the MAC even when the entire message is known.

32 bits from the CBC-MAC will be used but to make all bits count the 128 bit cipher text will be split in two 64 bit words which will be XORed with each other. The result will then be split in two 32 bit words and they will again be XORed. At the receiving end the MAC can be stripped from the message and the message can be encrypted using the same pre-shared key that the sensor node used. If the appended cipher text matches the calculated MAC the message is likely originating from the expected sensor.

A packet will consist of:

- 16 bit id
- 16 bit counter
- 8 bit command/state
- 32 bit MAC

resulting in a 72 bit message (Figure 4.6).

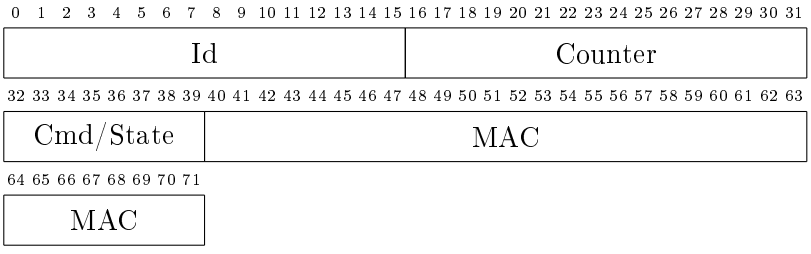


**Figure 4.6:** Packet description

# Chapter 5
# Verification

To verify the sensor's performance some key parameters were tested. Power consumption was measured using three data rates; 10 kbps, 38.4 kbps and 78.4 kbps using 0 dBm and +10 dBm output power. Indoor range, or rather Received Signal Strength Indication (RSSI), was measured at 10 kbps with an output power of 0 dBm and +10 dBm. The reason that only 10 kbps is used is that speed does not affect the signal strength but only the receiver sensitivity. RSSI is an estimated received power calculated by the receiving radio chip. It is not as precise as a measurement with a spectrum analyzer and a characterized antenna would be, but even then interference on different location due to bouncing signals could affect the measurements. Since the objective was to get a feel for the indoor range the RSSI was deemed to be good enough.

An outdoor line of sight range measurement was also made. For this measurement the sensors where separated until packages started dropping. The distance was then decreased until packages arrived reliably after which the actual distance was measured on Google Maps.

Measurement of the sensor's total power consumption was divided into current measurements of three different states; sleep mode, polling and RX/TX. Because of the great difference in current consumption between the three states different methods had to be used for each state.

The sleep mode current was measured using a Fluke 287 multimeter in series with the power supply. Power consumption during polling of the sensor state was estimated by measuring the time consumed for each poll and using the data sheet value for the active mode current consumption.

To measure the power consumption during RX and TX a 10 Ω resistor was connected in series with the power supply. The voltage drop over the resistor was measured using an oscilloscope, resulting in a measured voltage proportional to the current consumption. The waveform was captured in a .csv file and processed using SciPy.

Range was tested in two different indoor environments that get to represent the two extremes. Location 1 consists of rooms separated with both concrete and gypsum walls, with doors made of wood and glass (Figure 6.5). Location 2 consists of rooms separated with gypsum walls and glass doors (Figure 6.6). A sensor was placed at different locations, 1.5 m above ground, with the base station placed at a fixed location, 1 m above ground. The RSSI (in dBm) was measured at the base station for sensor locations 1-16 (Tables 6.3 - 6.4).

An alternative way to measure range could have been by measuring radiated power using a reference antenna, in a larger number of positions. The performed measurements are however believed to give a better idea of the sensors performance in practice.

# Chapter 6
# Results

## 6.1 Power consumption

The sleep mode current was measured to 700 nA using a multimeter. The current consumption during polling of the sensor was estimated as follows: According to the data sheet, the EFM32ZG110F32 takes 2 µs to wake from energy mode 3 (sleep mode) and draws about 1.6 mA at a clock frequency of 14 MHz [23]. Using an oscilloscope, the time for one poll was measured to 30 µs. Adding a 2 µs wakeup time and an asumed worst case estimate of 2 µs to go back to sleep, the approximate time for one poll is 34 µs. With a polling interval of 100 ms this results in an average current draw of

$$\frac{34\,\mu s}{100\,ms} \cdot 1.6\,mA \approx 544\,nA$$

which is about 140 times lower than the current draw when using the pull-up resistor as is.

The total stand-by current becomes

$$544\,nA + 700\,nA \approx 1.24\,\mu A.$$

Current consumption during RX/TX is shown in Figures 6.1 - 6.4. To lessen the impact of noise in the plots the values are filtered using a Savitzky Golay filter [37]. The Savitzky-Golay filter is a digital filter that smoothes the data by fitting low degree polynomials to subsequent sub-sets of the data. Here, a window size of 31 and a polynomial degree of 3 was used.

Using numpy.trapz a numeric integral was calculated for each of the filtered waveforms, giving the total Ampere-seconds of each communication event. Using a nominal voltage of 3 V, and thus ignoring the voltage drop over the resistor, an upper bound of the energy consumption for each communication event could be calculated, the results are presented in Table 6.1. Table 6.1 also shows expected battery life times under the assumption that the available battery capacity is 250 mAh and that there is one communication event every 60 s.

