



**Technische Hochschule**  
**Augsburg** Technical University of  
Applied Sciences

**Bachelorarbeit**

**Fakultät für  
Informatik**

Studienrichtung  
Technische Informatik

Daniel Schwinghammer  
Matrikelnummer: 2043094

**Evaluierung eines Lokalisierungssystems für  
Bluetooth-Geräte in Innenräumen**

Erstprüfer: Prof. Dr. Hubert Högl  
Zweitprüfer: Prof. Dr. Gundolf Kiefer

Abgabe der Arbeit am: 20.11.2023

Technische Hochschule Augsburg  
Augsburg Technical University  
of Applied Sciences  
An der Hochschule 1  
D-86161 Augsburg  
Telefon +49 821 55 86-0  
Fax +49 821 55 86-3222  
[www.hs-augsburg.de](http://www.hs-augsburg.de)  
[info@hs-augsburg.de](mailto:info@hs-augsburg.de)  
Fakultät für Informatik  
Telefon +49 821 5586-3450  
Fax +49 821 5586-3499  
Verfasser der Bachelorarbeit:  
Daniel Schwinghammer  
Neuschwansteinstraße 10  
86163 Augsburg  
Telefon +49 176 66551806  
[daniel.schwinghammer@hs-augsburg.de](mailto:daniel.schwinghammer@hs-augsburg.de)

© 2023 Daniel Schwinghammer

Diese Arbeit mit dem Titel

*»Evaluierung eines Lokalisierungssystems für Bluetooth-Geräte in  
Innenräumen«*

von Daniel Schwinghammer steht unter einer

*Creative Commons Lizenz vom Typ Namensnennung - Weitergabe unter gleichen  
Bedingungen 4.0 International Lizenz (CC BY-SA).*

<https://creativecommons.org/licenses/by-sa/4.0/legalcode.de>



Sämtliche, in der Arbeit beschriebene Ergebnisse in Form von Quelltexten und  
Software stehen unter einer GNU General Public License Version 3.

<https://www.gnu.de/documents/gpl.de.html>

## **Danksagung**

Bevor der eigentliche Inhalt meiner Bachelorarbeit beginnt, möchte ich mich an dieser Stelle zuerst bei meinen Eltern bedanken. Ohne die Unterstützung dieser, wäre ich vermutlich heute nicht an dem Punkt, an dem ich heute stehe und diese Arbeit würde so nicht existieren. Als nächstes möchte ich mich bei meiner Freundin bedanken, welche mich im gesamten Bearbeitungszeitraum dieser Arbeit immer wieder moralisch unterstützt hat. Ebenfalls gilt mein Dank auch den Korrekturlesern. Euer Feedback hat mir sehr geholfen, dass die Arbeit in dem Zustand wie sie ist, so entstehen konnte. Zuletzt möchte ich mich bei meinem Professor Dr. Hubert Högl bedanken, der mich stetig bei der Bearbeitung der Bachelorarbeit begleitet und wichtige Bearbeitungshinweise gegeben hat.

## Abkürzungsverzeichnis

<b>GPS</b>	Global Positioning System . . . . .	1
<b>Bluetooth BR</b>	Bluetooth Basic Rate . . . . .	3
<b>Bluetooth EDR</b>	Bluetooth Enhanced Data Rate . . . . .	3
<b>ISM</b>	Industrie, Wissenschaft und Medizin . . . . .	3
<b>Bluetooth LE</b>	Bluetooth Low Energy . . . . .	4
<b>RSSI</b>	Received Signal Strength Indicator . . . . .	4
<b>Bluetooth SIG</b>	Bluetooth Special Interest Group . . . . .	7
<b>AoA</b>	Angle of Arrival . . . . .	7
<b>AoD</b>	Angle of Departure . . . . .	8
<b>IQ Abtastung</b>	Inphase- und Quadraturabtastung . . . . .	15

