Sending Location-Based Keys
Using Visible Light Communication

Abdalah Hilmia
Abstract

Sending Location-Based Keys Using Visible Light Communication

Abdalah Hilmia

In this thesis we explore the possibilities of using Visible Light Communication (VLC) to build a system that is capable of securely sharing small amounts of data in the form of encryption keys in a boundary defined area. We design and implement the MAC and Physical layers of the system. On the transmission part, the system uses BFSK and Manchester encoding to modulate the signal. The reception is handled in windows of fixed size where the Goertzel algorithm is applied to detect the transmitted frequencies. The transmitters synchronize their transmissions using a simple static scheduling method. Based on this system we then build a secret sharing platform where we utilize Shamir’s Secret Sharing protocol. We finally create different scenarios for this platform and test them. Those scenarios include variations of transmitter angles, different kinds of receivers with different sensitivity levels and different heights of the receiver. The results of this research show that the angle of the transmitters and the sensitivity of the receiver are in close relation to how the contours of the reception area would look like. The results also show that the height of the receiver and its sensitivity play a major role of how large this area is.
Acknowledgements

Special thanks to my supervisor Kasun Hewage for taking the time and effort to guide me through this project and answering all my unintelligent questions. My reviewer Thiemo Voigt for giving me this opportunity and for correcting and grading this thesis report. Christian Rohner for proposing the security part of the project and providing ideas for it. Ambuj Varshney for giving me inspiration and motivation and kicking away boredom in many occasions. To everyone in Uppsala Networked Object group for the nice company and meeting which taught me a lot. And finally to my friends and family who supported me and bared with me thought this thesis.
# Contents

Abstract iii

Acknowledgements v

1 INTRODUCTION 1
   1.1 Motivation and Problem Statement ........................................ 1
   1.2 Goal .............................................................................. 1
   1.3 Approach .......................................................................... 2
   1.4 Results ............................................................................ 2
   1.5 Thesis Structure ............................................................. 2

2 BACKGROUND 5
   2.1 Photodiodes .................................................................. 5
   2.2 Driving LEDs ................................................................. 5
   2.3 Binary Frequency Shift Keying (BFSK) .............................. 6
   2.4 Manchester Encoding ..................................................... 7
   2.5 The Goertzel Algorithm .................................................. 8
   2.6 Shamir’s Secret Sharing .................................................. 8

3 DESIGN OF THE PHYSICAL AND MAC LAYERS 11
   3.1 Frame Structure ............................................................. 11
   3.2 Data Transmission .......................................................... 11
      3.2.1 Modulation ............................................................... 13
      3.2.2 Synchronization ......................................................... 13
   3.3 Data Reception ............................................................... 14
      3.3.1 Frequency Detection .................................................. 15
      3.3.2 Frame Reception Scheme .......................................... 15

4 VLC FOR SECURITY 21
   4.1 The Propagation of Light .................................................. 21
   4.2 Proposed Scenarios ........................................................ 22
      4.2.1 Single Source Scenarios ........................................... 22
      4.2.2 Multiple Sources Scenarios ....................................... 22
   4.3 The Secret Sharing Protocol .............................................. 23

5 IMPLEMENTATION 25
   5.1 Hardware Components ....................................................... 25
   5.2 Software Components ...................................................... 27

6 EVALUATION 29
   6.1 VLC ............................................................................... 29
      6.1.1 The Goertzel algorithm ............................................. 29
      6.1.2 Frame Delivery Rates ................................................. 30
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.2</td>
<td>Security</td>
<td>31</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Single Source Scenarios</td>
<td>31</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Multiple Sources Scenarios</td>
<td>32</td>
</tr>
<tr>
<td>7</td>
<td>RELATED WORK</td>
<td>37</td>
</tr>
<tr>
<td>8</td>
<td>CONCLUSION AND FUTURE WORK</td>
<td>39</td>
</tr>
<tr>
<td>8.1</td>
<td>VLC</td>
<td>39</td>
</tr>
<tr>
<td>8.2</td>
<td>VLC for Security</td>
<td>40</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Photodiode</td>
<td>5</td>
</tr>
<tr>
<td>2.2</td>
<td>LED Driving</td>
<td>6</td>
</tr>
<tr>
<td>2.3</td>
<td>BFSK Modulation</td>
<td>7</td>
</tr>
<tr>
<td>2.4</td>
<td>Manchester Encoding</td>
<td>7</td>
</tr>
<tr>
<td>3.1</td>
<td>Frame Structure</td>
<td>12</td>
</tr>
<tr>
<td>3.2</td>
<td>Schedule Structure</td>
<td>14</td>
</tr>
<tr>
<td>3.3</td>
<td>Reception Without Tight Synchronization</td>
<td>16</td>
</tr>
<tr>
<td>3.4</td>
<td>Pilot Reception Flowchart</td>
<td>17</td>
</tr>
<tr>
<td>3.5</td>
<td>Bit Reception Flowchart</td>
<td>18</td>
</tr>
<tr>
<td>3.6</td>
<td>Frame Reception Flowchart</td>
<td>19</td>
</tr>
<tr>
<td>5.1</td>
<td>Tmote Sky Node</td>
<td>25</td>
</tr>
<tr>
<td>5.2</td>
<td>UPWIS Node</td>
<td>26</td>
</tr>
<tr>
<td>5.3</td>
<td>LEDSAVERS 12V, 38°, 320lm Bulb</td>
<td>26</td>
</tr>
<tr>
<td>5.4</td>
<td>OPT101 Monolithic Photodiode and Single-Supply Transimpedance Amplifier</td>
<td>26</td>
</tr>
<tr>
<td>5.5</td>
<td>Constant Current LED Driving Circuit</td>
<td>27</td>
</tr>
<tr>
<td>6.1</td>
<td>Computation Time of The Goertzel Algorithm on Sky</td>
<td>29</td>
</tr>
<tr>
<td>6.2</td>
<td>Packet Delivery Rates using Sky Platform</td>
<td>30</td>
</tr>
<tr>
<td>6.3</td>
<td>Packet Delivery Rates using UPWIS Platform</td>
<td>31</td>
</tr>
<tr>
<td>6.4</td>
<td>Single Source with and without Noise</td>
<td>32</td>
</tr>
<tr>
<td>6.5</td>
<td>Key Reception Area with 2 Sources at 90°</td>
<td>34</td>
</tr>
<tr>
<td>6.6</td>
<td>Key Reception Area with 2 Sources at 45°</td>
<td>35</td>
</tr>
<tr>
<td>6.7</td>
<td>Key Reception Area with 2 Sources at 60° Using UPWIS</td>
<td>35</td>
</tr>
</tbody>
</table>
Chapter 1

INTRODUCTION

1.1 Motivation and Problem Statement

Illumination is everywhere around us and has ample access to power [12]. **Light Emitting Diodes (LEDs)** are reliable, energy efficient, subject to industry standardization and have a long lifetime [1]. Those advantages enable LEDs to be a good replacement for incandescent bulbs which are being phased out [5]. LEDs are also capable of changing states rapidly; much faster than a human eye can detect [8]. Those factors make high frequency-modulated LEDs an efficient communication channel that is imperceptible to the human eye.

**Visible Light Communication (VLC)** is a data communication method which uses visible light as its signal and free-space as its communication medium. A typical VLC system consists of a microcontroller-driven LED light bulb for transmission and a microcontroller-driven photodetector for reception [13, 14, 12, 8].