## 6.2 Range

Floor plans of the test locations with the different sensor positions are shown in

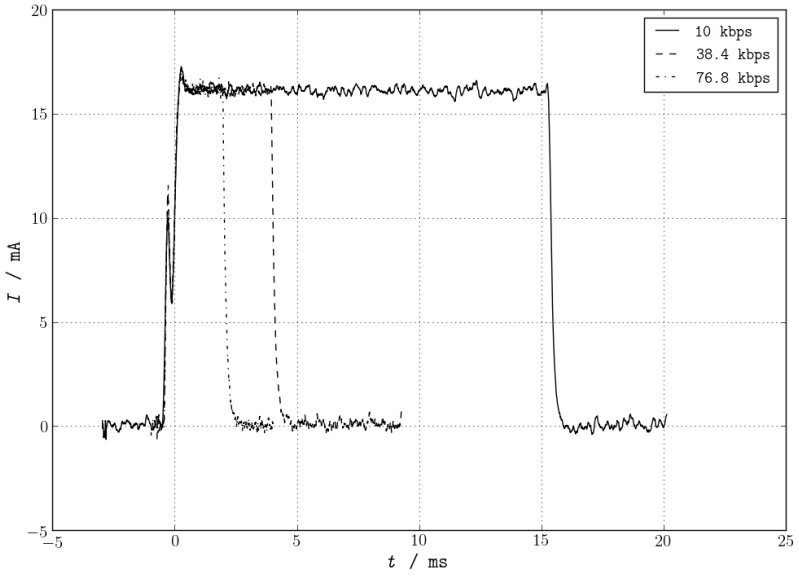


**Figure 6.1:** Current consumption when transmitting a packet at 0 dBm.

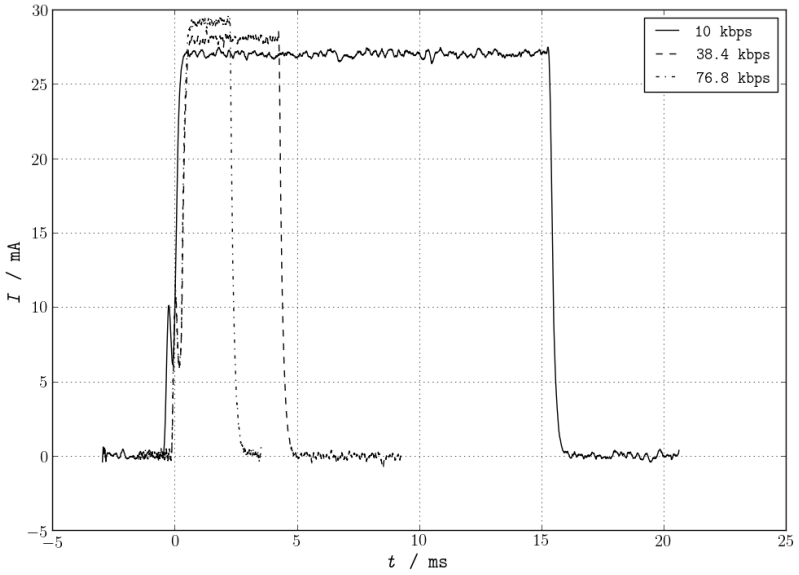


**Figure 6.2:** Current consumption when transmitting a packet at +10 dBm.

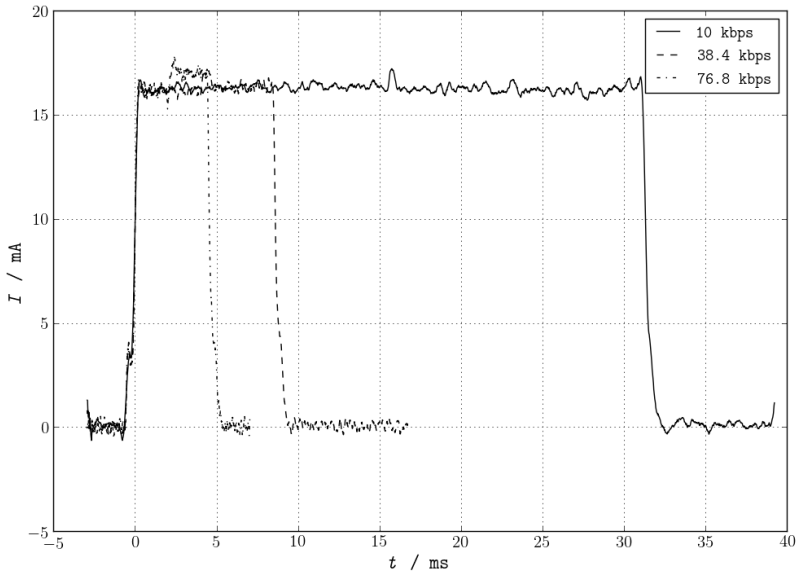


**Figure 6.3:** Current consumption when transmitting a packet and receiving an ack at 0 dBm.

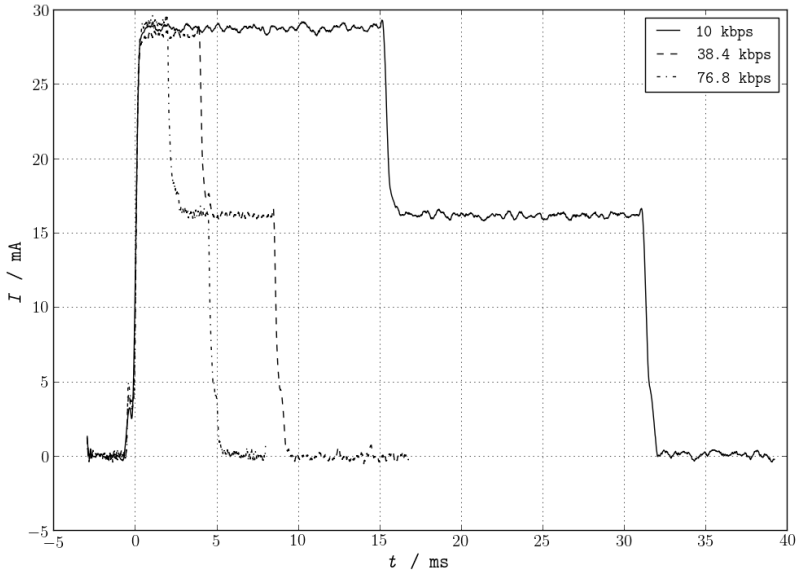


**Figure 6.4:** Current consumption when transmitting a packet and receiving an ack at +10 dBm.

**Table 6.1:** Energy consumption by speed for transmissions with ack as well as transmissions with no ack at 0 dBm and +10 dBm. All energy values are calculated using a nominal voltage of 3 V.