## Abbildungsverzeichnis

1	Lokalisierung mit einem Ankerpunkt . . . . .	5
2	Lokalisierung mit mehreren Ankerpunkten . . . . .	6
3	AoA-Methode - Antennen-Array im Empfänger . . . . .	7
4	AoD-Methode - Antennen-Array im Sender . . . . .	8
5	Darstellung eines Wellenzyklus . . . . .	9
6	Darstellung der Wellenlänge . . . . .	10
7	Darstellung einer Frequenz . . . . .	11
8	Bedeutung der Phase in einem Wellenzyklus . . . . .	12
9	Gleiche Phasenwerte bei gleichem Abstand zum Sender . . . . .	13
10	Ungleiche Phasenwerte bei unterschiedlichen Entfernungen zum Sender . . . . .	13
11	Verwendung der Phasendifferenz zur Berechnung des Ankunfts- winkels . . . . .	14
12	Verwendung der Phasendifferenz zur Berechnung des Abflugwin- kels . . . . .	15
13	Phasenwinkel und Amplitude als (I,Q) kartesische Koordinaten . .	15
14	Multi-Antennen-Anker (C211) . . . . .	17
15	Tag (C209) . . . . .	18
16	Anzeige der seriellen Schnittstelle im Gerätemanager . . . . .	18
17	Datenübertragung des Ankers über die serielle Schnittstelle . . . .	20
18	Winkelberechnung ausgehend vom Anker . . . . .	21
19	Darstellung der Daten mit s-center . . . . .	22
20	Person oder Objekt steht statisch . . . . .	23
21	Messung in der linken oberen Ecke . . . . .	24
22	Messung in der linken unteren Ecke . . . . .	25
23	Person oder Objekt bewegt sich im Raum . . . . .	26
24	Position A im Raum . . . . .	27
25	Position B im Raum . . . . .	27
26	Position C im Raum . . . . .	28
27	Zuordnung der Seite . . . . .	29

28	Seitenzuordnung Links . . . . .	30
29	Seitenzuordnung Rechts . . . . .	30
30	Maximale Reichweite des Systems . . . . .	31
31	Gemessene RSSI Werte . . . . .	31
32	Mittelwerte in Bezug auf die Distanz . . . . .	32
33	Visualisierung der Winkel zur Laufzeit . . . . .	35

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>ii</b>
<b>Inhaltsverzeichnis</b>	<b>iv</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Ziel der Arbeit . . . . .	1
1.3 Aufbau der Arbeit . . . . .	2
<b>2 Grundlagen</b>	<b>3</b>
2.1 Bluetooth Klassisch . . . . .	3
2.2 Bluetooth Niedrigenergie . . . . .	3
2.3 Bluetooth Beacon . . . . .	4
2.4 Bluetooth Tag . . . . .	4
2.5 Received Signal Strength Indicator . . . . .	4
<b>3 Lokalisierung via Bluetooth vor Version 5.1</b>	<b>5</b>
<b>4 Bluetooth-Richtungsbestimmung</b>	<b>7</b>
4.1 Angle of Arrival . . . . .	7
4.2 Angle of Departure . . . . .	8
4.3 Theorie der Bluetooth-Richtungsbestimmung . . . . .	9
<b>5 XPLR-AOA-1 kit</b>	<b>17</b>
5.1 Anker-Setup . . . . .	18
5.2 Tag-Setup . . . . .	19
5.3 Anwendung . . . . .	20

<b>6</b>	<b>Evaluierung des Systems</b>	<b>23</b>
6.1	Objekt oder Person steht statisch im Raum . . . . .	23
6.2	Objekt oder Person bewegt sich im Raum . . . . .	26
6.3	Seitenzuordnung einer Person oder eines Objekts . . . . .	29
6.4	Maximale Reichweite des Systems . . . . .	31
<b>7</b>	<b>Selbst entwickelte GUI</b>	<b>33</b>
7.1	Aufbau des Programms . . . . .	33
7.2	Anwendungsbeispiel . . . . .	35
<b>8</b>	<b>Fazit und Ausblick</b>	<b>36</b>
	<b>Literatur</b>	<b>37</b>