In normal conditions, visible light is bound to the boundaries of eyesight and cannot escape through walls. This makes visible light an easier alternative to Radio Frequency (RF) waves when it comes to localization applications.

The technological advances have made it possible and inexpensive to build small low-power wireless sensor nodes consisting of a low-power processor, some memory, wireless transceiver and the required sensors [11]. Those sensor nodes are deployed in the environment to determine some physical behaviors and communicate with each other to create **Wireless Sensor Networks (WSNs)** [3]. WSNs have many applications, such as environmental aspects monitoring, heath-care monitoring [10], traffic control monitoring, habitat monitoring [9], localization, etc. WSNs have many research issues that affect different aspects of design and performance of the networks. Those applications and research issues attracted the attention and interest of researchers in recent years.

**Security** is one of the interesting applications of WSNs. The uses of WSNs for security varies from monitoring certain areas to sharing secret data and many other interesting applications.

This thesis proposes the use of a VLC system for sharing encryption keys in a boundary defined area using wireless sensor nodes.

1.2 Goal

The goal of this thesis work is to build a VLC system that runs on wireless sensor nodes for transmission and reception. We also intend to port the system on different wireless sensor platforms and compare their performance. Following this, we intend to use the implemented VLC system to share a small amount of data in the form of encryption keys in a boundary defined area. For this purpose, we will study the affects of different
aspects that define, interfere with and modify the shape and size of the area where it is possible to receive those keys. We will also propose and evaluate scenarios and a key sharing scheme which can be used to transmit the keys.

1.3 Approach

Our approach to fulfill the goal of this thesis is to analyze the different aspects related to the transmission of data using VLC, then design and implement a functioning system. This part is important since we need to build a system suitable for the secret data sharing part. Following this, we conduct lab experiments to test and evaluate the robustness and performance of the implemented system. The next step is to analyze the different aspects related to the propagation of light and how it effects the area of reception. Then we propose some scenarios based on the previous analysis and implement them using the implemented VLC system. Finally, we perform lab experiments to evaluate those different scenarios and draw conclusions.

1.4 Results

On the VLC part, the result of this project is a functional VLC system built on wireless sensor nodes. However, this system uses low data rates because of the hardware limitations. Testing the system for robustness over distance and background light variations shows: The photodetector with a higher sensitivity performs much better than the lower sensitivity one in low background lighting conditions. However, when the lighting condition is high, the high sensitivity sensor is saturated and fails while the lower sensitivity sensor maintains a reasonably good transmission over distance.

Regarding the use of that system for security. The result of our work is a study of how the different aspects of the proposed scenarios affect the area of the key reception. This study shows that the angle of the transmitter additional to the range and sensitivity of the receiver play a major role in defining the contour of the reception area. It also shows that the height of the receiver plays a role in defining the size of the reception area, however it does not change how this area looks like.

1.5 Thesis Structure

In Chapter 1 Introduction we introduce the problem and the motivation behind it. Then state the goal of the thesis and the approach we followed to fulfill it. Finally, we summarize the results of this thesis work.

In Chapter 2 Background we go into some of the topics in the field of data communication, electronics and cryptography. That is to provide a brief background about the topics the reader should know about before proceeding to read this thesis.

In Chapter 3 Design of the MAC and Physical Layers we go through the design of the MAC and Physical Layers of the VLC system in details. First, we introduce the frame structure to be used in this system. This chapter is then split into two main parts. The first part handles the topic of data transmission, where we explain in detail the modulation scheme and synchronization of transmitters. The second part talks about data reception. It goes through the frequency detection technique and an in depth explanations of how the frame reception state machine works.
In Chapter 4 VLC for Security we go into the design of our security system. This is done by discussing the aspects that affect the propagation of light based on our observations. Based on those observations we later propose some possible set ups to be tested and evaluated.

In Chapter 5 Implementation we dive into the technical details of the hardware and software we use. It is divided into two section: The hardware components section where we describe the hardware we use to implement this system. The software components section where we illustrate some details about the implemented modules with focus on the introduced limitations.

In Chapter 6 Evaluation we test and evaluate the VLC and security parts of the implemented system. In the VLC part we first evaluate the time it takes for the Goertzel algorithm to compute. Following that we evaluate the frame delivery rates of both the implemented platforms and draw conclusions. Regarding the security part we describe the experimental setup, then present and go through the results collected from evaluating the different proposed scenarios.

In Chapter 6 Related Work we summarize and discuss different published work in the field of VLC.

In Chapter 7 Conclusions and Future Work we summarize the results of the work regarding both VLC and security parts of the thesis. Then, we point out and propose improvements. Finally we suggest how to build on this work to build a security system that is eligible for real-world deployment.
Chapter 2

BACKGROUND

In this chapter we go through some of the topics needed to proceed reading this thesis. Regarding the hardware we use, we explain basic principles about Photodiodes and LED driving. On the software part, we explain the modulation schemes, tone detection and secret sharing protocol we make use of.

2.1 Photodiodes

Photodetectors are devices used for light sensing. Photodiodes are a commonly used type of photodetectors. A photodiode behaves similarly to an ordinary diode, but it generates an electrical current when it absorbs light. Photodiodes have low price, long lifetime, high linearity and fast response. Photodiodes have two modes of operation: photoconductive and photovoltaic. Operation mode selection depends on required speed and dark current tolerance. In Photoconductive mode, an external reverse bias is applied. Using this mode increases responsivity and linearity but produces a larger dark current which can be limited depending on the photodiode material. In Photovoltaic mode the photodiode is zero biased. Using this mode produces minimum dark current. Dark Current is a leakage current that flows through the photodiode despite the absence of light when a bias voltage is applied to the photodiode. Applying a higher bias increases the dark current produced though the photodiode. The material of the photodiode affects the amount of dark current present in present.

![Photodiode](http://www.thorlabs.com/thorproduct.cfm?partnumber=FGAP71)

**Figure 2.1: Photodiode.**

2.2 Driving LEDs

Driving LEDs properly improves efficacy, increases reliability and extends lifetime. There are three commonly used ways for driving LEDs: Resistor-based Regulation, Constant Current Linear Regulators and Switching Regulators.

---

1 Figure source: http://www.thorlabs.com/thorproduct.cfm?partnumber=FGAP71
2 OSRAM LED Fundamentals: http://ledlight.osram-os.com/knowledge/led-fundamentals
Resistor-based Regulation is the simplest way of driving LEDs but is also a less efficient way of doing it. The resistor value depends on the source voltage \( V_s \) and the current draw of the LED \( I_f \) and it is computed using Ohm’s Law:

\[
R = \frac{V_s - V_f}{I_f}
\]  

(2.1)

Where: \( V_f \) is the voltage the LED requires at \( I_f \).

This means that the voltage drop from \( V_s \) to \( V_f \) is wasted as heat in the resistor which, depending on the source voltage, can be significant. The dependency on the source voltage brings up another problem with Resistor-based Driving as any voltage source has some tolerance. The variation in the voltage due to this tolerance will change the LED current draw which is undesirable.

Constant Current Linear Regulators are preferred over resistor-based regulators because it can take a wide range of voltage input and outputs a constant current, which is ideal for driving LEDs. Similar to resistor-based regulation, linear regulators are very inefficient. The delta between input and output voltages \( V_s \) and \( V_f \) is also wasted as heat within the regulator.

Switching Regulators feed the load with small portions of input energy at a time to maintain a static output value. They use an electrical switch with a controller. The controller generates a PWM signal controls its rate. This PWM signal is used to turn the switch on and off to maintain the value of the output. There are four different operation modes of switching regulators, namely buck mode, boost mode, buck-boost mode and fly back mode.