| Speed | Ack | Output power | Energy | Battery life† at 250 mAh |
|---|---|---|---|---|
| 10 kbps | ack | 0 dBm | $1.55 \cdot 10^{-3}$ W s | 2.90 years |
| 38.4 kbps | ack | 0 dBm | $436 \cdot 10^{-6}$ W s | 7.79 years |
| 76.8 kbps | ack | 0 dBm | $241 \cdot 10^{-6}$ W s | 11.1 years |
| 10 kbps | no ack | 0 dBm | $760 \cdot 10^{-6}$ W s | 5.22 years |
| 38.4 kbps | no ack | 0 dBm | $209 \cdot 10^{-6}$ W s | 11.9 years |
| 76.8 kbps | no ack | 0 dBm | $111 \cdot 10^{-6}$ W s | 15.4 years |
| 10 kbps | ack | +10 dBm | $2.12 \cdot 10^{-3}$ W s | 2.19 years |
| 38.4 kbps | ack | +10 dBm | $578 \cdot 10^{-6}$ W s | 6.41 years |
| 76.8 kbps | ack | +10 dBm | $317 \cdot 10^{-6}$ W s | 9.52 years |
| 10 kbps | no ack | +10 dBm | $1.26 \cdot 10^{-3}$ W s | 3.46 years |
| 38.4 kbps | no ack | +10 dBm | $350 \cdot 10^{-6}$ W s | 8.96 years |
| 76.8 kbps | no ack | +10 dBm | $190 \cdot 10^{-6}$ W s | 12.4 years |

†Assuming a standby current, including polling, of 1.24 µA and a period of 60 s.

Figures 6.5 - 6.6. The respective measured RSSI values are shown in Tables 6.3 - 6.4. The line of sight range is shown in Table 6.2.

**Table 6.2:** Line of sight range for 0 dBm and 10 dBm transmit power with A, integral antenna and B, quarter wavelength whip antenna at 10 kbps.

| Ant. | dBm | Range |
|---|---|---|
| A | 0 | 60 m |
| A | +10 | 160 m |
| B | 0 | 100 m |
| B | +10 | 200 m |

**Table 6.3:** Received Signal Strength by location for $0\,\mathrm{dBm}$ and $10\,\mathrm{dBm}$ transmit power with A, integral antenna and B, quarter wavelength whip antenna at $10\,\mathrm{kbps}$.

| Ant. | dBm | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 0 | -71 | -71 | -81 | -77 | -76 | -89 | -85 | -90 | † | -85 | † | † | -88 | † | † | † |
| A | +10 | -60 | -60 | -69 | -62 | -78 | -75 | -74 | -77 | -94 | -72 | -84 | † | -73 | † | † | † |
| B | 0 | -62 | -64 | -78 | -65 | -79 | -81 | -72 | -83 | † | -78 | † | † | -80 | † | † | † |
| B | +10 | -53 | -58 | -61 | -62 | -73 | -75 | -67 | -76 | -85 | -70 | † | † | -68 | † | † | † |

†Weak connection or no connection at all.

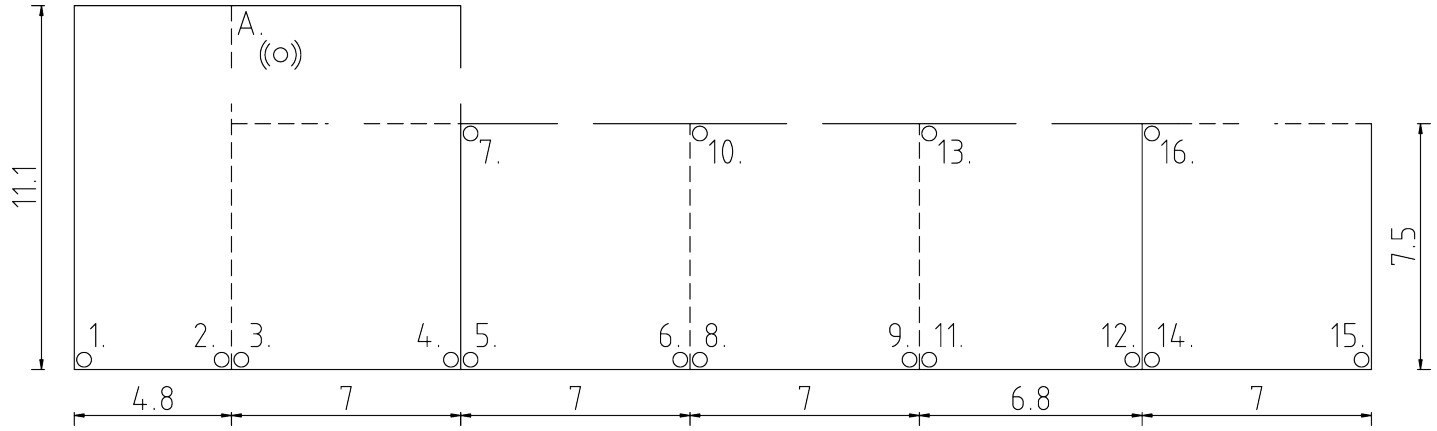


**Figure 6.5:** Floor plan of the "worst case" test environment, a basement in the LTH E-building. Dashed lines represent gypsum walls, solid lines are concrete. A is the hub, 1-16 are sensor positions. All measurements are in meters. Doors are wood and glass.