# 1 Einleitung

## 1.1 Motivation

Die fortschreitende Digitalisierung und Vernetzung von Geräten haben bereits heute einen tiefgreifenden Einfluss auf die Art und Weise, wie wir uns in Innenräumen bewegen und interagieren. Während sich traditionelle Lokalisierungssysteme, die auf dem Global Positioning System (GPS) basieren, als äußerst nützlich für die Ortung im Freien erwiesen haben, stößt diese Technologie in Innenräumen auf verschiedene Herausforderungen, die seine Wirksamkeit stark einschränken. Beispiele hierfür sind unter anderem eine schlechte Signaldurchdringung oder Mehrfachreflexionen. Diese Einschränkungen haben zu verstärktem Interesse an alternativen Ansätzen geführt, insbesondere im Kontext von Bluetooth-gesteuerten Lokalisierungssystemen. Die präzise Bestimmung eines Standortes innerhalb von Gebäuden bietet dabei vielfältige Anwendungsmöglichkeiten, die von der Innenraum-Navigation in einem Einkaufszentrum bis zur Verfolgung des Materialflusses in Fabriken oder zur Messung der Spielerperformance in einem Sportteam reichen. Die mit Bluetooth 5.1 eingeführte Technologie **Bluetooth-Richtungsbestimmung** (im technischen genannt Bluetooth Direction Finding) ermöglicht Entwicklern, bessere Lösungen im Bereich der Innenraumlokalisierung zu entwickeln. Die vorliegende Bachelorarbeit widmet sich der Evaluierung eines Lokalisierungssystems, welches auf der eben genannten Technologie basiert.

## 1.2 Ziel der Arbeit

Das Ziel dieser Arbeit besteht darin, das Lokalisierungssystem *XPLR-AOA-1* der Firma ublox, in verschiedenen Anwendungsszenarien zu analysieren und bewerten. Die Schlüsselfaktoren, welche sich genauer angeschaut werden, sind die Leistungsfähigkeit und Genauigkeit des Systems in den unterschiedlichen Anwendungsfällen.

### **1.3 Aufbau der Arbeit**

Das folgende Kapitel 2 *Grundlagen* beschreibt in Grundzügen Bluetooth Klassisch und Bluetooth Niedrigenergie. Darüber hinaus wird der Unterschied zwischen einem Bluetooth Beacon und einem Bluetooth Tag erläutert. Zuletzt wird in diesem Kapitel auf einen wichtigen Indikator, in Bezug auf kabellose Kommunikationsanwendungen, eingegangen.

Anschließend wird in Kapitel 3 *Lokalisierung via Bluetooth vor Version 5.1* beschrieben, wie die Lokalisierung mit Bluetooth realisiert worden ist, bevor die Funktion der Bluetooth-Richtungsbestimmung in der Version Bluetooth 5.1 hinzugekommen ist.

Das Kapitel 4 *Bluetooth-Richtungsbestimmung* beschreibt die beiden Architekturen, mit der die Bluetooth-Richtungsbestimmung realisiert wird. Ebenfalls werden an dieser Stelle die technischen Hintergründe betrachtet.

In Kapitel 5 *XPLR-AOA-1 kit* wird die Inbetriebnahme des Lokalisierungssystems, welches in dieser Arbeit evaluiert wird, genauer beschrieben.

In dem Kapitel 6 *Evaluierung des Systems* wird das System in unterschiedlichen Anwendungsszenarien getestet und bewertet.

Kapitel 7 *Selbst entwickelte GUI* beschreibt den Hintergrund für die Entwicklung dieses Programm und erklärt dessen Aufbau.

Kapitel 8 *Fazit und Ausblick* schließt mit einer persönlich Einschätzung über die gewonnen Erkenntnisse über die Bluetooth-Richtungsbestimmung Technologie und dem Lokalisierungssystem ab.