2.3 Binary Frequency Shift Keying (BFSK)

Frequency Shift Keying (FSK) is a frequency modulation technique in which discrete frequencies are used to transmit digital data. Binary FSK (BFSK) uses a pair of discrete frequencies to transmit binary data (0 and 1). Figure 2.3 illustrates the time domain of a BFSK modulated carrier wave.
2.4 Manchester Encoding

Phase Shift Keying (PSK) is a digital modulation technique in which data is transmitted by modulating the phase of the carrier wave. Binary PSK (BPSK) uses two phases separated by 180°. Manchester encoding is a special case of BPSK and it is one of the most common data coding methods. Manchester encoding represents data by a transition in the state which means that each data bit has at least one transition in its period. Therefore, it has no DC component and is self-clocking. Figure 2.4 illustrates the Manchester encoding of "10100111001" bit sequence.

Figure source: https://commons.wikimedia.org/wiki/File:Fsk.svg
2.5 The Goertzel Algorithm

The Goertzel Algorithm [7] is a Digital Signal Processing technique which analyses frequency components similarly to Discrete Fourier Transform (DFT) when the target frequencies are known. The Goertzel Algorithm can perform frequency detection using much less CPU power than DFT when the number of the target frequencies is low. The Goertzel Algorithm computes the relative squared magnitude of a frequency $f$ in an input sequence $x(n)$ of size $N$ where $n \in N$. The first step of the computation is to get the value of $K \in \mathbb{N}$ which indicates the frequency bin of the DFT.

$$K = \frac{N \times f}{f_s}$$ (2.2)

Where: $f_s$ is the sampling rate.

The second step is to compute the values of $s(N - 1)$ and $s(N - 2)$:

$$s(n) = x(n) + 2\cos(2\pi \frac{K}{N})s(n - 1) - s(n - 2)$$ (2.3)

Then get the real and imaginary components:

$$c_r(K) = s(N - 1) - s(N - 2)\cos(2\pi \frac{K}{N})$$ (2.4)

$$c_i(K) = s(N - 2)\sin(2\pi \frac{K}{N})$$ (2.5)

And finally get the relative squared magnitude:

$$m(K)^2 = c_r(K)^2 + c_i(K)^2$$ (2.6)

An optimized way to reduce the computation needed to get $m(K)^2$ would be to replace the last two steps with:

$$m(K)^2 = s(N - 1)^2 + s(N - 2)^2 - 2\cos(2\pi \frac{K}{N})s(N - 1)s(N - 2)$$ (2.7)

However, this comes with the expense of losing phase information.

From the precious equations, it is clear that the complexity of the Goertzel algorithm is $O(MN)$ where $N$ is the number of samples and $M$ is the number of DFT terms. The complexity of the Fast Fourier Transform (FFT) is $O(N \log(N))$. Therefore it is less expensive to use the Goertzel algorithm over FFT while $M \leq \log(N)$.

2.6 Shamir’s Secret Sharing

Shamir’s Secret Sharing [15] is an algorithm created by Adi Shamir. It is a cryptography algorithm that splits data into a number $N$ of portions where only $K \leq N$ number of them is enough to reconstruct the data. The knowledge of any number $i \leq K$ should not provide any information about the secret.

The basic idea of Adi Shamir is that 2 points are enough to define a line, 3 points are enough to define a parabola and 4 points are enough to define a cubic curve. i.e. a polynomial of degree $K - 1$ is defined by at least $K$ points.
By knowing the threshold number $K$ we define the degree of the required polynomial to be $K - 1$. Then we place the secret $S$ to be the coefficient of $x^i$ where $i = 0$ and generate $K - 1$ random numbers $a_i$ where $K > i > 0$ to be the coefficients of $x^i$ where $K > i > 0$. The polynomial then looks like:

$$f(x) = S + \sum_{i=1}^{K-1} a_i x^i$$

Then we create $N$ number of points $(x, f(x))$, where $N > i \geq 0$ and distribute them among the participants.

When $K$ number of share are collected, it becomes possible to reconstruct the secret. This is usually done by computing the Lagrange Interpolation.

$$f(x) = \sum_{i=0}^{K-1} y_i l_i(x)$$

This is a linear combination of Lagrange basis polynomials.

$$l_i(x) = \prod_{0 \leq m \leq K-1, m \neq i} \frac{x - x_m}{x_i - x_m}$$

Solving this equation will result in reconstructing the previous polynomial. Thus reconstructing the secret.

Although this algorithm works fine, there is a problem where an attacker can win some information if they collect a number $i < K$ of the shares. This problem is solved using Finite Field Arithmetic in a field of size $p \subseteq \mathbb{P} : p > S, p > N$. 

$$f(x) = S + \sum_{i=1}^{K-1} a_i x^i$$
Chapter 3

DESIGN OF THE PHYSICAL AND MAC LAYERS

The system we build is based on wireless sensor nodes. A typical VLC system that runs on wireless sensor nodes consists of a set of transmitter nodes whose objective is to transmit data or part of data to one or more receiver nodes.

There are many wireless sensor node platforms that can be used to create this system. We chose to work with Tmote Sky\(^1\) with a 3.9MHz MSP430 MCU because of its simplicity. Then we take the next step by porting the implementation to UPWIS platform\(^2\) with 32MHz STM32L ARM Cortex MCU because of its high capability.

While we can try and program those nodes directly, it is much easier, more convenient and less problematic to use an operating system (OS). There are quite a few Internet-of-Things Operating Systems (IoT-OS) to choose from. Contiki-OS\(^3\) is one of those operating systems. Contiki-OS is an open source OS that supports a wide range of hardware platforms. It also provides memory allocation and a simulator software which makes it easier to test and debug code before deploying it to the hardware. Therefore we use Contiki-OS as an underlying platform in this system.

The Physical VLC Layer is responsible for the delivery of Physical Protocol Data Units (PPDUs) – which are refereed to as frames – over the free-space medium. The MAC VLC Layer is responsible for providing data delivery services for the Application Layer. It is also responsible for synchronization with other nodes to prevent collision.

3.1 Frame Structure

The frame structure used is a modified version of IEEE 802.15.4 standard frame structure. The Physical Protocol Data Unit (PPDU) consists of 2 bytes for preamble, 1 byte for Start Frame Delimiter (SFD), 1 byte for Length and 4 to 127 bytes for the MAC Protocol Data Unit (MPDU). The MPDU structure consists of 1 byte for Sequence Number (SN), 1 to 124 bytes of payload and 2 bytes for Frame Check Sequence (FCS). Figure 3.1 illustrates the frame structure.

3.2 Data Transmission

Transmission using VLC require some additional hardware that might not always be included in a typical wireless sensor node. A high power LED bulb is a basic additions needed to a typical wireless sensor node. There are lots of available off-the-shelf

---

\(^1\)http://www.snm.ethz.ch/Projects/TmoteSky
\(^2\)http://upwis.com
\(^3\)http://contiki-os.org/
LED bulbs that can be candidates for such a system. We examined some commercial bulbs to find out which are easiest to drive. We chose to proceed the experiments with LEDSAVERS 12V, 38°, 320lm bulb\(^4\). This LED light bulb should be driven properly to improve efficacy, increase reliability and extend lifetime. There are three common ways to driver LEDs: Resistor-based Regulation, Constant Current Linear Regulators and Switching Regulators. While resistor-based regulation is too inefficient and switching regulators have an internal oscillation that interfere with our driving frequency, we decided to use constant current linear regulators. Although this regulation method is inefficient – since the voltage drop is wasted as heat – it provides a constant current which is suitable for driving the LED safely and properly.