**Table 6.4:** Received Signal Strength by location for $0\,\mathrm{dBm}$ and $10\,\mathrm{dBm}$ transmit power with A, integral antenna and B, quarter wavelength whip antenna at $10\,\mathrm{kbps}$.

| Ant. | dBm | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 0 | -63 | -49 | -63 | -64 | -71 | -57 | -65 | -66 | -63 | -58 | -62 | -54 | -73 | -52 | -74 | -77 |
| A | +10 | -63 | -47 | -61 | -62 | -63 | -57 | -70 | -60 | -63 | -56 | -58 | -57 | -60 | -54 | -69 | -63 |
| B | 0 | -56 | -49 | -58 | -54 | -54 | -53 | -75 | -62 | -59 | -54 | -58 | -54 | -58 | -58 | -74 | -59 |
| B | +10 | -54 | -44 | -56 | -58 | -57 | -48 | -59 | -55 | -57 | -51 | -55 | -46 | -60 | -50 | -70 | -51 |

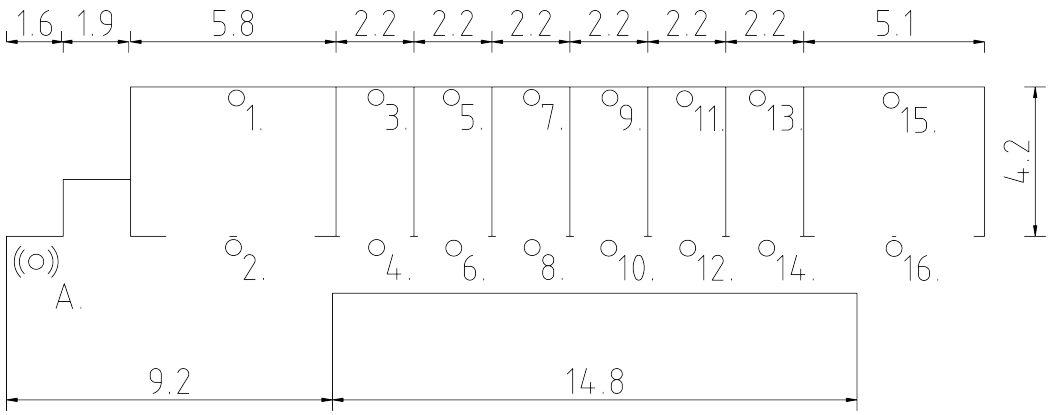


**Figure 6.6:** Floor plan of the "best case" test environment, inside an office building. All walls are made of gypsum. A is the hub, 1-16 are sensor positions. All measurements are in meters. Doors are glass only.

# Chapter 7
# Discussion

## 7.1 Power consumption

Using the selected MCU and radio chip there is no way to further decrease the power consumption in sleep mode, as it has reached the rated minimum according to the data sheets [23], [24]. There are however other choices of MCUs and radio chips (Table 4.1) that may result in lower power consumption in sleep mode.

One thing that came as somewhat of a surprise is the fact that polling of the sensor actually consumes less power than a fully interrupt driven solution. Interrupts are often touted as the best way to let the processor idle while waiting for work, and while that is true in a sense, as soon as there are other factors, like pull-ups, those needs to be taken into account. And after doing the math, it is obvious that polling is the way to go. The power consumed by polling the sensor could be reduced further, for example by polling at a lower rate. The polling rate is of course a parameter that is highly dependent on the sensor application. Further code optimizations could also reduce the power consumption caused by polling the sensor.

Power consumption in RX/TX mode is mostly a function of data rate and output power. Both these parameters affect the maximum achievable range and it is of course a trade-off between range and battery lifetime. Also in this case much can be won by choosing a radio chip (Table 4.1) that consumes less power in RX and TX mode.

A way to optimize battery lifetime could be to develop a system where the sensor's output power is automatically adjusted to achieve a reliable connection without wasting energy on unnecessary output power. This could be a part of the pairing process.

## 7.2 Range

Tests in both test environments show that the setup using +10 dBm and a whip antenna gave best results. Connection was generally stable without any packet loss down to -95 dBm, below that level the packet loss increased rapidly.

As expected, range is highly dependent on output power. The test also showed that an omnidirectional antenna, in our case a quarter-wave whip antenna, was a better choice than the modules integrated PCB antenna, probably since the

PCB antenna is more unidirectional and alignment of the antenna becomes more important. The maximum achievable range was generally much lower in the test environment with concrete walls, as a result of greater attenuation of the radio signal.

The line of sight range was about as good the 2.4 GHz range in the initial tests. This can be a bit surprising given that Friis formula (Equation 3.1) states that line of sight range should go up when the frequency is lowered. This is probably an effect of the fact that the antennas used are not optimally matched and that a proper matched antenna with good selectivity would provide a longer line of sight range. The indoor range is still better than for the initial tests of the 2.4 GHz radio. This confirms that the penetration of 868 MHz is better than for 2.4 GHz.

## 7.3 Open issues

There are several issues that that require a deep understanding of how the system will be used before decisions can be made about how to properly solve them. In this section some of these issues will be discussed and possible solutions to the underlying problems will be presented. Since the typical use cases of the final system are not fully determined no solutions in this section will be proposed as better or worse but rather they will serve as testimony that the problems can in fact be solved.

