Power Effect Control for IEEE 802.15.4 Based Wireless Devices

Elise Holmstedt
Abstract

Power Effect Control for IEEE 802.15.4 Based Wireless Devices

Elise Holmstedt

The goal of this project was to develop a method that can control the power level of IEEE 802.15.4-based wireless devices by analyzing the incoming signal and steering the strength of the transmitted signal in a way that diminishes the power requirements while keeping an adequate signal. Local wireless networks connecting simple devices e.g. sensors or power switches, use the standard IEEE 802.15.4 and within it comes a number of built-in methods for analyzing the quality of a signal. By studying their behaviour it is determined that the link quality indicator is best suited for these circumstances. Its statistical behaviour is studied for a range of power-levels to create a foundation for the simulations and to map the expected target areas.

A model is constructed in Matlab and a method for power-steering is built. The method analyzes the statistical parameters of the last 16 messages of the incoming signal. Based on where these are in relation to the map created earlier a message is included to the other device about whether to increase or decrease the signal strength. To create more stability as soon as the signal is within the target area the number of sampled messages is increased to 160.

This method is implemented on the IEEE 802.15.4-based wireless devices and the stability is observed. To explore the method further four different target areas are implemented for different requirements of the network and experiments are made with each.
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCA</td>
<td>Clear channel assessment</td>
</tr>
<tr>
<td>CSMA-CA</td>
<td>Carrier sense multiple access with collision avoidance</td>
</tr>
<tr>
<td>BER</td>
<td>Bit error rate</td>
</tr>
<tr>
<td>ED</td>
<td>Energy detection</td>
</tr>
<tr>
<td>EVM</td>
<td>Error-vector magnitude</td>
</tr>
<tr>
<td>FER</td>
<td>Frame error rate</td>
</tr>
<tr>
<td>FFD</td>
<td>Full-function device</td>
</tr>
<tr>
<td>FCS</td>
<td>Frame check sequence</td>
</tr>
<tr>
<td>GTS</td>
<td>Guarded time slot</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium access control</td>
</tr>
<tr>
<td>MFR</td>
<td>MAC footer</td>
</tr>
<tr>
<td>MHR</td>
<td>MAC header</td>
</tr>
<tr>
<td>MLME</td>
<td>MAC sublayer management entity</td>
</tr>
<tr>
<td>MPDU</td>
<td>MAC protocol data unit</td>
</tr>
<tr>
<td>LQI</td>
<td>Link quality indicator</td>
</tr>
<tr>
<td>O-QPSK</td>
<td>Offset quadrature phase shift keying</td>
</tr>
<tr>
<td>PAN</td>
<td>Personal area network</td>
</tr>
<tr>
<td>PHR</td>
<td>PHY header</td>
</tr>
<tr>
<td>PHY</td>
<td>Physical layer</td>
</tr>
<tr>
<td>PLME</td>
<td>Physical layer management entity</td>
</tr>
<tr>
<td>PPDU</td>
<td>PHY protocol data unit</td>
</tr>
<tr>
<td>PSDU</td>
<td>PHY service data unit</td>
</tr>
</tbody>
</table>
RF  Radio frequency
RFD  Reduced-function device
RSSI  Received signal strength indicator
SAP  Service access point
SFD  State of frame delimiter
SHR  Synchronization header
SNR  Signal to noise ratio
WPAN  Wireless personal area network
# Contents

1 Introduction  
   1.1 Background .................................. 1  
   1.2 Problem Description ......................... 2  
   1.3 Purpose .................................... 3  

2 IEEE 802.15.4-based Wireless Devices and Networks  
   2.1 Wireless Devices ............................ 6  
   2.2 Fundamentals of Wireless Communications  
       and Networks ................................ 7  
       2.2.1 Radio Wave Propagation ................. 7  
       2.2.2 Modulation ................................ 7  
       2.2.3 Multiple Access Techniques .......... 11  
   2.3 IEEE 802.15.4 and WPAN ..................... 12  
       2.3.1 Protocol Structure ..................... 12  
       2.3.2 Network Model ......................... 13  
       2.3.3 Frame Structure ....................... 15  
       2.3.4 PHY - Physical Layer .................. 18  
       2.3.5 MAC - Medium Access Control Layer ... 19  
       2.3.6 Performance Metrics ................. 20  
   2.4 ZigBee .................................. 23  

3 Methods for Power Control  
   3.1 Signal Quality Indicators ................... 25  
       3.1.1 RSSI ................................ 25
3.1.2 LQI ............................................. 26
3.1.3 Method chosen for Quality Control ................. 27
3.2 Power Control ........................................ 27

4 Experiment and Simulation .................. 29
4.1 Setup ............................................. 29
4.1.1 Packet Layout ................................ 29
4.2 Experimental Results .............................. 31
4.2.1 Analyzing the statistical behaviour of a sample .... 31
4.3 Simulations ....................................... 33
4.3.1 One-Way Communication ...................... 34
4.3.2 Two-Way Communication ...................... 37
4.4 Discussion ........................................ 39
4.4.1 Code for Transmission .......................... 41
4.4.2 Code for Reception ............................ 41

5 Results ........................................... 43

6 Conclusions and Future Work ................. 46
6.1 Conclusions ...................................... 46
6.2 Future Work .................................... 47
Chapter 1

Introduction

This report will first present the theory behind the devices and how the features relevant to this project function. Following this there is a presentation of the development of the method. This is done in three steps:

- Gathering data and evaluating devices behaviour.
- Building a model that simulates the devices and developing a method to regulate the signal.
- Implementing the method on the devices and testing.

1.1 Background

Wireless devices have been widely used in our daily lives and industries for performing simple tasks, e.g., a power switch or reporting data from a thermometer. Those devices communicate with each other over a radio based network using the IEEE 802.15.4 standard. For this network to preform in a way that is optimal for users there are a number of different things to take into consideration when adapting a network for a specific application. Many of these devices are battery driven. Thus, power efficiency is one of the most important issues.
To deal with the power efficiency of this network it is important to begin with looking at the communication between two single devices, the problems that can occur and what the quality of the signal is dependent on.

1.2 Problem Description

Generally, to reduce the power consumed while still keeping the desired quality of the signal, there needs to be an evaluation of what the acceptable quality is and what the cost of transmission is. The cost is defined as the sum of the power used and the loss of quality for that power level. In Fig 1.1 a graph of an expected cost for general systems is presented. As the power drops the losses get higher at different rates depending on the system and the environment. The challenge is to construct a method that finds the target area having the lowest cost for the specific circumstances of our setup. How expensive the losses are depends on the demand put on the system. In a system where the secure transmission of every single message is vital the cost would sky rocket for just minor disturbances. If the system has the possibility to resend the lost or corrupted packages the cost for the loss would be lower but in a time sensitive system the gain might be cancelled out by the lost time. The choice of packet length also influences the cost. Shorter packages have a higher probability to get through unaffected but since the header on each package is the same length more transmissions are needed for the same amount of information. The same goes for security coding. Harder security makes the packages longer for the same amount of information but it also gives a better chance of recovering corrupted data [1].