## **2 Grundlagen**

### **2.1 Bluetooth Klassisch**

Das Funkmodul Bluetooth Klassisch, auch Bluetooth Basic Rate (Bluetooth BR) oder Bluetooth Enhanced Data Rate (Bluetooth EDR) genannt, ist ein Modul, welches Daten in dem nicht lizenzierten 2,4 GHz-Frequenzband für Industrie, Wissenschaft und Medizin (ISM) über 79 Kanäle überträgt. Die Kanäle haben dabei einen Abstand von 1 MHz. Die Datenübertragung kann sowohl asynchron als auch synchron Verbindungsorientiert sein. Bluetooth Klassisch unterstützt dabei die Punkt-zu-Punkt-Kommunikation von Geräten. Hauptsächlich wird Bluetooth Klassisch für die kabellose Datenübertragung, wie beispielsweise für das Audiostreaming verwendet. [2]

### **2.2 Bluetooth Niedrigenergie**

Bluetooth Niedrigenergie (auch genannt Bluetooth Low Energy) ist eine drahtlose Kommunikationstechnologie, die entwickelt worden ist, um eine energiesparende Verbindung zwischen Geräten herzustellen. Es ist eine Variante des herkömmlichen Bluetooth-Protokolls und wurde speziell für Anwendungen mit geringem Energieverbrauch und Datenübertragungsraten optimiert. Ursprünglich war die Funktechnik für eine energiesparende Vernetzung zwischen zwei Geräten bekannt. So wird sie heutzutage aber auch als Technologie zur Lokalisierung eingesetzt, um die immer größer werdende Nachfrage nach hochpräzisen Lokalisierungssystemen in Innenräumen zu befriedigen. Stand heute umfasst Bluetooth Niedrigenergie Funktionen, mit denen es einem Gerät ermöglicht wird, die Anwesenheit, Richtung und Entfernung eines anderen Gerätes zu bestimmen. [6] [2]

## **2.3 Bluetooth Beacon**

Ein Bluetooth Beacon ist ein kleines, oft fest installiertes drahtloses Gerät, das Informationen an andere Bluetooth-fähige Geräte in seiner Nähe sendet. Der Beacon verwendet die Bluetooth Low Energy (Bluetooth LE) Technologie, um kurze, regelmäßige Signale auszusenden, die von anderen Geräten wie zum Beispiel einem Smartphone oder Tablet, empfangen werden können. Die Signale enthalten oftmals neben den eigentlichen Daten auch eine Kennung, um den Beacon identifizieren zu können. Das Einsatzgebiet eines Beacons ist unterschiedlich, oft werden sie aber im Bereich von standortbasierten Diensten genutzt. Anzumerken ist, dass die meisten Beacons passiv sind. Sie senden also nur Signale und empfangen keine Daten. Die Kommunikation erfolgt daher in der Regel nur in eine Richtung, von einem Beacon zu einem empfangendem Gerät. [9]

## **2.4 Bluetooth Tag**

Ein Bluetooth Tag ist ein kleines, tragbares elektronisches Gerät, das ebenfalls wie ein Bluetooth Beacon die Bluetooth LE Technologie verwendet, um in regelmäßigen Abständen mit anderen Bluetooth-fähigen Geräten zu kommunizieren. Die Tags dienen unter anderem zur Lokalisierung des Standorts von Gegenständen oder Personen. [9]

## **2.5 Received Signal Strength Indicator**

Der Received Signal Strength Indicator (RSSI) ist für einen Empfänger in kabellosen Kommunikationsanwendungen ein Indikator für die Signalstärke. Zu beachten ist, dass der Indikator eine Verhältniszahl ist. Somit muss der Wert abhängig von der jeweiligen Anwendung interpretiert werden. [8]

### 3 Lokalisierung via Bluetooth vor Version 5.1

Bereits vor der Veröffentlichung von Bluetooth 5.1 hatte sich die Technologie Bluetooth LE als führende Lösung für Lokalisierungsanwendungen in Innenräumen etabliert. Der zugrunde liegende Ansatz bestand darin, den RSSI des Signals von einem Beacon mit einem Bluetooth-Empfänger zu messen. Alternativ könnten feste Ankerpunkte den RSSI von einem sich bewegendem Bluetooth-Sender messen. Die beiden Methoden basierend auf dem RSSI können dabei eine Genauigkeit von einigen Metern erreichen. Die Abbildungen 1 und 2 zeigen, wie Bluetooth LE vor der Veröffentlichung von Bluetooth 5.1 für die Lokalisierung in Innenräumen genutzt wurde. [4]