### 7.3.1 Usability

Usability of the system is one important factor when it comes to user perceived quality of the system. During work with this thesis we have identified one particular area where we believe usability to be of utmost importance. When the sensors are installed it should be easy to configure them with id and key so that they can communicate with the base station. Since the key can't be sent in plain text without risking that some potential attacker is eavesdropping there needs to be some way of transmitting the key securely. While this could be done using some kind of public key cryptography the limited resources of the embedded hardware raises the question if there is another way. There are several technologies that can be used for key transport; NFC and IrDA are two examples. A more novel method is used by the electric imp from Electric Imp, Inc. wherein network setup data is transmitted using an ordinary cell phone screen that affects an optical sensor which in turn is connected to the system in question.

While the electric imp solution is elegant in the sense that almost everyone has a cellphone that can be used to configure the device, it might not be a valid solution since it's patent-pending.

On the hub side the usability is more of a non-issue. While it is important there as well, the re-configurability of that part of the system makes it easy to adapt the user experience at any stage of the development process, even after the system has been deployed.

### 7.3.2 Security

The system has been implemented using AES-128 based CBC-MAC for authentication. To lower energy consumption only 32 of the 128 bits are used as verification of the message. This can be a security concern if the system is to be used in a high security application where the $> 4 \cdot 10^9$ combinations might not be enough. In such cases more of the bits from the 128 bit CBC-MAC can be used at the expense of battery time.

The decision was made that heartbeats should be ack:ed. This eliminates the possible problem of sensors not "calling in" for a while due to collisions. If the probability for a successful transmission (figure 3.1) is sufficiently high, heartbeats can be changed to not require an ack, which would reduce power consumption.

One feature that could be implemented when using ack:ed heartbeats is that clock drift in the sensors could be minimized by including a clock correction value with the ack. This would prevent sensors communicating with the hub from drifting into the same rhythm of sending heartbeats, something that could potentially cause more heartbeats to be lost.

By assuring that more heartbeats are received by the hub the implications of a lost heartbeat can be made more severe. If many heartbeats are lost when not using acks it would probably be a bad idea to raise an alarm for each lost heartbeat. It could turn into a "boy who cried wolf" sort of situation where alarms are not taken seriously.

Since AES-128 is already used for message authentication it would be very easy to encrypt the entire message if sensor data is in fact secret.

## 7.4 Summary

In summary the goal of this thesis project was met. Wireless sensors where developed and their performance was as good or better than the initial requirements. The battery life was calculated to at least five years using acknowledged messages, and at a bit-rate that permit above 20 m indoor range. Security features where added to make message spoofing extremely hard. Furthermore a network of five sensors was interfaced to the Axis A1001 access controller and the work done in this thesis has been incorporated in a proof of concept system that is in use as a demo, internally on Axis. While some parts, such as usability and security need more work, on the whole the result can be considered a success.

# References


[1] J. G. Proakis, *Digital communications.* McGraw-Hill series in electrical and computer engineering: Communications and signal processing, Boston: McGraw-Hill, 2001.

[2] B. A. Forouzan, *Data Communications and Networking.* New York: McGraw-Hill Higher Education, 2007.

[3] M. S. Gast, *802.11 Wireless Networks: The definitive guide.* Sebastopol, Calif.: O'Reilly, 2002.

[4] "IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)," *IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)*, pp. 1–314, 2011.

[5] C. Gomez, J. Oller and J. Paradells, "Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology.," *Sensors (Switzerland)*, vol. 12, no. 9, pp. 11734–11753, 2012.

[6] N. Hunn, *Essentials of Short-Range Wireless.* The Cambridge Wireless Essentials Series, Cambridge: Cambridge University Press, 2010.

[7] Bluetooth SIG, "Specification of the Bluetooth System," Version 4.1, Bluetooth Special Intrest Group, 2013.

[8] J. Jonsson, "On the Security of CTR + CBC-MAC NIST Modes of Operation - Additional CCM Documentation," 2002.

[9] E. Barker and A. Roginsky, "Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths," tech. rep., National Institute of Standards and Technology, 2011.

[10] ZigBee Standards Organization, "ZigBee Specification," Version r17, ZigBee Standards Organization, 2007.

[11] B. Fouladi and S. Ghanoun, "Security Evaluation of the Z-Wave Wireless Protocol."

[12] C. Gomez and J. Paradells, "Wireless Home Automation Networks: A Survey of Architectures and Technologies.," *IEEE Communications Magazine*, vol. 48, no. 6, pp. 92 – 101, 2010.

[13] DASH7 Alliance, "DASH7 Alliance Protocol Specification," DRAFT 0.2 Release, 2013.

[14] Bujdei, C. and Moraru, S. - A., "Wireless communications standards for intelligent buildings," *Annals of DAAAM and Proceedings*, pp. 489 – 490, 2010.

[15] J. D. Kraus and R. J. Marhefka, *Antennas For All Applications*. Boston: McGraw-Gill, 3 ed., 2003.

[16] Nexa Trading AB, "Nexa Electronics." Available at `http://www.nexa.se/index.html`. Accessed: 2014-01-31.

[17] C. Streiffert, "Radioträngsel kan blockera fjärrkontrollen." Available at `http://blogg.pts.se/blog/2012/10/29/radiotrangsel-kan-blockera-fjarrkontrollen/`. Accessed: 2010-09-30.

[18] RF Monolitics, Inc., "Survey of Radio Regulations for RFM Transceiver ICs," Application note - AN0900-04/10/09.

[19] ETSI, "EN 300 220-1, Electromagnetic compatibility and Radio spectrum Matters (ERM); Short Range Devies (SRD); Radio equipment to be used in the 25 MHz to 1000 MHz frequency range with power levels ranging up to 500 mW; Part 1: Technical characteristics and test methods," V. 2.4.1 (2012-05), 2012.