Also to take into consideration is the robustness of the algorithm. If the method can quickly follow the signal it can adapt itself after only a short time, however, it is also possible for the system to react to what are only statistical aberrations to the signal and be led astray, or sudden momentary disturbances that might be best to ignore for the performance sake. A signal strength that goes up and down a lot might also make the overall results
worse. On the other hand a method that is too slow to change might have difficulty in keeping up with quicker environmental changes, especially if there is a long period between transmitted messages.

Figure 1.1: The cost of a system against energy for different losses of quality.

The method operates by device one transmitting messages and device two receiving them as illustrated in Fig. 1.2. Device two then studies the incoming message and by comparing its quality to the preset desired values calculates how device one should optimize its transmission power according to the radio channel condition. The command to change is included in the next transmission from device two and once device one receives it the power is changed. This is done continuously by both devices so that the power regulation can follow changes in the environment.

### 1.3 Purpose

The purpose of this project is to develop a method that controls the transmission power for mobile tranceivers and optimizes it for lower power con-
sumption while still sustaining a reliable signal. This is achieved by studying how the signals transmitted by the devices behave, especially around the break point, where the signals become too weak to be received correctly by the receiver.

The main reasons for optimizing transmission power are for:

- Longer battery life: A lot of these devices are probably run on batteries and prolonging their lifetime is always important.

- Reducing interference: A wireless device in a network sending radio waves can interfere with other devices or even themselves due to reflected signals. They can also create noise pollution that interfere with other systems in the environment.

- Reducing the number of corrupted messages and assuring that the message reaches its receiver in a timely manner.

The challenge is to discover exactly how a signal behaves as the power changes in that specific environment and design an algorithm that directs
the power to the preferred levels according to the specific demands without lowering it too far and loosing an objectionable amount of data.
Chapter 2

IEEE 802.15.4-based Wireless Devices and Networks

A wireless device consists of a microcontroller, an antenna and a transceiver that is IEEE 802.15.4 compliant. To create a Wireless personal area network (WPAN), a number of wireless devices are connected in different configurations depending on the usage of the network.

2.1 Wireless Devices

Two wireless radio devices based on a Microchip MRF24J40MA tranceiver were used in this project. They communicate by using the the IEEE 802.15.4 standard protocol, a low-cost way to convey simple messages over a relatively short distance in wireless personal area networks (WPAN) where long battery life is important.

The basic driver runs on Microchip PICDEM$^\text{TM}$ demonstration board and the code for the Microchip MRF240J40 Tesing interface, written in C, was used as a base for the program. The board is connected to the Hyper Terminal that is running on a computer and can receive commands and deliver results to it [2].
2.2 Fundamentals of Wireless Communications and Networks

When transmitting a signal between two wireless devices there are many things to take into consideration. Knowing how a radio-wave behaves in different media and when encountering obstacles, a method has been developed for how a signal is modulated and transmitted. The necessity of many signals being transmitted by different devices in the same general area has led to different methods for multiple access being developed over time.

2.2.1 Radio Wave Propagation

When a wireless signal is transmitted it is affected by multiple elements in the environment that it passes through. Since the signal is an electromagnetic wave that is decoded by the receiver to discrete integers a small deviation in the wave is possible to compensate for but a bigger deviation can be interpreted as a completely different numerical value or be impossible to interpret at all. Waves being sent between two devices can pass through a number of different medium such as air, wood, glass and other building materials. They are also exposed to several sources of noise such as other electrical devices and background radiation. Different weather patterns also have effects on the signal. It is also a possibility that part of the signal can be reflected off a surface and is received with a certain time shift by the receiver. Therefore it is important to have a robust signal with many ways of checking the quality.

2.2.2 Modulation

A frame is put together in the physical layer (PHY) of the device and needs to be modulated before it is transmitted over the radio. This means that the binary data is transformed to a modulated wave signal suitable for the radio transmission. This is done in three steps, illustrated in Fig. 2.1.
Each frame is mapped byte for byte in order from the first byte in the preamble to the last in the Frame Check Sequence (FCS). Each byte contains eight bits numbered from least significant bit, \(b_0\), to the most significant, \(b_7\). The least significant bit is the rightmost in each byte. So if a frame is considered written the normal way, from left to right, it is read like that byte wise, but each byte is read backwards.

The first step during modulation is mapping the bits to symbols four by four. First the four least significant \((b_0, b_1, b_2, b_3)\), and then the most \((b_4, b_5, b_6, b_7)\). These bits can be considered a data symbol with an equivalent decimal value from 1 to 15 [3].

Symbol to Chip

The symbols are mapped to a 32-chip sequence (each byte corresponds to eight chips). These are nearly orthogonal sequences of pseudo random noise.

The reason for using nearly orthogonal codes is that it makes it easier for the receiver to interpret the symbol correctly. Orthogonal codes are a base that span a multidimensional space and a symbol transmitted without interference should respond to 1 on the axis representing that symbol. A mildly disturbed symbol will still respond mainly to the correct axis even if it has some lower values on the other. Therefore the receiver will map the symbol to each different code and the one closest to one is most probably the correct one [3].
The most common method of transmitting chips is to map them onto a sine wave that changes phases depending on if it’s a one or a zero it’s transmitting. This produces a wave form that ‘jumps’ whenever a different type of bit is transmitted, as seen in Fig. 2.2, and is called phase shift keying.

![Figure 2.2: The carrier wave changes phase depending on if it is transmitting a one or a zero.](image)

When mapping the chips onto a carrier wave the fact that the wave travels in three dimensional space is exploited. The space is described as a complex plane with an I and a Q axis perpendicular to each other. The transmitted wave function has an amplitude of one and a phase angle of