Each transmitter node consists essentially of a microcontroller with timer capability, an attached high power LED light source and an RF module.

The microcontroller performs the MAC and Physical layer procedures to achieve synchronization with nearby nodes, generate the signal output and control the LED light source. The LED light source propagates the generated signal over the free-space medium while keeping the lighting environment static for human perception. The RF module is used for synchronization among transmitter nodes when two overlapping nodes need to transmit signals at the same time. This time synchronization is needed to ensure collision-free transmissions.

When the Application Layer requests data to be sent, the MAC Service Access Point (MAC-SAP) send function computes the Cyclic Redundancy Check (CRC) and builds an MPDU with the provided data, Sequence Number (SN) and CRC. Then waits for the node’s transmission slot to pass the MPDU to the Physical Layer. The Physical Service Access Point (PHY-SAP) send function then constructs the PPDU from the MPDU data passed from the MAC-SAP by adding the SFD and preamble. The PHY-SAP send function then pushes the frame into the data buffer and starts modulating the bits.

\(^4\)http://www.kjell.com/se/sortiment/el/belysning/lampor-ljuskallor/led-lampor/ledsavers-led-spotlight-gu5-3-320-lm-p64056
3.2.1 Modulation

To achieve this transmission there are many modulation schemes proposed in VLC systems. Those schemes are very similar to those of radio frequency communication such as One Off Keying (OOK), Pulse Position Modulation (PPM), Pulse Amplitude modulation (PAM). However, most of them require special hardware, complex decoding algorithms or special workarounds to avoid flicker. In our approach, Binary Frequency Shift Keying (BFSK) with, optionally, Manchester encoding are used to modulate the light signal. BFSK is used because of its simplicity and natural prevention of flicker, as well as its high Signal to Noise Ratio (SNR) even when multiple signals are being transmitted in the same medium. Manchester encoding is used because it provides more robust synchronization and data delivery.

In order to use this modulation scheme we use a pair of frequencies \( f_0 \) and \( f_1 \) to represent binary data symbols 0 and 1. When Manchester encoding is disabled, one binary symbol is represented by the propagation of one frequency for a frequency propagation-period, equals to a bit period \( T_f = T_b \). i.e. propagating \( f_0 \) for a period \( T_b \) represents 0 and, similarly, \( f_1 \) represents 1. If Manchester encoding is enabled, the presence of both frequencies is needed to represent one symbol. Each frequency is propagated for a frequency propagation-period, equals to half a bit period \( T_f = \frac{T_b}{2} \), to generate the transition in frequency needed to detect the symbol value. i.e. the transition \([f_1, f_0]\) is needed to represent binary symbol 0. This is achieved by propagating \( f_1 \) for a period \( \frac{T_b}{2} \) followed by \( f_0 \) for a period \( \frac{T_b}{2} \). Similarly, \([f_0, f_1]\) is needed to represent binary symbol 1.

Additionally, we use a Pilot frequency \( f_p \) to modulate the frame preamble. \( f_p \) is propagated for the period needed to transmit 16 bits \( (16T_b) \) at the beginning of each frame to announce that a frame will be transmitted.

To maintain the consistence of the light intensity an Idle frequency \( f_i \) is propagated by all transmitter nodes when not transmitting any data. \( f_i \) should not pose any interference to other transmitters in the same optical medium.

3.2.2 Synchronization

Since we could have multiple transmitters in the same optical space, time synchronization is required. There are many ways to achieve this synchronization through medium access protocols such as Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA), Code Division Multiple Access (CDMA), Orthogonal Frequency Division Multiple Access (OFDMA) and Spatial Division Multiple Access (SDMA). Since synchronization is not one of our priorities a simple static schedule synchronization method is used. The idea for such a schedule came from schedule-based communication in Low-power Wireless Bus (LWB) [6]. Although their method is adaptive, it is much more complex. There are two proposed approaches to implement this synchronization, one of them is to use in-band synchronization using visible light which means that the transmitter nodes have to be in visible range of each other and receivers will be aware of this synchronization. This is not practical for our purposes. The other approach is to use out-of-band synchronization using Radio Frequency (RF) signals based on IEEE 802.15.4 standard for Low-Rate Wireless Personal Area Networking (LR-WPANs), which is a good alternative.

We organize the schedule to include two types of transmitter nodes, namely host node and source nodes. The host node is responsible for creating the and transmitting the schedule. The source nodes receive and follow the schedule. Since it is a waste
of resources to have one node just for scheduling and since no node transmits during schedule creation and propagation time, the host node is also a source node.

We built the schedule structure to have a single schedule creation slot \( (t_{\text{schedule\_create}}) \) when the host node launches. The created schedule contains transmission slots \( (t_{\text{transmit}}) \) during which source nodes have the time to transmit one frame. A gap time \( (t_{\text{gap}}) \) is introduced between the each \( t_{\text{transmit}} \) to prevent collision due to small clock and synchronization errors. The sum of \( t_{\text{transmit}} \) and \( t_{\text{gap}} \) plus the schedule creation time \( (t_{\text{sync}}) \) should be less or equal to round period time \( (t_{\text{round\_period}}) \). Figure 3.2 illustrates the structure of schedule round period.

![Schedule Structure Diagram](image)

**Figure 3.2: Schedule Structure.** In the first slot the host node creates the schedule and propagates it to the other nodes. Each node has its own slot during which it can transmit. The gap is introduced to avoid collision due to small synchronization errors.

When the host node boots, it creates the schedule and propagates it to the source nodes which have their radios on and waiting. When the source nodes receive the schedule the synchronize their clocks to the host’s. All source nodes now wait \( t_{\text{gap}} \) time while the host nodes waits \( t_{\text{gap}} + t_{\text{sync}} \) time. That is to take in mind the time required for the frame to reach the source nodes. After this waiting time all nodes follow the same schedule. During the schedule each node transmits VLC data in its assigned slot keeping a \( t_{\text{gap}} \) time between slots. When all scheduled slots are passed the source nodes turn on their radios and wait for the schedule. The host transmits the schedule after \( t_{\text{round\_period}} \) time.

This synchronization is simple yet good enough to keep the nodes well synchronized to suit our purposes.

### 3.3 Data Reception

A receiver node consists essentially of a microcontroller with timer and Analog to Digital Conversion (ADC) capability, and a light intensity sensor.

There are lots of variations of photodetectors that can be used. Photodiodes are frequently used as photodetectors because of their low price, long lifetime, high linearity and fast response. Photosynthetic Active Radiation (PAR) sensors are photodiodes that detect the spectral range of solar radiation from 400 to 700 nm which is more or less the light visible to the human eye. Tmote Sky has an on board PAR sensor which we use for this system. On the other hand UPWIS platform does not have built in photodetector. Thus, we use TI OPT101 Monolithic Photodiode and Single-Supply Transimpedance Amplifier\(^5\) because of its high sensitivity and built in Operation Amplifier.

The microcontroller detects the light intensity in the free-space medium by sampling the photodiode. The microcontroller also handles frequency detection and frame reception on the MAC and Physical levels.

Data reception in the Physical Layer is initiated from the MAC Layer, where the latter uses the Physical Layer Service Access Point (PHY-SAP) to start listening to the medium. When a frame is received, the frame is returned to the MAC Layer for CRC verification. The MPDU payload is then returned to the Application Layer.