[20] GS1, "Regulatory status for using RFID in the EPC Gen 2 band (860 to 960 MHz) of the UHF spectrum," 2013.

[21] Electronic Communications Committee, "Adjacent band co-existence of SRDs in the band 863-870 MHz in light of the LTE usage below 862 MHz," ECC Report 207, 2013.

[22] Ofcom, "LTE User Equipment Coexistence with 862 - 870MHz," Research Document v1.1, 2012.

[23] Silicon Labs, "EFM32ZG110 Datasheet," Rev 0.61, 2013.

[24] Texas Instruments, "CC110L Value Line Transceiver," SWRS109A, 2014.

[25] STMicroelectronics, "SPIRIT1 Low data rate, low power sub-1GHZ transceiver," Rev 5, 2013.

[26] Freescale Semiconductor, "MKW01Z128 Highly-integrated, cost-effective single-package solution for sub-1 GHz applications," Rev 5, 2014.

[27] Atmel, "AT86RF212B Low Power, 700/800/900 MHz Transceiver for ZigBee, IEEE 802.15.4, 6LoWPAN, and ISM Applications," Rev 42002C, 2013.

[28] Freescale, "MC12311 Highly-integrated, cost-effective single-package solution for the sub-1 GHz, Wireless MBUS Standard," Rev 1.0, 2011.

[29] Silicon Labs, "Si106x/108x Ultra-Low Power MCU with Integrated High-Performance Sub-1 GHz Transceiver," Rev 1.0, 2014.

[30] Texas Instruments, "CC430 MSP430 SoC With RF Core," tech. rep., 2013.

[31] Anaren, "A110LR09x User's Manual," 2011.

[32] Jhih Hong Technology Co., LTD, "Specification of product CR2032," 2008.

[33] J. Ganssle, "How much energy can you really get from a coin cell?." Available at http://www.embedded.com/electronics-blogs/break-points/4429960/How-much-energy-can-you-really-get-from-a-coin-cell-. Accessed: 2014-04-30.

[34] K. Furuset and P. Hoffman, "High pulse drain impact on CR2032 coin cell battery capacity," tech. rep., Nordic Semiconductor and Energizer, 2011.

[35] Silicon Labs, "Hardware Design Considerations, AN0002 - Application Note," Rev 1.36, 2013.

[36] Atmel application note, "EMC Improvement Guidelines," Revision 4279B-8051-08/03, 2003.

[37] Various, "Savitzky Golay Filtering." Available at http://wiki.scipy.org/Cookbook/SavitzkyGolay. Accessed: 2014-05-06.