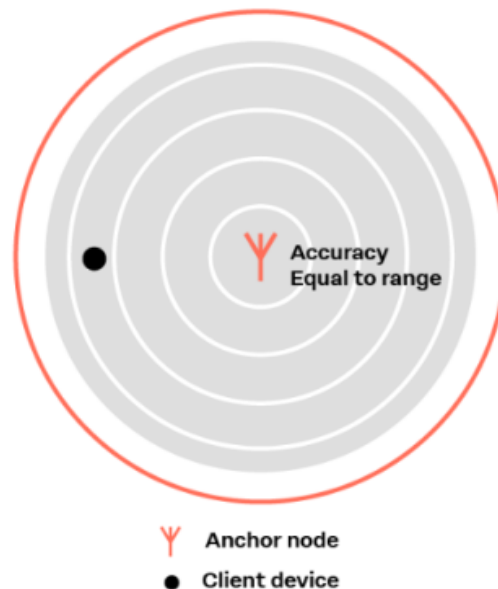


Abbildung 1: Lokalisierung mit einem Ankerpunkt<sup>1</sup>

In Abbildung 1 betritt beispielsweise eine Person mit einem Bluetooth-fähigen Gerät einen Raum, in dem sich ein Ankerpunkt befindet. Das mobile Geräte, beispielsweise ein Smartphone, sendet dann in kurzen Zeitabständen immer wieder ein Signal, welches von dem Ankerpunkt im Raum empfangen wird. Der Anker kann im Folgenden aufgrund des RSSI eine grobe Lokalisierung des Senders tätigen. Dieser Ansatz wurde zur Umsetzung von Lösungen verwendet, die erkennen, ob Personen ein Zielgebiet, zum Beispiel in einem Einkaufszentrum, betreten haben. [4]

<sup>1</sup><https://www.u-blox.com/en/technologies/bluetooth-indoor-positioning>

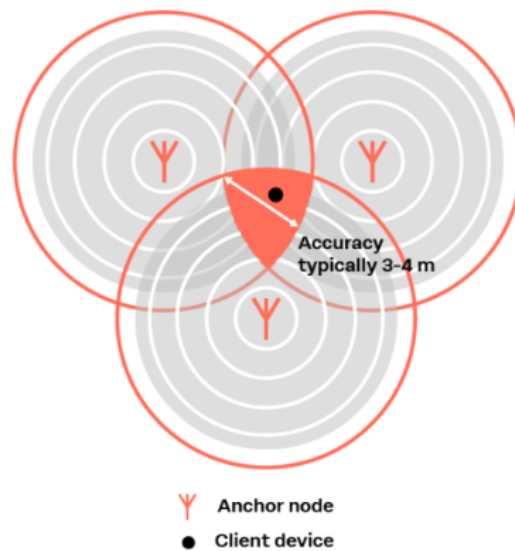


Abbildung 2: Lokalisierung mit mehreren Ankerpunkten<sup>2</sup>

Wenn wie in Abbildung 2 gezeigt der Raum mit mehreren Ankerpunkten ausgestattet ist, können komplexe Algorithmen verwendet werden, um die Position einer Person im Verhältnis zu den Ankern genauer zu bestimmen. Beide Ansätze nutzen den RSSI als Basis für die Lokalisierung. In dem Fall mit nur einem Anker, kann nur eine begrenzte Aussage über die Lokalisierung der Person getätigt werden. Stehen mehrere Ankerpunkte zur Verfügung, so ist die Lokalisierungs Genauigkeit schon deutlich erhöht. [4]

<sup>2</sup><https://www.u-blox.com/en/technologies/bluetooth-indoor-positioning>