<table>
<thead>
<tr>
<th>Data symbol</th>
<th>Data symbol, binary</th>
<th>Chip value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0 0 0 0</td>
<td>1 1 0 1 1 0 0 1 1 1 1 0 0 0 0 1 1 0 1 0 1 0 0 1 0 0 1 1 0 1 1 0 1 0 0 1 1 0 1</td>
</tr>
<tr>
<td>1</td>
<td>1 0 0 0</td>
<td>1 1 1 0 1 1 0 1 1 1 0 0 1 1 1 1 0 0 0 1 1 1 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0</td>
</tr>
<tr>
<td>2</td>
<td>0 1 0 0</td>
<td>0 0 1 0 1 1 1 0 1 1 0 1 1 0 1 1 1 1 0 0 0 1 1 0 1 1 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0</td>
</tr>
<tr>
<td>3</td>
<td>1 1 0 0</td>
<td>0 0 1 0 0 0 1 0 1 1 1 0 1 1 0 1 1 1 0 1 0 0 1 1 1 1 1 0 0 1 1 1 0 0 0 1 1 0 1 0 0 1 0 0 1 0</td>
</tr>
<tr>
<td>4</td>
<td>0 0 1 0</td>
<td>0 1 0 1 0 0 1 0 0 1 0 1 1 1 1 1 1 0 0 1 1 1 0 1 0 0 1 1 0 0 1 1 0 0 1 0 0 1 0 0 1 0 0 1 0</td>
</tr>
<tr>
<td>5</td>
<td>1 0 1 0</td>
<td>0 0 1 1 0 1 0 1 0 0 0 1 0 1 1 0 1 0 1 1 1 1 1 1 0 1 1 0 0 1 1 0 0 1 1 0 1 1 0 1 1 0 1 1 0 0</td>
</tr>
<tr>
<td>6</td>
<td>0 1 1 0</td>
<td>1 1 0 0 0 1 1 1 1 0 1 1 0 0 0 1 0 1 0 0 1 1 0 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 0 1 1 0 0 1 1 0 1 1 0 0</td>
</tr>
<tr>
<td>7</td>
<td>1 1 1 0</td>
<td>1 0 0 1 1 1 1 1 0 0 0 1 1 1 1 0 1 0 1 1 0 0 1 0 0 1 1 1 1 1 1 0 1 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 0</td>
</tr>
<tr>
<td>8</td>
<td>0 0 0 1</td>
<td>1 0 0 0 1 0 0 1 0 0 1 0 1 1 0 0 0 0 1 1 1 1 0 1 1 1 1 1 0 1</td>
</tr>
<tr>
<td>9</td>
<td>1 0 0 1</td>
<td>1 0 1 1 1 0 0 0 1 1 1 0 1 0 1 0 0 1 1 1 0 1 0 0 0 0 0 1 1 1 0 1 1 1 0 0 1</td>
</tr>
<tr>
<td>10</td>
<td>0 1 0 1</td>
<td>0 1 1 1 1 0 1 1 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 1 0 0 0 0 0 1 1 1 1 1 1 1 0 1 1 1 0 0 1</td>
</tr>
<tr>
<td>11</td>
<td>1 1 0 1</td>
<td>1 1 1 0 1 1 1 1 1 0 1 1 0 0 0 1 1 1 0 0 1 0 1 1 0 1 1 0 0 0 0 0 1 1 1 0 1 1 1 0 1 1 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>12</td>
<td>0 0 1 1</td>
<td>0 0 0 0 0 1 1 1 1 1 1 0 1 1 0 1 1 0 0 1 1 0 0 1 1 0 1 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 0 1</td>
</tr>
<tr>
<td>13</td>
<td>1 1 1 1</td>
<td>0 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 0 0 0</td>
</tr>
<tr>
<td>14</td>
<td>0 1 1 1</td>
<td>1 0 0 1 0 1 1 0 0 0 0 0 1 1 1 1 1 1 1 1 0 1 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 0 0 1 1 1 0</td>
</tr>
<tr>
<td>15</td>
<td>1 1 1 1</td>
<td>1 1 0 0 1 0 0 1 0 1 0 1 1 0 0 0 0 0 1 1 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 0 0</td>
</tr>
</tbody>
</table>

O-QPSK
either $\pi/4, 3\pi/4, 5\pi/4$ or $7\pi/4$ depending on which two bits it is mapped against. This is illustrated in Fig. 2.3. Even indexed chips are mapped onto the I carrier and the odd to the Q carrier. This method is called quadrature phase shift keying.

![Figure 2.3: Chip placement on the complex plane.](image)

In offset quadrature phase shift keying (O-QPSK) there is a delay between when the I and Q waves change chip. The delay, $T_c$, is the inverse of the chip rate since by using two carriers each chip is transmitted for $2T_c$ and the switch takes place in half of that period. This is illustrated in Fig. 2.4. The main advantage of using this method is that each phase change is never bigger than $\pi/2$.

The chip rate $T_c$ used for the 2450 MHz band is set to 2.0 Mchip/s. This means that the symbol rate becomes 625.5 ksymbols/s. This is a fourth of the system preset data rate of 250 kb/s since the mapping has made each byte four times as long [3].
2.2.3 Multiple Access Techniques

There are different methods for multiple devices to communicate over the same medium, like in wireless communication, without disturbing each other and assuring that the right recipient gets its message. Multiple access can be realised based on the divisions of time, frequency, code and space. There are various multiple access technologies in common use:

**TDMA, Time Division Multiple Access:** In TDMA the medium is divided into time slots. Each device in the network is assigned a specific slot to transmit on and the network cycles through them at an even speed. This method is most effective if all devices are expected to communicate the same amount, otherwise a big portion of the time slots will be empty.

**FDMA, Frequency Division Multiple Access:** The frequency band is divided and each different device communicates on a slightly different frequency. It allows devices to talk any time but can be vulnerable to interference between the frequency bands. It also demands higher preforming hardware.

**CDMA, Code Division Multiple Access:** Each user is assigned a code from a near orthogonal code base. This code is used to modulate the signal to make it unique. This means that every device can transmit at once and each receiver can filter out the appropriate signal if it knows...
the right code. A disadvantage is that it creates multiple times more data to transmit.

**SDMA, Space Division Multiple Access:** In this method the base station keeps track of which direction and how far away each device is located and instead of just transmitting all over, the broadcast is targeted at the device. This method saves power but demands a lot from the hardware.

**CSMA-CA:** In an IEEE 802.15.4 wireless network, carrier sense multiple access with collision avoidance (CSMA-CA) is used for the multiple access. The basic idea behind the method is that a device listens to the channel to see if it’s in use at the moment. If it’s free it transmits its data and if it’s in use it waits a certain time and then tries again [3].

### 2.3 IEEE 802.15.4 and WPAN

A WPAN has a layered communications architecture from physical (PHY) layer, medium access control (MAC) layer, to network, transport and application layers. Each layer is responsible for its own things and offers services to the higher layers while performing them for the lower. The layer on the bottom is the physical layer responsible for the radio communication of a message transformed into bits. The layers above are more and more complex and perform more and more complex instructions until we reach the top layer that is the user interface or the external application that performs the intended function of the device. The lower levels (PHY and MAC layers) are of interest in this project.

#### 2.3.1 Protocol Structure

A WPAN uses the IEEE 802.15.4 protocols that specifies low level communications in the physical (PHY) layer and the medium access control (MAC) layer which are explored further in sections 2.3.4 and 2.3.5. These protocols
are the foundation for the ZigBee protocols (section 2.4) that include the network and security layers and the application framework. On top of this is the application that runs on the Microchip PICDEM\textsuperscript{TM} Z. An illustration of the stack is given in Fig. 2.5 [3, 4].

![Diagram of the protocol stack.](image)

2.3.2 Network Model

The IEEE 802.15.4 standard defines two different types of network devices. These are full-function devices (FFDs) and reduced-functioning devices (RFDs), respectively. RFDs are simpler, intended to be used for simple applications and need only to be able to communicate reliably with an FFD. RFDs do not have the full implementation of the IEEE 802.15.4 protocol and can sometimes be set to be only a transmitter or a receiver. They can also be off when idle and are thus preferred when a battery powered device is used. RFDs run simple tasks and programs and need only to be able to receive instructions and transfer results. The FFDs on the other hand are intended to be able to communicate with many devices at the same time. They function as
nodes and a coordinator of the network, they are responsible for forwarding messages to the correct target and interacting with the users through, for example, a computer. At the same time an FFD can function as an RFD. To build a WPAN there needs to be at least two devices and a minimum of one FFD.

The network can operate under two different topologies; star topology or peer-to-peer topology. Common for both is that they need one FFD with the role of PAN coordinator, this is the device that coordinates the traffic in the network. A PAN coordinator can also have its own application but its main use is to initiate, terminate and route communication. All devices in either network have a unique 64 bit extended address and each independent PAN will select an identifier. This identifier allows communication between devices and enables transmissions between different networks.