# Appendix A
# Schematic

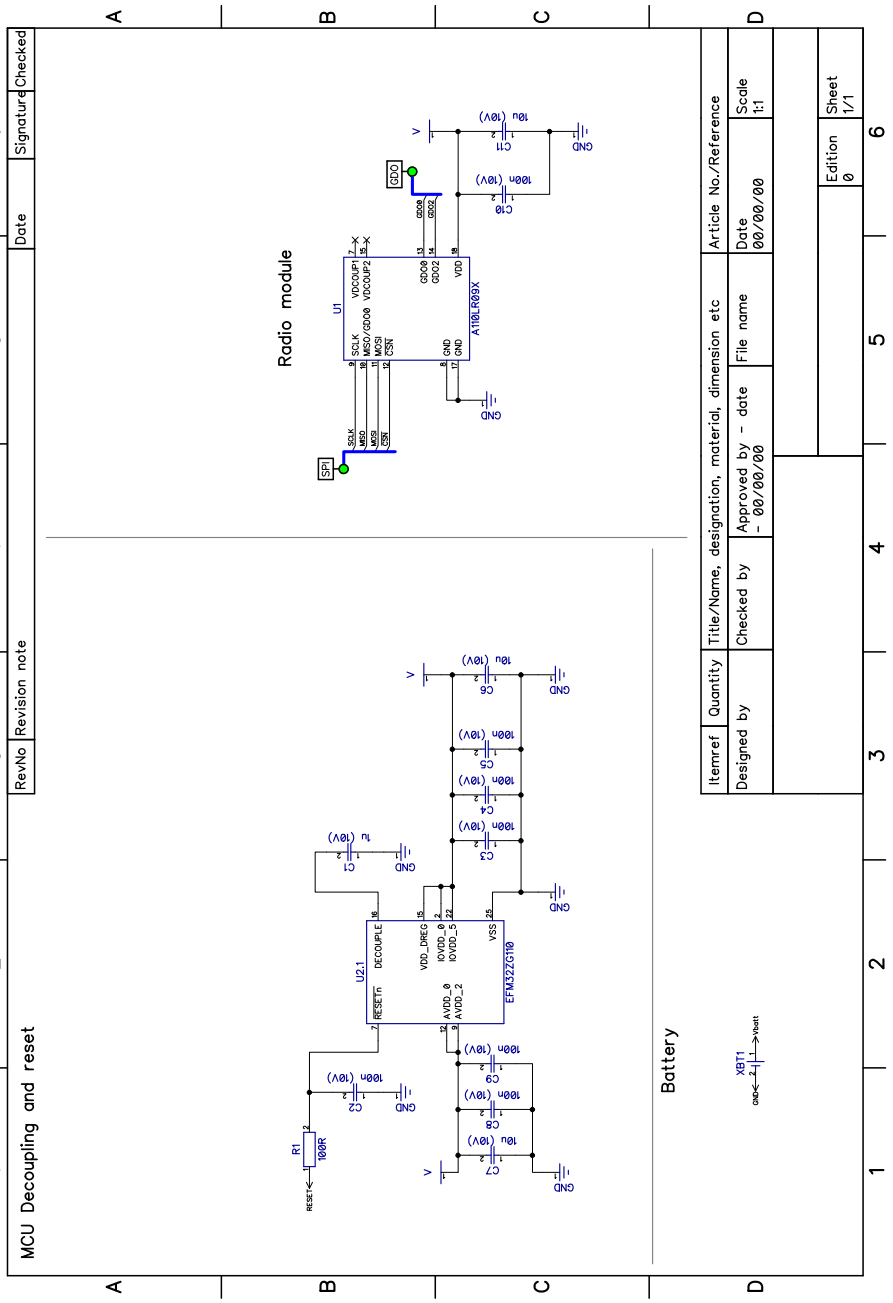


**Figure A.1:** Schematic sheet detailing decoupling and power and the radio module communication bus.

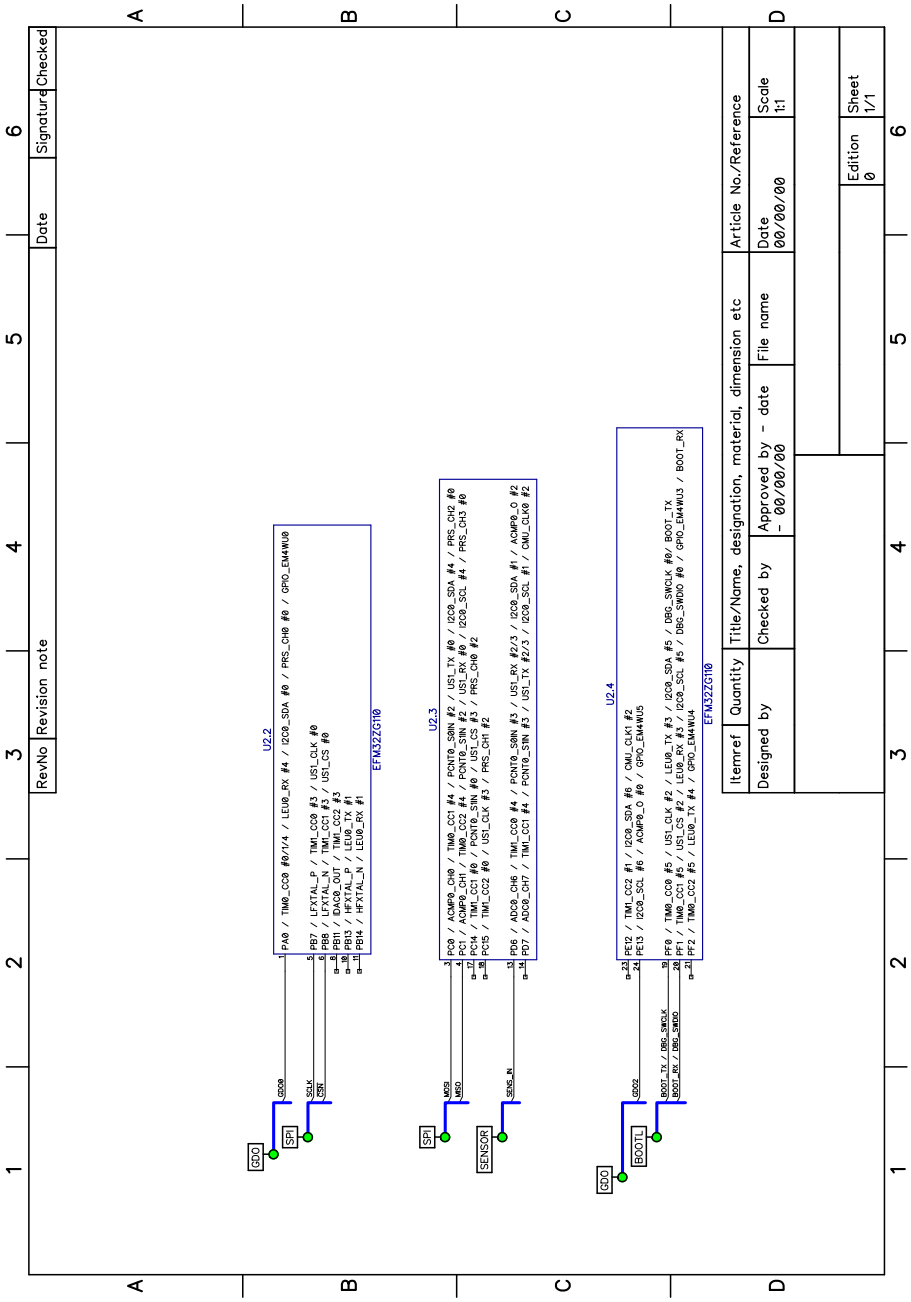


**Figure A.2:** Schematic sheet detailing EFM32ZG110F32 I/O.

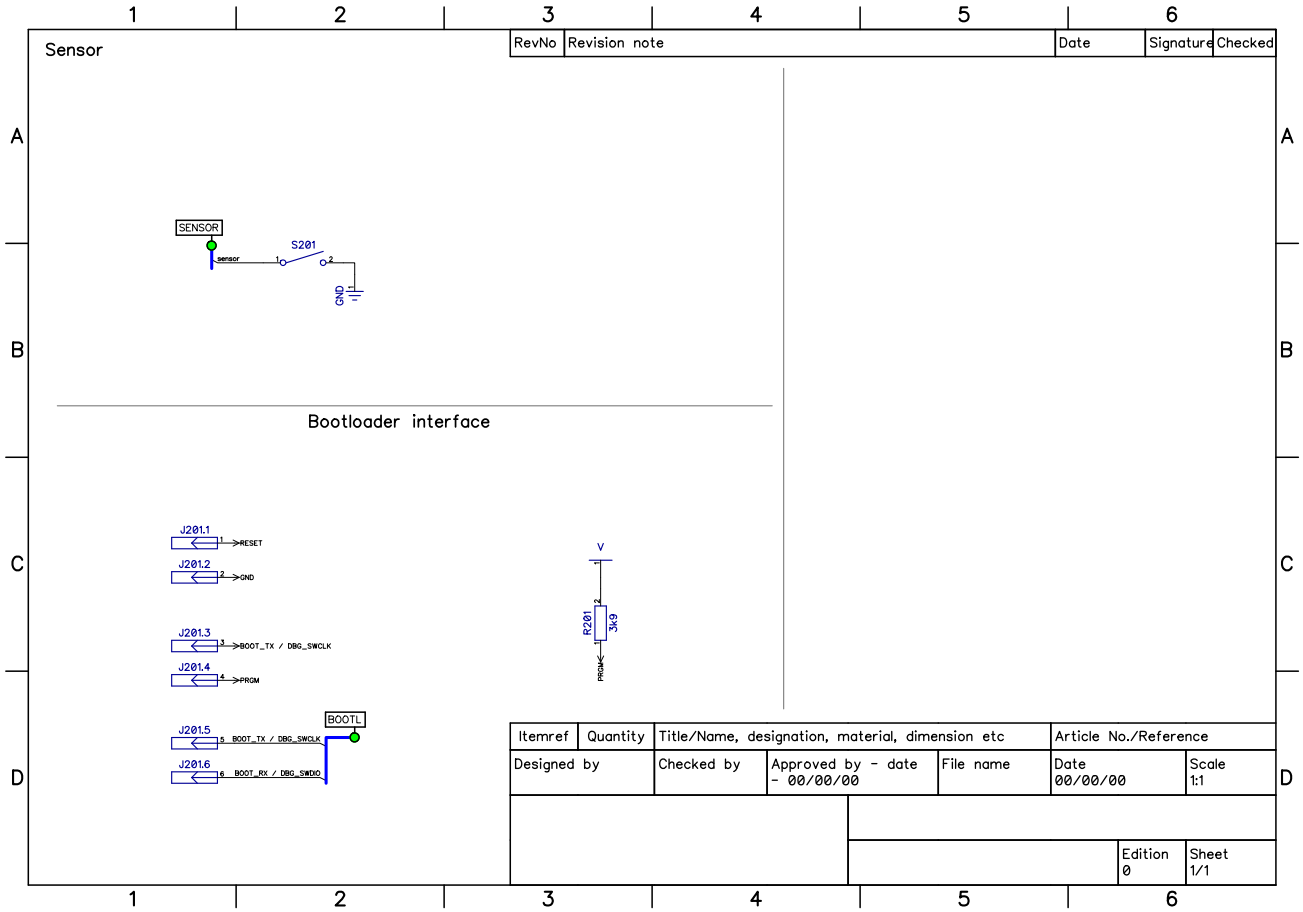