## 4 Bluetooth-Richtungsbestimmung

In Anbetracht der wachsenden Nachfrage nach präzisen Lösungen für die Lokalisierung in Innenräumen hat die Bluetooth Special Interest Group (Bluetooth SIG) im Januar 2019 mit der Version Bluetooth 5.1 einen Weg geschaffen, bessere Lösungen für die Innenraumortung zu realisieren. Die Technologie Bluetooth-Richtungsbestimmung basiert darauf, dass die Richtung des Funksignals von einem Sender mit Hilfe eines Multi-Antennen-Ankers bestimmt werden kann. Werden mehrere dieser Anker verwendet, so ist die Entwicklung von Positionsbestimmungssystemen mit einer Genauigkeit im Zentimeterbereich möglich. [4] [1] In den folgenden beiden Abschnitten werden die technischen Hintergründe der Technologie Bluetooth-Richtungsbestimmung genauer erläutert.

### 4.1 Angle of Arrival

Die Bluetooth-Richtungsbestimmung kennt zwei unterschiedliche Architekturen oder Methoden, die beide auf der gleichen Basis beruhen. Bei beiden Methoden wird ein spezielles Signal von einem Gerät gesendet. Dieses Signal wird dann bei dem anderen Gerät genutzt, um die Richtung, aus der das empfangene Signal gekommen ist, zu berechnen. Die erste Methode trägt den Namen Angle of Arrival (AoA). Im Falle von AoA besitzt der Empfänger, wie in Abbildung 3 zu sehen ist, ein Multi-Antennen-Array.

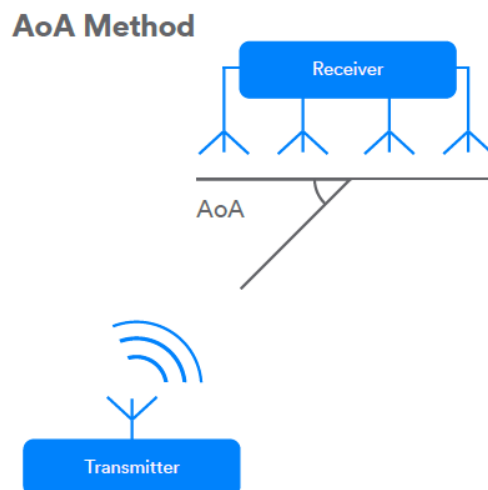


Abbildung 3: AoA-Methode - Antennen-Array im Empfänger<sup>3</sup>

<sup>3</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

Bei AoA werden die Messungen des Antennen-Arrays des Empfängers genutzt, um die Richtung des eingehenden Signals mit Hilfe eines Algorithmus, der lokal im Empfänger abläuft, zu berechnen. [10] [3]

## 4.2 Angle of Departure

Im Falle von Angle of Departure (AoD) besitzt der Sender, wie Abbildung 4 zeigt, ein Multi-Antennen-Array. In diesem Fall empfängt das mobile Gerät spezielle Signale von einem oder mehreren Multi-Antennen-Arrays. Bei AoD versucht das mobile Gerät die Richtung des eingehenden Signals oder der Signale für den Fall, dass es mehrere Antennen-Arrays gibt, zu berechnen. Bei der Verwendung von AoD erreichen die Signale, die von jedem Element der Antennengruppe eines Ankers ausgesendet werden, das mobile Gerät mit einer leichten Phasenabweichung im Vergleich zu den übrigen Elementen. Sind dem mobilen Gerät Informationen über die Geometrie der Antennengruppen bekannt, so kann die Richtung des Signals von der Antennengruppe anhand der gemessenen Phasenabweichung berechnet werden. [10] [3]