In a star topology all of the traffic goes through the PAN coordinator and directly to each other device in the network forming a star shape, hence the name. All star networks are independent from any other network in the vicinity. When a FFD is turned on for the first time it can establish its own network by becoming a PAN coordinator. A PAN identifier not currently associated to any neighbouring networks is chosen and the PAN coordinator can allow other devices to join the network, both FFDs and RFDs.

A peer-to-peer network has a more complicated structure allowing each node to communicate with every other node in reach. It is much more complex and can be self-organizing and self-healing and allows for multiple routes for a single message to take. RFDs can only, due to their physical limitations, communicate with a single device, but that device need not be the coordinator. Peer-to-peer networks also allow for the building of cluster trees, many networks communicating with each other, with one PAN coordinator but with a cluster head in each cluster, acting as a PAN coordinator. Examples of both networks are illustrated in Fig. 2.6 [3].
2.3.3 Frame Structure

There are four different types of messages available in IEEE 802.15.4. They are designed to be as simple as possible while still being robust enough to transmit over a noisy channel. The frame types are data frame, beacon frame, acknowledgment frame and MAC command frame.

Each frame is managed in the same way in the PHY layer and this is done as shown in Fig. 2.7. The PHY receives a packet from the MAC layer, a MPDU (MAC protocol data unit). This becomes the PHY service data unit (PSDU) in the transmitted message. The PSDU package is first prefixed with a synchronization header (SHR) containing information to help synchronize the package for the receiver. The first four bytes in the SHR is the preamble sequence, used to establish chip and symbol synchronization. Then comes one byte presenting the state of frame delimiter (SFD), used to indicate that the message is about to start. This is transmitted as the byte 1 1 1 0 0 1 0 1. The second header prefixing the PSDU is the PHY header (PHR) containing a byte specifying the number of bytes in the PSDU.
as an integer.

<table>
<thead>
<tr>
<th>Preamble Sequence</th>
<th>Start of Frame Delimiter</th>
<th>Frame Length</th>
<th>MPDU</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHR</td>
<td>PHR</td>
<td>PSDU</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2.7: Format of the PHY control data unit (PPDU) packet.

**Beacon Frame**

If the coordinator of the PAN has enabled the use of beacons they are transmitted over the network to identify the PAN and synchronize the attached devices. Beacons originate and are managed by the MAC sub-layer and transmitted from the coordinator of the network. The PAN can also support a super frame structure, in which 16 frames are bundled together and bounded by beacon frames. This means that every 16th frame is a beacon that synchronizes the network and describes the structure of the super frame.

The general layout of a beacon frame can be seen in Fig. 2.8 (a). In the payload of a beacon frame, first comes 16 bits of super frame specifications, different flags and numbers that describe the behaviour of the super frame. Following this there is the Guaranteed time slot (GTS) field that gives information of when and if there is a GTS. The pending address field contains information about the addresses of the devices that have a message pending in the coordinator. The beacon payload field is optional in the beacon frame and is mainly used for increased security [3].
Data Frame

The data frame contains, beside the header and footer, simply the data payload that the layer above the MAC has requested the MAC to transmit. The data frame structure can be seen in Fig. 2.8 (b) [3].

Acknowledgment Frame

The acknowledgment frame is used where a transmission needs to be verified that it has reached the recipient successfully. It contains simply a header and a footer and no additional payload. The sequence number shall reflect which message it’s responding to. The absence of an expected acknowledgment is taken as an unsuccessful transmission and the previous frame is resent. The structure is seen in Fig. 2.8 (c) [3].

MAC Command Frame

A MAC command is an instruction sent from one MAC sub-layer to another. The MSDU contains information on what command type is to be preformed, and the command payload, the command itself. A MAC command can demand that the responding device performs any of the services or features a MAC can do. MAC commands are used to associate/disassociate a device with a PAN, request data, a beacon, or a guaranteed time slot and notify status or request notification [3].

The different types of MPDUs shown in Fig. 2.8 have some parts in common. All MPDU frame types start with a MAC header (MHR). Thw first two bytes in the MHR represents frame control, which is the information on what type of frame it is, security, addressing fields and other flags. Then comes one byte representing the sequence number, a byte that is a unique sequence identifier for the frame. In beacon data and MAC command frames this is a counter that increases by one in each consecutive packet and is stored in each device’s MAC PIB while a acknowledgment frame copies the number from the frame it’s responding to. As shown in Fig. 2.8 all but the
acknowledgment frames have addressing fields. The addressing fields may vary in length and information depending on the setup of the network but it may contain information on the PAN identifier and the source address of both the recipient and the sender.

After the MHR comes the payload, the MAC service data units (MSDU), which differs for different types of frames.

The ending part of each frame is the MAC footer (MFT), which contains the frame check sequence (FSC). This is a security measure against corruption of the data during transmission and is calculated from the MHR and the MSDU with a cyclic redundancy check method. This works by recalculating the FSC in the receiver and comparing it with the received bits. If they are the same the packet is assumed to be correct, if they are not the packet is assumed to be corrupted and is discharged.

2.3.4 PHY - Physical Layer

The physical level (PHY) is situated at the very bottom of the architecture containing the radio frequency (RF) transceiver along with its control mechanism. PHY provides two services to the layer above. The first is the PHY data service that enables transmission and reception of PHY protocol data units (PPDUs) across the physical radio channel. This is the coding of the actual packet that is to be transmitted between different devices. The second service is PHY management, interfacing to the physical layer management entity (PLME). This entity is responsible for managing the level, performing instructions given to it from the above layer, and storing the managed objects in the PHY PAN information base (PIB). Each service is conducted through a service access point (SAP). This architecture is illustrated in Fig. 2.9.

The features that the PHY can provide are activation and deactivation of the radio transceiver, energy detection (ED), link quality indicator (LQI), channel selection, clear channel assessment (CCA), transmitting/receiving packets along the physical medium, and act as a interface between the MAC
sublayer and the physical radio channel. The radio operates on the license free band of 2400-2483.5 MHz worldwide [3].

2.3.5 MAC - Medium Access Control Layer

The medium access control (MAC) layer is the layer above PHY and handles all access to the physical radio channel. It is structured in the same way as the PHY layer, as shown in Fig. 2.10, with a management entity (MLME) containing a database with the objects managed by the MAC.

The MAC layer provides the PHY layer with the MPDUs to be transmitted and is the origin of the MAC command frames, that is presented in

---

**Figure 2.8:** The MPDU of the different types of frames. (a) Beacon Frame, (b) Data frame, (c) Acknowledgment Frame, (d) MAC command frame.
section 2.3.3. The MAC layer is also responsible for beacon management, channel access, guarded time slot (GTS) management, frame validation, acknowledged frame delivery and association/disassociation [3].

2.3.6 Performance Metrics

To evaluate a known signal a very useful method is to compare the message received with the one that was expected. There are two factors to take into consideration when doing this, the first is the number of packets that are discarded along the way or simply do not reach the receiver. The factor between the expected number of messages and the actual number of received messages is the frame error rate (FER).