**Figure A.3:** Schematic sheet detailing switch and programming interface.

# Appendix B
# PCB layout

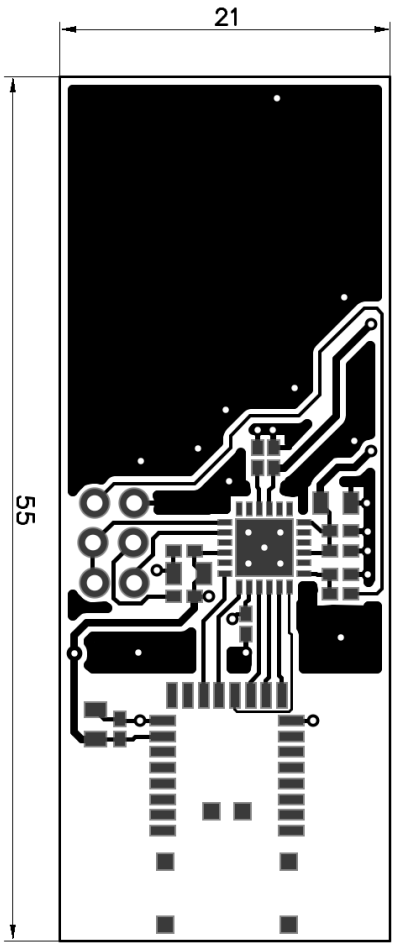


**Figure B.1:** PCB top side, dimensions in mm.

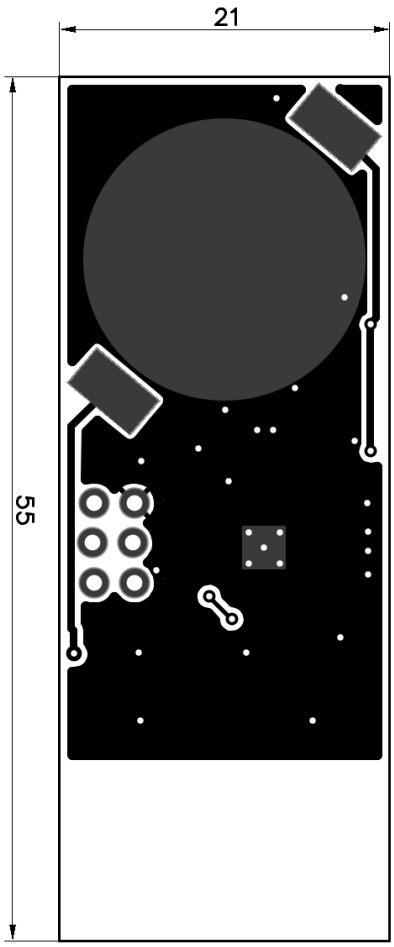


**Figure B.2:** PCB bottom side, dimensions in mm.

LUND
UNIVERSITY

http://www.eit.lth.se