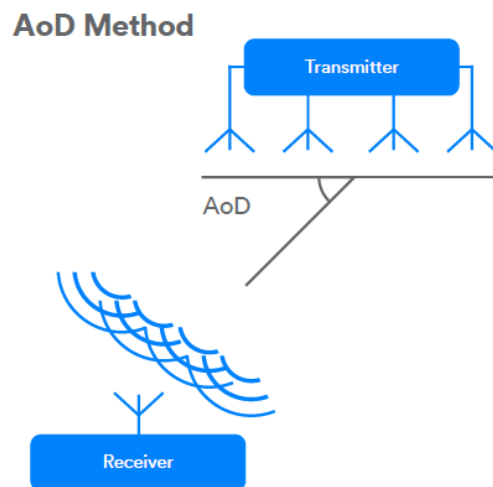


Abbildung 4: AoD-Methode - Antennen-Array im Sender<sup>4</sup>

<sup>4</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)



### 4.3 Theorie der Bluetooth-Richtungsbestimmung

Die Technologie Bluetooth-Richtungsbestimmung nutzt einige der fundamentalen Eigenschaften von Funkwellen, um Daten für die Richtungsbestimmung zu erlangen. Anwendungen können dann die gewonnenen Informationen in ihre internen Berechnung durch Zunahme von Trigonometrie und Wissen über das Design des Antennen-Arrays für die Richtungsbestimmung nutzen. Es folgt eine Erklärung der Eigenschaften von Funkwellen, die im Bezug zur Richtungsbestimmung via Bluetooth wichtig sind. [10]

#### Wellenzyklus:

Wellen folgen einem sich wiederholenden Muster mit Spitzen und Tälern mit maximaler und minimaler Amplitude. Jede Wiederholung der Welle auf ihrem Weg von einer Amplitude von Null zu ihrem Höchststand über die Amplitude null zu ihrem Tiefpunkt und wieder zurück zu null wird als Wellenzyklus bezeichnet. In Abbildung 5 ist das Konzept eines Wellenzyklus illustriert. [10]

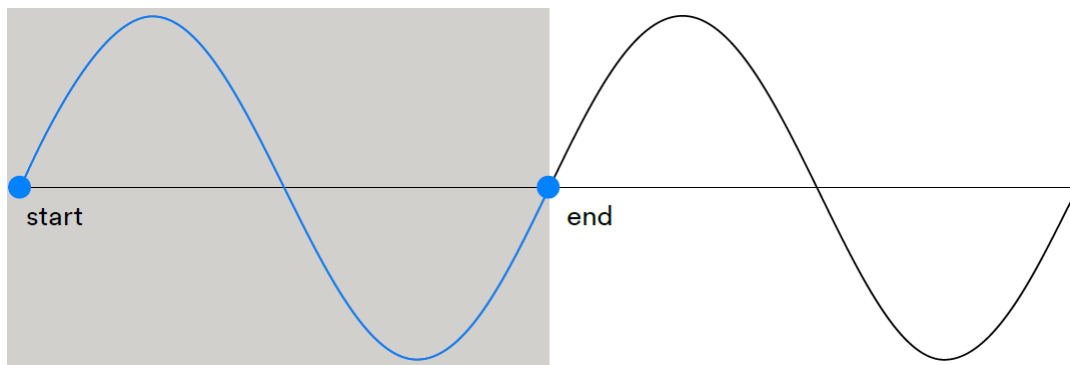


Abbildung 5: Darstellung eines Wellenzyklus<sup>5</sup>

<sup>5</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

**Wellenlänge:**

Eine Wellenlänge (auch genannt Periodendauer) misst den Abstand zwischen dem Beginn und Ende eines vollständigen Wellenzyklus. Die Periodendauer lässt sich, wie in Formel (1) beschrieben, berechnen. In Abbildung 6 ist eine Wellenlänge dargestellt. [10]