### 3.3.1 Frequency Detection

Since BFSK is used for transmission, the receivers should be able to detect and distinct different propagated frequencies. While Discrete Fourier Transform (DFT) is the first thing to come in mind when talking about frequency detection, it is too expensive to run on the resource-constraint devices we are targeting. This makes the use of DFT unfeasible. The Goertzel algorithm \[^7\] can perform frequency detection using much less CPU power than DFT when the number of the target frequencies is low. This makes the Goertzel algorithm a suitable frequency detection method for our system.

We handle frequency detection in a window manner. When sampling the medium, the collected samples are organized into windows of constant size. Those windows are then sent to be analyzed using the Goertzel algorithm. The algorithm computes the magnitude of each of our target frequencies. The presence of each of those frequencies is then determined by thresholding the magnitude values.

Varying light conditions makes finding a convenient threshold a bit tricky. The photodiode circuits we use do not have hardware gain controlling. Therefore, software-based gain controlling is needed. We take the mean of samples where no gain controlling is needed as a reference mean. The ratio of the current window mean to the reference mean is then used to scale down the input. This simple gain controlling method makes it easy to find a suitable threshold that fits different lighting conditions and distances.

### 3.3.2 Frame Reception Scheme

When listening is initiated, the node starts sampling the free-space medium and generating windows. We designed a state machine to handle a successful frame reception within a given timeout. Reaching the timeout without a successful frame reception aborts the state machine and returns a timeout flag.

In the beginning, a receiver node is only interested in listening to the Pilot frequency \((f_p)\). The Goertzel algorithm is used compute the relative squared magnitude of \(f_p\). The magnitude is then used to query the presence of \(f_p\) against a threshold. \(f_p\) is transmitted for the period of 16 bits \((16T_b)\). Since the Pilot is a single frequency \((f_p)\), only two windows per bit period are needed \((T_w = \frac{T_b}{2})\). Which means \(f_p\) should be present for 32 consecutive windows. When \(f_p\) is detected for the first time, a counter is started. The counter is incremented every time \(f_p\) is detected in a window. When \(f_p\) is present for 32 consecutive windows, the Pilot-received flag is set and the Pilot reception state machine is terminated. If the Pilot frequency disappears before the counter reaches 32, the counter is reset and the state machine is restarted. Figure 3.4 illustrates the \(f_p\) reception mechanism.

The Pilot frequency is used to indicate the beginning of a frame and synchronize frequency detection windows with the beginning of the bit stream. After receiving the
When a bit reception is complete the bit is buffered and the bit reception state machine is restarted. When enough bits are buffered to construct a byte, this byte is placed in its designated location in the frame. The second received byte is the length. When the length is received the expected frame length is updated so the receiver node knows exactly the number of bytes it should receive. When the last byte is received the frame data is returned to the MAC Layer and the state machine is terminated. Figure 3.6 illustrated the frame reception mechanism.

When a frame is returned to the MAC layer, CRC verification is performed to check the MPDU data. (possibly talk about the polynomial). If the frame passes the CRC verification, the MPDU payload is returned to the Application Layer. Otherwise the frame is discarded and the reception state machine is restarted.
Figure 3.4: Pilot Reception Flowchart. This figure illustrates the mechanism using which the receiver nodes waits for the $f_p$. 
Figure 3.5: Bit Reception Flowchart. This figure illustrates the mechanism using which the receiver nodes receives a bit.
FIGURE 3.6: Frame Reception Flowchart. This figure illustrates the mechanism using which the receiver nodes receives a frame.
Chapter 4

VLC FOR SECURITY

As mentioned earlier, this thesis aims to explore the use of VLC to secretly share a small amount of data in a boundary defined area. There are many aspects that play a role in how this area looks like. In this chapter we discuss those aspects and propose different scenarios to be tested. We also describe the secret sharing protocol we use.

4.1 The Propagation of Light

When thinking of the aspects related to how the area in which the light intensity of a specific bulb is perceptual looks like, the intensity of the light and sensitivity of the receiver play the major role. Those factors are in close relation to each other. They mainly determine the size of the area. i.e. the stronger the light and the more sensitive the receiver, the larger the area.

The angle of the bulb and its beam angle also play a major role. The angle of the bulb defines the elliptical shape of the beam while the beam angle plays a role in the size of this ellipse. i.e. the closer the bulb angle is to 90°, the more the beam ellipse looks like a circle, while the further the angle is the more the ellipse looks like an egg with the narrower part closer to the light. On the other hand, the wider the beam angle is the bigger this ellipse will be.

The measuring distance also plays a role. The further you measure – with regards to having enough light intensity – the bigger the perceptual area is.

The presence of other light sources in the same medium plays a major role. Other light sources affect the light intensity by increasing the overall intensity and decreasing the difference between on and off states of the transmitting bulb. This makes it harder for the receiver to measure the difference and makes thresholding even harder. Therefor, the signal reception area is shrunk.

It is an interesting relation between the presence of other light sources, the intensity of the transmitter light and the sensitivity of the receiver. While a sensitive receiver is more likely to be able to receive a signal despite the presence of other lights, higher light intensities are likely to saturate it. This is because there is often a trade-off between resolution and range in receivers. On the other hand, a receiver with lower sensitivity is less likely to be saturated. This makes it easier for it to receive in high light intensities, however harder in lower intensities.

Other factors that affect the area of reception are the reflection of light over objects, walls and lenses. Those factors we will not look into as they are more complicated to work with and require a different set of expertise.

The previously mentioned aspects matter the most when only one transmitter is introduced in the medium. i.e. the key is shared using the beam area created by one light. In the case of having the key shared on multiple light sources the related aspects increase and the reception area differs. The reception area is no longer defined by
having enough light intensity to receive but also includes having enough intensity from both lights and the interference between the lights.

Intuitively speaking, the area where the receiver receives close enough light intensity from both lights – taking in mind that it is not too high or too low depending on the sensitivity of the receiver – should be the ideal area for reception. This is because the interference between the lights is minimal. On the other hand, the more light intensity a receiver receives from one of the sources, the harder it will be to receive from the other. This is because the high light intensity from one of the lights makes the difference in the intensity that the other light poses smaller and thus harder to detect.

After going through those aspects we should mind you that it is extremely difficult to build a system with unbreakable security. In other words, it is always possible to argue that this system is breakable by using a very sensitive receiver with a high range. Such a receiver can detect higher and lower light intensities wherever it is possible to detect. Of course, this is true taking in mind the scope of this thesis. But there is much more to do than what we have done to make it secure against a larger range of possible attacks.

4.2 Proposed Scenarios

In this section we propose some possible setups to be used which will later be evaluated.

4.2.1 Single Source Scenarios

As mentioned in the previous section, there are many aspects that play a role in defining how the reception area looks like. While there aren’t many setups to test with a single transmitter light, the effect of the presence of other light sources will be looked into. The scenarios that will be tested are the following:

- single light transmitter with no other light source. It is simple to say that it is possible to receive the key wherever there is enough light intensity from the transmitting light. Which means that this area depends mainly on the light intensity and the sensitivity of the receiver.

- Single light transmitter where other lights are present. In this case the area is most likely to shrink away from the other light and closer to the transmitter light.