The other factor is how many of the bits correspond with the expected message. This is the bit error rate (BER), the number of incorrect bits divided with the number of bits in the message.

In IEEE 802.15.4 there are a number of ways to measure performance. First there are a number of built-in methods in the PHY layer to determine
the quality of the channel they are transmitted on. There are also ways to check that the message that reaches the receiver is the same message that was sent and that there are no data corrupted or lost [3].

Error-Vector Magnitude

The error-vector magnitude (EVM) is used to determine the modulation accuracy of a transmitter. It is calculated by looking at a number of received complex chip values ($\tilde{I}_j, \tilde{Q}_j$). It is determined which value the chip has by looking at it in the complex plane and then comparing it to the ideal vector ($I_j, Q_j$). The difference ($\delta I_j, \delta Q_j$) is defined as the difference between the two (see Fig. 2.11). The EVM is then calculated by comparing the mean of the errors to the magnitude of the vectors and is presented as a percentage. The EVM has to be less than 35% [3].
Figure 2.11: Calculating the error between a received chip and the ideal.

Receiver Energy Detection (ED) and Clear Channel Assessment (CCA)

Receiver ED is used to estimate the signal power within the used bandwidth. The received value is reported to the MLME as an 8 bit integer between 0x00 and 0xff.

The PHY layer is also capable of doing a clear channel assessment (CCA). This is done in one of three ways. Either it runs an ED on the channel that it wants to transmit on and if it has a lower energy level than the threshold it is declared available. It can check if there is a signal on the channel with the characteristics of the system in use no matter what the energy level is or do a combination so it’s declared busy only if the signal is above the threshold [3].
LQI and RSSI

The received signal strength indicator (RSSI) and link quality indicator (LQI) are built-in signal quality indicators that are calculated by the PHY and added to the end of every incoming message. They will be discussed further in sections 3.1.1 and 3.1.2.

2.4 ZigBee

The wireless network devices use ZigBee, a network protocol specifically developed by the ZigBee Alliance for low data rate sensors and control networks. ZigBee offers low complexity and reduced resource requirements, best suitable for low cost battery-operated devices giving them a potential lifetime of several years. The goal when developing the protocol was to create a cheap, lightweight and easy to use way to connect everyday devices wirelessly.

ZigBee is a secure network technology that is layered on top of the IEEE 802.15.4 specifications for the MAC and PHY layers. The PHY layer was designed to allow for low cost, yet high levels of integration while the MAC layer allows multiple topologies, a large number of devices and the presence of RDFs. The ZigBee specifications for the network layer allow for the network to grow without requiring higher power levels and a large number of nodes connected to the network without creating large latencies.

ZigBee operates in three different unlicensed radio bands, 868 MHz, 915 MHz and 2.4 MHz for which there are a fixed number of channels. The bit rate depends on the operational frequency and can be up to 250 kbps. The low data rate in comparison to other wireless options is counteracted by the fact that for most devices using ZigBee transmission of big data files or streaming of audio files is simply not the goal. The typical message sent by a device using ZigBee is usually only a few bytes [4, 5, 6].
Chapter 3

Methods for Power Control

During wireless transmission there are many things affecting the signal as it is propagated between two different transceivers. The biggest influences on the signal are:

- The surrounding area: Is the system placed in an outdoor environment, in a office building or in a underground cave system? Are there any other sources of noise? Are they constant or do they vary greatly over time? Are there any reflective surfaces causing the devices to interfere with themselves?

- Placement of the devices: Are devices in direct line of sight or are there walls blocking the transmission? How far are there between different devices?

- How the devices are planned to be used: Are they supposed to be stationary or not? How often do they need to transmit? How important is it that every transmission reaches its receiver intact?

- The technology available: How good are the antennas? What kind of modulation is available? Which type of communication method is used? How long are the messages? Is resending messages a viable alternative?
Therefore it is important to be able to control the output power of the signal in a way that is suitable for the specific network. To be able to control the power it is important to first see what the quality of the signal is [3].

3.1 Signal Quality Indicators

The IEEE 802.15.4 standard comes with two different built in measurement parameters related to the quality of a received signal. These are RSSI and LQI. When receiving a package the devices automatically calculate these values based on the received signals and a simple program is used that sends a continuous stream of messages and prints out the values of these quality indicators.

Two nodes are placed approximately three metres apart and a signal with a gradually lowered output is transmitted. The behaviour of the quality indicators is observed. Especially interesting is what happens as the signal becomes progressively weaker and ultimately reaches its breaking point. The goal is to find a way to predict that the signal is about to become too weak to be useful without passing that limit [3].

3.1.1 RSSI

RSSI is a built in signal quality indicator for the receiver. The receiver measures the power of the received signal and displays it on a unit-less scale between 0 and 255, where 0 represents the lowest detected value. This scale is divided evenly into 32 different power levels [3].

While gradually lowering the transmitted signal power the value of the RSSI becomes so small that bit errors start to show up amongst the received packages. During the interval where the RSSI value deteriorates no measurable difference in the LQI value is observed (LQI will be discussed in the next section. 3.1.)

For each output effect a great number of messages are sent to achieve a more reliable statistical basis. For this test 1000 messages where sent for
each output power level.

![Figure 3.1: The received value of RSSI and LQI plotted against the descending transmission power level in dBm.](image)

### 3.1.2 LQI

LQI is a measure of the quality of the received package. It is calculated from the ED and the signal to noise ratio of the message [3].

In the same way that the RSSI value was evaluated, the LQI value is evaluated by lowering the signal transmission power even further until it is possible to detect a change in its behaviour.

The signal gets progressively worse as the LQI values drop, resulting in a signal that becomes more and more distorted, resulting in more bit errors and frame errors. The signal becomes too weak for detection well before the LQI value reaches zero. Fig. 3.2 shows the behaviour of LQI and how big the BER is for the corresponding transmission levels. BER is plotted on a log axis to better display it.

The fact that the signal is so weak around this point has made it harder to get more reliable data. Therefore for these tests between 5000 and 10000 messages have been sent at each transmission power level. Noticeable changes
in the signals behaviour have also been observed during different times of the day due to the natural interference of a normal work place environment. Steps to make the data more consistent have been taken, for example using more samples and running more tests.

### 3.1.3 Method chosen for Quality Control

After evaluating RSSI and LQI in this way it was decided that using the LQI was more beneficial in this study. It is also concluded that there needs to be more than one sample made to establish a reliable measure. Statistical evaluation needs to be done on a multitude of samples to accurately determine the quality of the signal.

### 3.2 Power Control

The transmitter has a built in function to change the power of the transmitted signal at will by the programmer. The simplest way to ensure a good quality signal is to simply set the output power set to ‘max’. But this will lead to large power consumption. Therefore, one strives to optimize transmission
power under the energy constraint discussed earlier in this thesis. To this end, there should be an optimization algorithm that takes into consideration the quality of signal using the metric LQI. In section 4.3 the method chosen for this work will be further discussed.
Chapter 4

Experiment and Simulation

The project has necessitated both an experimental phase and a simulation phase before the actual program controlling the power levels can be built. In the experimental phase the behaviour of the unadulterated signal is studied in the environment for which the program is built. The signals behaviour and all its built in quality metrics are also studied for a range of power that is possible for the transmitter. The results from these experiments are the basis for the simulation. In the simulation environment the basic algorithm for the final program is sketched out.