4.2.2 Multiple Sources Scenarios

After looking into single source setups it seemed to be interesting to see how transmitting the key from two different lights can affect the shape and size of the area. This opens up the doors to many setups that can be tested. We will propose setups that vary in bulb angle, distance to the receiver and receiver sensitivity. All the proposed setups will use the same light bulb and have the same height from the ground and distance between them. Those setups are the following:

- Different angles. As mentioned earlier the angle plays a role in how the reception area looks like. We set up both lights to face each other with an angle of 45°, then with 60°. Later on, they were placed facing the ground with 90°.
• **Different distances.** Measurements from all three previous setups where taken from two distances, ground level and 1m height.

• **Different receivers.** Measurements from the previously mentioned six setups where taken from two different receivers, the receiver on board Tmote Sky and TI OPT101 attached to UPWIS node.

The twelve setups mentioned earlier will be evaluated and tested in the *Evaluation* section.

### 4.3 The Secret Sharing Protocol

Since most of the proposed scenarios require transmitting a key from two light bulbs, a secret sharing method is needed. Shamir’s Secret Sharing [15] – which we explain in Chapter 2 *Background* – is a cryptography algorithm that splits data into a number $N$ of portions where only $K \leq N$ number of them is enough to reconstruct the data. This protocol allows us to split the key on two or more nodes and share them so that only receivers which have the possibility to receive enough shares of the key can reconstruct it. This protocol is particularly interesting because of its simplicity and robustness. Each of the transmitter nodes will have one share of the key and will transmit this share. The receiver should be able to pick up both shares to reconstruct the required key.
Chapter 5

IMPLEMENTATION

This chapter illustrates what we implemented and the architecture we used to make this system feasible in more details.

As mentioned earlier, there are hardware and software parts which are needed to implement this system. To put everything together, the overall structure of the systems is the following.

5.1 Hardware Components

For this system, we use some already available hardware and create others. The hardware components that were used in this system are the following:

- A set of Tmote Sky nodes. We use those nodes for both transmission and reception. Figure 5.1.

![Tmote Sky Node](image)

**Figure 5.1: Tmote Sky Node.**

- A set of UPWIS nodes. We use those nodes for transmission and reception also. Figure 5.2.

- A set of LEDSAVERS 12V, 38°, 320lm bulbs. Those bulbs were used to propagate the signal. Figure 5.3.

- A constant current linear regulator LED driving circuit for each bulb. We use those circuits to properly drive the LED bulbs. Figure 5.5b.

- A TI OPT101 Monolithic Photodiode and Single-Supply Transimpedance Amplifier. We use it to allow UPWIS to receive light signals since it does not have a built-in light receiver. Figure 5.4.
Chapter 5. IMPLEMENTATION

Figure 5.2: UPWIS Node.

Figure 5.3: LEDSAVERS 12V, 38°, 320lm Bulb.

Figure 5.4: OPT101 Monolithic Photodiode and Single-Supply Transimpedance Amplifier.
The only hardware component that was created in this system is the constant current linear regulator LED driving circuit. Figure 5.5a is a block diagram of this circuit. LM338T is an adjustable linear regulator which is used to provide a constant current supply to LED1. In operation, LM338T develops a nominal 1.25 V reference voltage between its pins 2 and 1. As the reference voltage is constant, a constant current flows through R2 and LED1. As LED1 is supposed to be turned on and off very fast when communicating, an N-channel MOSFET (Metal–Oxide–Semiconductor Field-Effect Transistor) (Q1) is used to drive LED1. The PWM signal is given by a microcontroller which provides an output signal that typically is limited to a few milliamperes of current. Therefore, a gate driver (IC2) is used to provide the high current required for Q1 in order to have fast switching. The parasitic inductance from wires and resistors result in voltage overshoot and ringing when switching Q1 very fast. Therefore, C1 together R3 is used as a voltage snubber for minimizing voltage overshoot and ringing. In order to reduce high frequency noise in the power supply, C2 and C3 are used as decoupling capacitors.

Figure 5.5: Constant Current LED Driving Circuit.

5.2 Software Components

In the software part, some available software was used to create the underlying structure for implementing the functionality of this system. The software components that were used in this system are the following:

- Contiki-OS as an underlying structure for the system.
- Generation of BFSK signal Using PWM.
- Static scheduling medium access control technique using RF. We use it to synchronize transmissions in the same medium.
- The Goertzel algorithm.
• Shamir’s Secret Sharing.

To generate the required PWM signal a hardware timer is required. Thus, Tmote Sky and UPWIS platform needed a slightly different implementation to generate the signal. The Sky platform only has two hardware timer where one of them is used by Contiki-OS for the clock module. The second timer is free to use, however, it is not directly connected to an output pin. This means that generating a PWM signal by directly toggling a pin is not feasible. Therefore, system level interrupts were used to toggle the pin. Interrupt handling with high rates on the 3.9 MHz MCU the Sky platform runs is too expensive. This makes it hard to propagate high frequency values. One the other hand, the UPWIS platform is more powerful with more hardware timers and a configurable pin configuration. This allows the use of one of the timers to directly toggle one of the pins. Hence, granting it the ability to generate high frequency values with high resolution. We use this method to generate the PWM signal on the UPWIS platform.

The implementation of the scheduler works with minor modifications on both Sky and UPWIS since it relies on Contiki-OS Rime \[4\] stack for communication.

Although the Goertzel algorithm is far less expensive than DFT where it is used in this system, it is still quite expensive. Hence, we introduce a fixed-point implementation of the Goertzel algorithm to reduce the computation cost of the algorithm since floating-point arithmetics are usually very expensive on resource-constraint devices. Although, this implementation performs much faster, it has a slightly problematic limitation. The coefficients used to compute the magnitude of the frequency are only allowed to be unsigned integers. This means that it only works for target frequency values which are lower than the quarter of the sampling frequency. This limitation can be compensated, however we chose to proceed into looking at more important topics. Although this implementation drastically reduced the time needed to apply the Goertzel algorithm, it still turned out to be expensive to use on the Sky platform. Some measures were taken to allow the algorithm enough time to compute before the deadline. Those measures included the reduction of the sampling rate \(f_s\) to around 3276 Hz and the number of samples per given window to 51 samples per window \(\text{spw}\). Given that at least two windows are needed to receive a frequency \(\text{wpf}\) and following this equation.

\[
n_{bps} = \frac{f_s}{n_{spw} \times n_{wpf}} \tag{5.1}
\]

Those limitations affected the transmission even further by limiting the transmitted frequencies to below 800 Hz and the data rate to a maximum of 32 bps which is very low but works well enough for our purposes.

While UPWIS does not suffer from the limitations introduced by the Sky platform, the use of OPT101 introduces another problem. OPT101 has a bandwidth of 14 KHz which means the sampling rate cannot go higher than this value. Using a sampling rate \(f_s\) of 13312 Hz, the same number of samples per window and window per frequency as in Sky and following equation 5.1 results in a maximum of 128 bps. While this is a big improvement, it is still a low data rate. The maximum transmission frequency is limited to 3 KHz which is good enough.

Shamir’s Secret Sharing algorithm was implemented using Finite Field Arithmetic in a field of size \(2^8\). This means that the string of input data to the function will be divided into bytes where each byte will have its own polynomial and own shares. One share from each byte are then put together to generate the final shares to be distributed to the nodes. This implementation is simple yet provides a very decent security level.
Chapter 6

EVALUATION

In this chapter we perform experiments to evaluate the VLC and security parts of the system.

6.1 VLC

There are multiple aspects in the VLC system to be tested. Here we are interested in the performance and robustness of the system. Hence, we perform experiments to evaluate the Goertzel algorithm computation time and the frame delivery rates in various scenarios.

6.1.1 The Goertzel algorithm

As mentioned earlier, the computation time of the Goertzel algorithm limited the data rates on the Tmote Sky platform. Sky’s small processing power makes the Goertzel algorithm seem expensive. Figure 6.1 shows the time needed to compute the algorithm in two cases. No sampling means that the time shown in the figure is computed when
the algorithm runs without being preempted by the sampling timer. In our system the algorithm should run while the receiver is collecting samples for the next window. This means that the algorithm is preempted by the sampling timer. The interrupt handling routine introduces an extra delay which is shown in the figure. This result shows concrete relation to the complexity of the algorithm, where the time required to compute the algorithm is a multiple of the number of the DFT terms (target frequencies). When using sky and since we need to compute the algorithm for only 2 target frequencies per window, the interval of each window should be slightly over $8\text{ms}$. That is to account for time needed to compute the magnitudes and other bit reception operations. Thus, we conclude that the limitations we put on the Sky platform were too conservative, since the used window interval is $15.6\text{ms}$. Hence, opening a door for improvement. The computational power of UPWIS allows it to compute the Goertzel algorithm in time without posing additional limitations on the data rate.

6.1.2 Frame Delivery Rates

There are quite a few factors that interfere with the quality of reception. Those factors mainly are the light intensity of the bulb, sensitivity of the receiver, distance between the source and receiver, the intensity of the background light and the duty cycle of the transmitter. We are interested in experimenting how our system performs with a variety of those factors. Figure 6.2 shows the results of the experiments conducted to evaluate the Packet Delivery Rates (PDR) on Sky platform. The experiments were conducted by sending 200 packets of $16B$ payload. This was repeated 3 times, then the results were averaged. This limitation is due to time constraints since the data rate is very low. The second limitation was the distance. As the charts show, the reception exceeds $5\text{m}$, although we stopped there for time and space constraints. In the case of high background lighting conditions, the receiver was set not to face the background light source but face the transmitter.

When transmitting at $16\text{bps}$ with 4 windows per frequency ($4\text{wpf}$) and little background light ($\sim 20\text{lux}$), the reception is stable until $4\text{m}$ with a few errors starting at $5\text{m}$. It also seems stable until $4\text{m}$ when the background light is increased to ($\sim 400\text{lux}$), however, this stability drops at $5\text{m}$. This drop can be explained by the lack of sensitivity of the receiver. With such high background light the difference between the on and off states of the transmitter becomes small. In those conditions the receiver might miss the preamble or have difficulty differentiating frequencies.
On the other hand, when transmitting at $32\text{bps}$ with $2\text{wpf}$, the reception has bigger problems. The distance and duty cycle have a bigger impact on the reception. This can be explained by the lack of synchronization. Since the implemented synchronization between the transmitter and receiver is not tight, the small errors in synchronization – that result from distance, duty cycle and background light – become a bigger issue.

The next step was to repeat the same experiments with UPWIS with OP101 photodiode attached to it. Figure 6.3 shows the results of those experiments. Those charts only include only 90/10 duty cycle because the result for 50/50 duty cycle is almost identical.

Those results show a big improvement in the reception in little background light conditions. This is because of the high sensitivity of the attached photodiode. On the other hand in the presence of high background light conditions, The reception completely fails at the distance of 4m. This is because the photodiode is saturated by the background light, since it does not have a big range. The drop at the distance of 3m is because the background light makes the photodiode close to saturation. This leaves small room for the difference between on and off states of the transmitter to be detected.

### 6.2 Security

The results presented in this section are for the proposed scenarios in Chapter 4 VLC for Security. The presented results are in the form of colored graphs. The graph represents the areas where it is possible to receive the key, the indications for the areas where light from a certain source is present are just for illustration, they are not actual measurements of the light intensity. The lights in those experiments were placed at 3m height, then measurements were taken at ground level and 1m offset from the ground level. We took measurements every 5cm with 15 attempts to get the key in each measurement. The duty cycle of the transmitter was kept at 90/10 to simulate a real-world use where no change in the light intensity is notable.

#### 6.2.1 Single Source Scenarios

The first scenario we evaluated was having a single light source transmit the key in a low background light environment. This scenario was straightforward as expected, the reception area bound by having sufficient light intensity to stimulate the photodiode enough for reception to happen. The second scenario was to introduce noise from
a different light source. This light source was a set of 4 fluorescent lights, the closest of which was 1.5m away from the transmitter. As expected, the reception area shrunk away from the fluorescent lights. Another observation was that the area away from the fluorescent lights also shrunk. This is because the overall background light of the environment increased making the reception harder. Figure 6.4 shows how the reception area shrinks in the previously mentioned setup. This data was collected using Tmote Sky at ground level. The transmitter source was 90° facing the ground.

6.2.2 Multiple Sources Scenarios

As mentioned in Chapter 4 there are plenty of possibilities to try out. We were interested in the scenarios where we use two light sources to transmit the key.

The first attempt was to place the lights at 90° angle facing the ground. The distance between the two lights was 3m. Figure 6.5 shows the area where the key can be received. The measurements were taken using UPWIS and Sky nodes. Each point of measure was repeated for ground level and 1m height.

In figure 6.5a the area looks like an intersection of both lights minus the areas where one of the lights is dominant. This is because the high intensity of the close light hides the changes in the overall intensity which the further light produces.

Figure 6.5b shows a reception area that looks different than what was measured with UPWIS. This is because the photodiode on board Sky is less sensitive than OPT101 – which is attached to UPWIS. On the other hand, it has a higher range, which means it is less prone to noise introduced by other sources. Hence, the reception areas is tighter away from the x axis center and wider towards it.

All four graphs suggest that changing the height of the receiver does not affect the shape of the area; rather its size. The higher the receiver is placed the smaller the area of reception becomes.

To get a better understanding of how the angle of the transmitter and the height and sensitivity of the receiver affect the reception, we go on with our previously proposed scenarios.

Figure 6.6 shows the results of tilting the light sources 45° toward each other. The rest of the experiment parameters were kept the same. The results of this experiment agree with the observations taken from the previous results. The shape of the area did
not change when varying the height; rather its size. The measured areas at 1m height were smaller of those at the ground level. Also Figure 6.6a shows that the reception area is the result of the intersection of both lights minus the areas where one of them is dominant for the same reasons mentioned earlier. And finally, Figure 6.6b shows that when using Sky, the area is tighter away from the x axis and wider towards it. This is also for the same previously mentioned reason.

An interesting result emerged when the last angle scenario was tested using UPWIS nodes. Figure 6.7 shows the results of the experiment where the transmitter lights were placed at 60° facing each other. The rest of the experiment parameters were maintained the same. When using 60°, the area of the reception when using UPWIS extended vastly to include all the intersection of the two lights. This is because the light intensity of the area where one of the lights is dominant is no longer high enough to hide the effects of the further source. Additionally, the reception was also possible for a relatively long distance behind the two lights. This is due to the large beam angle of the used bulb (38°). On the other hand, the results collected in this scenario from Sky were similar to those of 6.6b with the exception of the size of the area, which was smaller.

The last scenario to test was the effect of high background light environment. The results of this scenario were straightforward. UPWIS completely failed, since the attached OPT101 was saturated by the background light. The Sky platform managed to keep some transmission going, thanks to the higher range of its photodiode. The reception area shrunk similarly to what was illustrated in Figure 6.4.
Chapter 6. EVALUATION

(A) Key Reception Area with 2 Sources at 90° Using UPWIS.