4.1 Setup

This test was performed in a standard office environment using two mobile devices with the ZigBee protocol installed. The devices were placed approximately two metres apart with no visible barrier, the test was performed during normal office hours with all the interference that this entails.

4.1.1 Packet Layout

A packet that is transmitted in this project from the PHY layer across the radio channel contains some meta data and the actual data being transmitted.
Transmitted Packet Layout

The packets are $9 + m$ bits in length and these bits are divided as shown in Table 4.1. The first two bits specify how a package is built, i.e., the header length and the package length. The next two bits define frame control, and the fifth bit defines sequence number, determining which message in a succession this is. Then the data to be transmitted follows. The footer of the message consists of frame check sequence (FCS) 1 and 2, coded bits added to improve the security of the transmission. After this there is an extra bit added for this project and this is used as communication to the other device regarding the need for power regulation. The table also shows which address each bit has in the memory.

<table>
<thead>
<tr>
<th>Address</th>
<th>Memory information</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x000</td>
<td>Header length</td>
</tr>
<tr>
<td>0x001</td>
<td>Packet length ($m + 5$)</td>
</tr>
<tr>
<td>0x002-0x003</td>
<td>Frame control</td>
</tr>
<tr>
<td>0x004</td>
<td>Sequence number</td>
</tr>
<tr>
<td>0x005</td>
<td>Data[0]</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>0x005+$m$</td>
<td>Data [$m - 1$]</td>
</tr>
<tr>
<td>0x005+$m$</td>
<td>FCS[0]</td>
</tr>
<tr>
<td>0x006+$m$</td>
<td>FCS[1]</td>
</tr>
<tr>
<td>0x007+$m$</td>
<td>Power control byte</td>
</tr>
</tbody>
</table>

Table 4.1: The addresses and contents of a transmitted message [3]

Received Packet Layout

The received package layout is similar to that of the transmitted layout, the difference is that since it is stored in a different part of the memory the addresses are different. It is also two bits longer, these are the quality indicators RSSI and LQI, that the PHY layer automatically performs on each received message and adds on to the end of it.
<table>
<thead>
<tr>
<th>Address</th>
<th>Memory information</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x300</td>
<td>Packet length ((m + 5))</td>
</tr>
<tr>
<td>0x301-0x302</td>
<td>Frame control</td>
</tr>
<tr>
<td>0x303</td>
<td>Sequence number</td>
</tr>
<tr>
<td>0x304</td>
<td>Data[0]</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>0x304+m−1</td>
<td>Data ([m − 1])</td>
</tr>
<tr>
<td>0x304+m</td>
<td>FSC[0]</td>
</tr>
<tr>
<td>0x305+m</td>
<td>FSC[1]</td>
</tr>
<tr>
<td>0x306+m</td>
<td>Power control byte</td>
</tr>
<tr>
<td>0x307+m</td>
<td>LQI</td>
</tr>
<tr>
<td>0x308+m</td>
<td>RSSI</td>
</tr>
</tbody>
</table>

Table 4.2: The addresses and contents of a received message [3]

### 4.2 Experimental Results

To be able to construct the required program, a program that transmits a continuous stream of packages is used. The program transmits the packages at the preset output power level. There is a delay between the packages to accommodate for two way communication where the devices need to switch to receiving mode to be able to receive a package before sending the next one.

#### 4.2.1 Analyzing the statistical behaviour of a sample

On analyzing the data further it is revealed that the distribution of the measured LQI value can vary much with the single power level. While the mean value appears to follow a clear curve indicating a consistent behaviour, as the signal gets weaker this, sometimes very wide distribution, makes it impossible for the observer to determine how strong the actual signal is from a single package. The behaviour must be analyzed further if this indicator is to be used in a reliable way.

Looking at the result from a single output level plotted on a graph conclusions of the statistical behaviour of the distribution can be drawn. Fig. 4.1
shows the results from a run with an output power of -22.5 dB, the frequency of the responses is plotted for LQI. The different statistical variables used for describing the distribution are calculated. For the normal distribution the characteristic variables are the mean $m = 105.4$ and the standard deviation $\sigma = 6.91$. The normal distribution curve with those characteristics comes rather close to the measured distribution, but is not that precise. One major difference between the measured distribution and the normal distribution is that the mean and the median are not the same, and the normal distribution is slightly skewed towards lower values.

The sample measured distribution is compared to the Gamma distribution. The characteristics for the Gamma distribution are the scale parameter $\theta$ and the shape parameter $k$. To get the numerical value for these properties the following relations

\[
\begin{align*}
    m &= k\theta \\
    \sigma^2 &= k\theta^2
\end{align*}
\]

are used. But since the measured distribution has not got a minimum of zero as a standard gamma function, but instead has a maximum of 121, the properties need to be adjusted for this. This means that the formula for calculating the characteristics becomes

\[
\begin{align*}
    k &= \frac{(121 - m)^2}{\sigma^2} \\
    \theta &= \frac{\sigma^2}{121 - m}
\end{align*}
\]

and the numerical values are gathered using the above formula. Plotting the Gamma function with those properties next to the measured distribution (Fig. 4.1), we can see a much closer match than when using the normal distribution.

This is done to all the relevant power levels. The different statistical properties are plotted together in Fig. 4.2 and their numerical values are shown in Table 4.3. These variables will be what are used to steer the final program for observing their behaviour around the breaking point.
4.3 Simulations

To understand how the signal behaves a model is built in Matlab that sim-
ulates the transmission. The data gathered from the test runs are used to
make this model.
Table 4.3: The statistical parameters against output power level

<table>
<thead>
<tr>
<th>dB</th>
<th>m</th>
<th>σ</th>
<th>θ</th>
<th>k</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>115.348</td>
<td>3.017</td>
<td>1.611</td>
<td>3.507</td>
</tr>
<tr>
<td>-1.25</td>
<td>115.471</td>
<td>2.829</td>
<td>1.448</td>
<td>3.817</td>
</tr>
<tr>
<td>-2.5</td>
<td>115.612</td>
<td>2.878</td>
<td>1.537</td>
<td>3.504</td>
</tr>
<tr>
<td>-3.75</td>
<td>115.75</td>
<td>2.922</td>
<td>1.626</td>
<td>3.227</td>
</tr>
<tr>
<td>-5</td>
<td>115.815</td>
<td>2.837</td>
<td>1.552</td>
<td>3.338</td>
</tr>
<tr>
<td>-6.25</td>
<td>115.844</td>
<td>2.806</td>
<td>1.527</td>
<td>3.375</td>
</tr>
<tr>
<td>-7.5</td>
<td>115.96</td>
<td>2.895</td>
<td>1.663</td>
<td>3.030</td>
</tr>
<tr>
<td>-8.75</td>
<td>115.948</td>
<td>3.258</td>
<td>2.101</td>
<td>2.403</td>
</tr>
<tr>
<td>-10</td>
<td>114.953</td>
<td>3.570</td>
<td>2.108</td>
<td>2.868</td>
</tr>
<tr>
<td>-11.25</td>
<td>115.142</td>
<td>3.265</td>
<td>1.819</td>
<td>3.218</td>
</tr>
<tr>
<td>-12.5</td>
<td>115.088</td>
<td>3.744</td>
<td>2.371</td>
<td>2.493</td>
</tr>
<tr>
<td>-13.75</td>
<td>115.09</td>
<td>3.745</td>
<td>2.373</td>
<td>2.489</td>
</tr>
<tr>
<td>-15</td>
<td>115.076</td>
<td>4.058</td>
<td>2.780</td>
<td>2.130</td>
</tr>
<tr>
<td>-16.25</td>
<td>114.875</td>
<td>4.798</td>
<td>3.758</td>
<td>1.629</td>
</tr>
<tr>
<td>-17.5</td>
<td>114.704</td>
<td>5.744</td>
<td>5.240</td>
<td>1.201</td>
</tr>
<tr>
<td>-18.75</td>
<td>114.495</td>
<td>6.242</td>
<td>5.990</td>
<td>1.085</td>
</tr>
<tr>
<td>-20</td>
<td>109.422</td>
<td>5.554</td>
<td>2.665</td>
<td>4.343</td>
</tr>
<tr>
<td>-21.25</td>
<td>107.978</td>
<td>5.856</td>
<td>2.633</td>
<td>4.944</td>
</tr>
<tr>
<td>-22.5</td>
<td>105.370</td>
<td>6.915</td>
<td>3.059</td>
<td>5.107</td>
</tr>
<tr>
<td>-23.75</td>
<td>103.905</td>
<td>9.529</td>
<td>5.312</td>
<td>3.217</td>
</tr>
<tr>
<td>-25</td>
<td>98.147</td>
<td>10.539</td>
<td>4.860</td>
<td>4.701</td>
</tr>
<tr>
<td>-26.25</td>
<td>91.266</td>
<td>12.068</td>
<td>4.898</td>
<td>6.070</td>
</tr>
<tr>
<td>-27.50</td>
<td>90.879</td>
<td>11.679</td>
<td>4.528</td>
<td>6.650</td>
</tr>
<tr>
<td>-28.75</td>
<td>90.538</td>
<td>11.258</td>
<td>4.160</td>
<td>7.320</td>
</tr>
<tr>
<td>-30</td>
<td>70.215</td>
<td>10.929</td>
<td>2.352</td>
<td>21.590</td>
</tr>
</tbody>
</table>

4.3.1 One-Way Communication

The first step is to write code that produces a result similar to that of the physical devices. By using the approximation of a gamma distribution a program is built that takes a certain power level as an input, looks up its corresponding statistical data and produces a random number within the appropriate interval. The power levels are, for the simplicity’s sake, represented by an integer between 0 (max) and 25 (min). To make sure that this is a valid approximation, simulations are done where the simulation result is
compared to the test data. In Fig. 4.3 a comparison between the measured results for an output power of -22.5 dB and its corresponding simulated value is shown. It can be observed that they are in a very good agreement.

Figure 4.3: A comparison between the measured values for an output of -22.5 dB and the results produced with a random number generator using the same statistical parameters.

An important decision to be made is how many data points are necessary to determine how good the received signal is. Since the LQI values can be widely distributed, especially for signals of a lower quality, as many data points as possible are desired. However, on the other hand the program needs to be relatively fast and collecting unnecessary data takes time. If the devices transmit bad signals, they need to be able to correct for that as soon as possible. There needs to be a balance so there can be a reasonable prediction with as few data points as possible. Therefore the second step is to build a program that determines how many data points are needed to calculate the signals stability.

This program works by generating random numbers with the same dis-
tribution as a signal, and cumulatively calculates the statistical parameters for the growing sample size. These are compared to the expected parameters for the distribution and a compilation of the errors, $\epsilon$, is studied. To gather reliable results a large number of simulations are run and the mean is calculated. This is done for various power levels and in Fig. 4.4 the plot for the power level 17 is shown.

![Graph showing error parameter $\epsilon$ depending on sample size for random numbers with the distribution equal to a signal with power level 17.](image)

By studying these graphs the number of values used to calculate reliable results is chosen. It is set to 16 since it is thereabout that the graph starts to pan out and more values will not make the results more accurate.
4.3.2 Two-Way Communication

To simulate two-way communication, feedbacks between the transmitter and receiver are necessary. The receiver calculates its measured data points and tells the transmitter how it wants the signal to change. For it to work effectively the receiver needs to be told what it wants. By looking at the LQI data results and the number of wrongly transmitted messages a goal area is picked, and the receiver is given instructions to see where in relation to this area the signal is. The spectrum is divided into three different types of areas; ‘red’, ‘yellow’ and ‘green’, where green is the target. If the signal is observed in the red area, the message to the transmitter is ‘take a big step’ and if the signal is in the yellow area the message is ‘take a small step’. If the signal is green it’s simply ‘hold steady’. The direction of these steps is based on if the signal is above or below the green area. All of this is represented in the code by integrals, the greater the number, the greater the step, with the sign determining the direction. A test case is shown in Fig. 4.5 where each ‘o’ shows the output power level and the limits from Table 4.4 are used. It takes about 5 to 8 changes of the power level to reach the target and the method is somewhat stable there.

<table>
<thead>
<tr>
<th></th>
<th>Red</th>
<th>Yellow</th>
<th>Green</th>
</tr>
</thead>
<tbody>
<tr>
<td>m</td>
<td>$&gt; 108$</td>
<td>$&gt; 108$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$&lt; 95$</td>
<td>$&gt; 95$</td>
<td></td>
</tr>
<tr>
<td>σ</td>
<td>$&gt; 9$</td>
<td>$&lt; 9$</td>
<td>$&gt; 1$</td>
</tr>
<tr>
<td></td>
<td>$&lt; 1$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4.4: The set limits for the test run.

When the method has reached the green zone it would be preferable if it became less sensitive to small disturbances. And as seen in Fig. 4.4, a bigger sample size gives a more accurate result. So once the rough calibration is made a more stable, i.e. longer, sample can be used. Therefore when the green zone is reached the long sampling starts, growing out to 160 received messages. The longer sampling also gets slightly wider boundaries and can compensate by itself if it reaches its yellow area. However, for the method to
still contain its sensitivity there need to be failure safes so that it can return to short sampling if there are major disturbances. Two of those conditions are if either the long sampling reaches its red zone or if the values of the 16 last messages reaches red. Fig 4.6 shows a trial run with these conditions implemented. The method becomes much more stable. The blue ‘o’ are for when the shorter sample size of 16 messages are used and the red ‘x’ are for the longer sample size of 160 messages.

It’s also interesting to study how well it handles disturbances. When two disturbances are added, the first one lowering the signal and the second one raising it, the method manages to correct itself quickly. All this can be seen in Fig. 4.7a which is a trial run with these conditions. If a disturbance that constantly raises the signal just a little is added the method behaves like in
Figure 4.6: A trial run with the matlab simulation with implemented longer time steps. The signal reaches it’s target and is stable once there.

Fig. 4.7b.