(B) Key Reception Area with 2 Sources at 90° Using Sky.

Figure 6.5: Key Reception Area with 2 Sources at 90°.
Chapter 6. EVALUATION

(A) Key Reception Area with 2 Sources at 45° Using UPWIS.

(B) Key Reception Area with 2 Sources at 45° Using Sky.

**Figure 6.6:** Key Reception Area with 2 Sources at 45°.

**Figure 6.7:** Key Reception Area with 2 Sources at 60° Using UPWIS.
Chapter 7

RELATED WORK

There has been a lot of research going on in the field of data communication regarding Visible Light Communication. In this chapter we look into and summarize some of this published work.

- The IEEE 802.15.7 standard for short-range communication using visible light [1] specifies the Physical and MAC layers in a visible light communication stack. The protocol supports a variety of modulation schemes, as well as dimming through modifying pWM and flicker prevention. The standard defines data communication and management functions. It supports data rates varying from 11.67 kbps to 96 kbps.

- In OpenVLC: software-defined visible light embedded networks [16] the authors look into building an open-source VLC networking platform to be used by researchers. They go through the design and implementation of the MAC and Physical layers of the platform. They implement the system on a Linux networking platform and provide the required primitives. The Linux OS they implement their system on enabled them use traditional networking measurement tools for evaluation. They evaluate the performance of the system using ping and iperf. Their results show a ping Round-Trip Time of 420 ms. The maximum achievable throughput, measured by iperf, is 1.76 kbps. The average measured throughput for one-hop is 1.6 kbps and half of that for a two-hop scenario.

- In LED-to-LED Visible Light Communication Networks [13] the authors use LEDs for both transmission and reception. They build a system which enables low bit rate wireless networking over short distance. Their system is designed to be used in IoT applications where devices communicate over short distances. In their MAC layer, they use a CSMA/CA protocol for medium access controlling. Their physical layer consists of an MCU which controls an LED that is used for transmission and reception. They evaluate the performance of the system over a single link where they show the throughput over distance and the transmission delays for different payload sizes. They also evaluate the performance over a network scenario where they show the saturation throughput. Finally, they evaluate the power consumption over times.

- In Using Consumer LED Light Bulbs for Low-Cost Visible Light Communication [14] the authors describe a prototype where they integrate an MCU in a commercial light bulb and implement a software-based MAC and Physical layers. They use the LED light bulbs for transmission and reception in a way that is similar to their previous work [13]. They compare different commercial light bulbs and choose the most suitable one for their system then go through the steps of implementing their prototype. In the evaluation section they evaluate the
throughput over distance and varying payloads of the system when communicating between their bulbs and a VLC device and when communicating between light bulbs. Their results show a stable throughput until a distance of $2m$.

- In Hybrid Visible Light Communication for Cameras and Low-Power Embedded Devices [12] The authors look into building a hybrid link which can be used to communicate with mobile phone camera at low data rates and with a photodiode receiver at a higher rate. Their modulation scheme uses BFSK for communicating with mobile phone cameras at low rates while it uses BFSK and Manchester encoding for higher rates. Their approach to design this scheme is by using the high data rate to determine the frequency of the LED while the low data rate determines the duty cycle. In the evaluation section they evaluate the bit error rate (BER) to signal to noise ratio (SNR) for camera communications and the performance of both channels when changing the light intensity. Finally they conclude a trade-off in performance between both channels when changing the duty cycle in different light intensity levels.
Chapter 8

CONCLUSION AND FUTURE WORK

In this chapter we summarize and discuss the conclusions drawn from the work of this thesis. We also point out where there is room for improvements. And finally, propose improvements and future work.

8.1 VLC

After evaluating this part it is safe to conclude the following:

- **Data rate can be improved.** In this project the data rates were low. This was due to limitations introduced by the hardware we used. Sky platform faced problems computing the Goertzel algorithm fast enough, which limited its data rates. Although UPWIS has a much higher computation power, increase in data rate was only 4 times what it was with Sky. The photodiode we used had a low bandwidth which caused this limitation. Combining UPWIS platform with a photodiode which has a higher bandwidth is sufficient to increase the data transmission to an acceptable rate.

- **Robustness of the transmission can be improved.** After evaluating the performance of the two platforms, we see that a photodiode with a higher range – the photodiode on board Sky – performs better with high background light conditions, while it decays when extending the distance. A higher sensitivity photodiode can perform better when receiving at a longer distance, while it fails in the presence of a high background lighting conditions. Thus, using a photodiode with higher sensitivity and resolution can improve the robustness of the transmission over distance and noise introduced by the background light. However, this improvement is costly since those photodiodes are expensive.

The mentioned conclusions open up a gate for improvements and stimulate ideas for future work. The next step here is to further more study the relation between photodetectors, light sources and background light to infer a robust relation that can be used when designing VLC systems. Additionally, there is plenty of room for hardware improvements to create a system with suitable data rates for real-world deployment. Those hardware improvements can be:

- Better photodiodes for reception.
- A separate hardware chip to generate the PWM signal.
- A more compact and efficient solution for LED driving.
• Hardware gain controlling for a more efficient reception and better adjustment to
different conditions.

Improvements to software can be a better implementation of the Goertzel algorithm
to support detecting tones higher than quarter of the sampling rate, this is to allow
using higher frequencies without having to use an at least 4 times higher sampling rate
to detect it.

8.2 VLC for Security

The evaluation of this part introduced a set of interesting results which can be used to
conclude the following:

• Lots of related aspects. As mentioned earlier there are many aspects that play
a role in the shape and size of the area of reception. Those factors are namely
the beam angle, intensity and angle of the bulb, the sensitivity and range of the
receiver, the distance between the transmitters and between the transmitters and
receiver, background light intensity, lenses and the reflections over walls and sur-
rounding objects.

• Angle of the bulb and sensitivity and range of the receiver play a role in defin-
ing the contours of the reception area. The results of the experiment conducted
in this project provide evidence to say that the mentioned factors are at close re-
lation to how the area looks like, however the time and scope of this thesis does
not allow us to provide a concrete relation between those factors.

• The height of the receiver plays a role in defining the size of the area of recep-
tion. As seen in the results of the conducted experiment, the height of the receiver
does not change the shape of the area; rather the size of it. Basically the higher
you place the receiver, the smaller the area becomes. Again, we do not provide a
concrete relation due to the scope and time constraints.

• Background light can be used to shrink the area of reception. When attempting
to measure the reception area with a high background light intensity, the recep-
tion area did not only shrink away from the background light source, but also
shrunk for the other side. This suggests that it is possible to use background light
to further more limit the area of reception.

The results of our experimental scenarios show that there is much more to do than
simply place two lights and share a key. There are many aspects that define, interfere
with and modify the reception area. Those aspects open up a lot of room for future
work. A further more study of the relation between those aspects and building a con-
crete understanding of how they effect the security of this system is one of the most
important next steps of this project. Additionally, building a more complex key shar-
ning protocol that takes into account the relation to the received light intensity can be
a possible way to implement this system. This protocol can achieve the purpose of
securely sharing the data without worrying about all the aspects that can extend the
reception area for longer distances. Extending this work to study how having addi-
tional transmitters and looking into the relation between those transmitters affect the
reception area is an interesting step for future work. Also having a scenario where the
receiver should go through a certain path to receiver the key can be a possible way
of making use of using more light sources. And finally, porting this system to support reception with mobile phones can make this project eligible for real-world deployment.
Bibliography