4.4 Discussion

The results from the simulations form the basis for the code implemented on the devices. The original code built for the experiments makes one device as a transmitter and which transmits a preset message of 13 bytes over and over again. The other device is made as the receiver which receives the message, checks its status and prints it. There are numerous changes that need to be made on the code to adapt it to the intended method. First of all it must be set up so that both devices are capable of both transmitting and receiving at the same time. This is technically impossible as the radio is only
(a) Two bigger disturbances.

(b) One smaller, constant disturbance.

Figure 4.7: The signal handles disturbances.
physically capable of doing one thing at a time. The work around is done by having the device switch back and forth between receiving and transmitting. Since transmitting a message takes a very short time a pause is included after each message is sent where the device switches to being a receiver. This gives an even spacing to the messages and can also conveniently be used as a counter and clock. It also makes the time a device can’t receive very small in proportion to the time it can.

4.4.1 Code for Transmission

The major change in the transmitted code is that an extra byte is attached to the message. This byte, \texttt{volume\_ch}, is there to inform the other device how to make corrections in its transmission frequency. This byte is calculated in the receiver. If no changes need to be made its value is set to 0x00, otherwise it’s set to how much higher or lower the other transmitter needs to change. Since this is an integer between 00 and FF it can’t have a negative value. Therefore the negative numbers are represented as very large, e.g. -0x08 is represented as 0xF8.

The program works in groups of 16. Before sending the first message in the group it checks if there is a request to change, if the incoming \texttt{volume\_ch} is not zero, the transmitter changes its frequency here. Then the program adds the outgoing \texttt{volume\_ch} to the message and sends the same message 16 times.

4.4.2 Code for Reception

It is in the code for reception that the major changes are done. After receiving each message the LQI value is stored and it’s statistical parameters $m, \sigma, k$ and $\theta$ are calculated every 16th message, both for the 16 and 160 most recent values. The variables are compared to those of the target area and if they are outside those the value of \texttt{volume\_ch} is adjusted accordingly.

A problem that arose was if one signal suddenly became too weak for its
data to be reliably interpreted by the receiver. If this happens the receiver has no way to know that it needs to boost its signal since that message simply does not reach it and it will just continue transmitting on the same power level or in some cases even turn it down completely, if that was the instruction in the last received message. To get around this problem a counter is inserted. Since the devices transmit messages in a continuous and even way this is used as a metronome of a sort. If the discrepancy between the number of received messages and the number of transmitted messages becomes big this is a sign that drastic measures are needed. When this occurs the transmitter boosts its signal significantly and the volume_ch is set to 0x80. As soon as the receiver gets this message that device’s transmission frequency is also increased.
Chapter 5

Results

Running the program it is possible to observe that the power of the transmitted signal behaves like it’s supposed to do. It quickly reaches a suitable area and stays there with only some minor fluctuations. To evaluate the stability and security of the transmissions four different target areas have been preset, and tests have been run on all those areas. The goal of this exercise was to get a stable transmission. But what is acceptable as stable can depend on what demands need to be fulfilled by the user. By evaluating the quality of the four different target areas and comparing it to the output power level used to keep the signal in the target area, data is presented to help one choose an appropriate method for one’s needs.

The areas used for the different test cases are chosen by looking at the statistical parameters found in the trial run and presented in Fig. 4.2. The areas are chosen by looking at how the statistical parameters relate to one another. The numerical values for each area are presented in Table 5.1. Three different limits are used, the inner limit is where the mean of the last 16 messages should be when using the short control group. The outer limit is for the mean of the long control group, i.e. the last 160 messages. The sigma limit is used as a sign that the signal is a big step away from the target and a bigger step needs to be taken.

In the Fig. 5.1 the bit error rate of the received packages is presented on
The bit error is calculated and a comparison between the quality of the signal versus the gain in saved energy is made.

Table 5.1: Preset limits for the four different test cases.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inner limit</td>
<td>103-112</td>
<td>100-108</td>
<td>98-106</td>
<td>95-105</td>
</tr>
<tr>
<td>Outer limit</td>
<td>100-114</td>
<td>98-112</td>
<td>95-110</td>
<td>90-105</td>
</tr>
<tr>
<td>Sigma limit</td>
<td>14-100</td>
<td>14-100</td>
<td>14-100</td>
<td>14-100</td>
</tr>
</tbody>
</table>

Figure 5.1: The bit error rate of the received packages with the four different parameters is represented with the red line, the number of lost messages with the blue. Both are represented by percentages on a log-axis.

In Fig. 5.2 the mean transmission power for each test case is shown. The behaviour might seem odd since it increases for case one and two when a reduction would be more expected. The theorised reason for this is that the signal tries to keep a lower transmission power which leads to more transmission errors and more frequently receiving a signal that is too low to be acceptable or completely lost. Therefore, it will more often make drastic increases in the transmission power and thus create a jumpier signal which will in its turn raise the mean.
Figure 5.2: The mean value which the method adjusts the output power level to during the use of the four different parameters.
Chapter 6

Conclusions and Future Work

6.1 Conclusions

The main specified goals have been met. The wireless devices can communicate in the basic setup in which the signal transmission power level can be changed according to small changes in radius and interference.

First extensive testing was carried out of the devices and the signal they transmit. Studying the error rates and the quality indicators LQI and RSSI for a variety of signal strengths and distances created a map of their behaviour. This was used as a base for programming a Matlab simulation of the two transceivers. In the Matlab environment a program for adjusting the transmission power according to the goals of this project was written. This program was later implemented on the devices and tested.

Extensive tests were done both for the initial setup and for the new code to give the experimental results a high statistical reliability. The limitations on the experiments were due to the environment. All the testing was done in an office environment only using the built-in antennas. Such antennas made the radius of use very limited and a small change in radius significantly changed the quality of signal received. Therefore, the method needs to be tested in various environments.

The simulated setup was tested thoroughly and the results did match
those from the experiment to a high degree. The behaviour of the power controlling program was also similar in the simulations as it was in the experiments. However, it was somewhat more prone to adjusting the signal strength in the experiments due to the interference of real life noise sources that were not included in the simulations.

Modifications on the program’s behaviour were necessary due to unforeseen problems that arose along the way. One such issue was the necessity of adding a way to deal with a completely lost signal.

Four different test cases were implemented using the statistical mean and sigma values of the received messages. The results of these are seen in Fig. 5.1 and Fig. 5.2. Which target should be used depends on the needs of the system, however a preference for test case ‘3’ exists due to it having a low mean transmission power with minimal errors occurring in the transmission.

6.2 Future Work

A primary aspect of future work would be adding acknowledgement packages (ACK). The current code assumes that the two devices are speaking continuously with each other, know about each other and are expecting messages at a continuous rate. If messages are sent only sometimes, implementing ACKs is necessary to make sure that the sent message is received. This is also important if each individual message needs to be received and individual packages lost in transmission need to be resent.

To improve the Matlab model, more random noise interference should be added. At the moment the model only transmits a signal that is randomised according to the distribution of the power level it is set to. The only way to introduce changes in the signal is to change the power level. Studying a random noise introduced at random intervals would prove interesting and possibly make the models behaviour closer to reality.

To use a higher end of the scale, i.e. if the signal needs to be louder either because the error rate needs to be lower or when using the method in
a different setting than the one used here. If the experimental results show that the signal is just not strong enough then RSSI can be used instead.
Bibliography