$$T = \frac{1}{f} \quad (1)$$

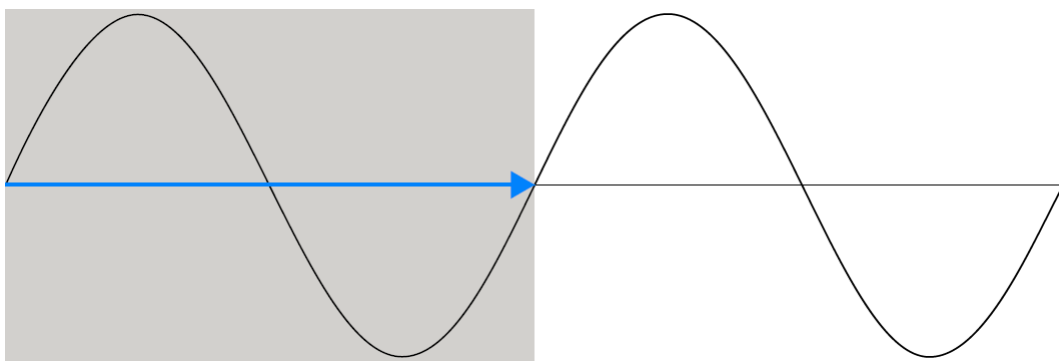


Abbildung 6: Darstellung der Wellenlänge<sup>6</sup>

<sup>6</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

### Frequenz:

Bluetooth arbeitet, wie in Kapitel 2 schon beschrieben, im ISM Band von 2,40 GHz bis 2,41 GHz. Bluetooth LE unterteilt dieses Band in 40 Kanäle mit einer Kanalbreite von jeweils 2 MHz. Wenn Geräte in einer Verbindung sind, nutzen diese 37 von den 40 vorhandenen Kanälen mit Frequenzänderungen, die durch einen dynamischen Frequenzsprung-Algorithmus gesteuert werden. In verbindungslosen Anwendungsszenarien werden bei Verwendung der erweiterten Nachrichtenverbreitung (technisch auch advertising genannt), die in der Bluetooth Kernspezifikation v5.0 eingeführt wurde, alle 40 Funkkanäle verwendet. Hieraus folgt, dass die Kommunikation via Bluetooth nicht nur eine, sondern verschiedene Frequenzen nutzt. Diese Information ist wichtig, denn die Wellenlänge, welche für die Bluetooth-Richtungsbestimmung von großer Bedeutung ist, variiert also je nach verwendeter Frequenz. Abbildung 7 zeigt das Konzept von Frequenzen auf. [10] [11]

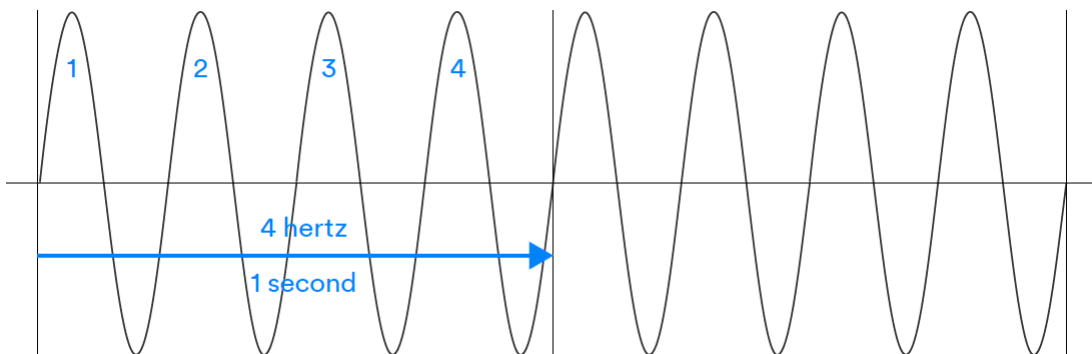


Abbildung 7: Darstellung einer Frequenz<sup>7</sup>

<sup>7</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

**Phase:**

Einen beliebigen Punkt in einem Wellenzyklus, der beispielsweise beim Durchgang der Welle über eine Antenne gemessen wird, bezeichnet man als Phase. Die Phase wird dabei als Winkel zwischen  $0^\circ$  am Anfang des Wellenzyklus und  $360^\circ$  oder  $2\pi$  Radiant am Ende des Wellenzyklus gemessen. Abbildung 8 veranschaulicht das Konzept der Phase. [10]

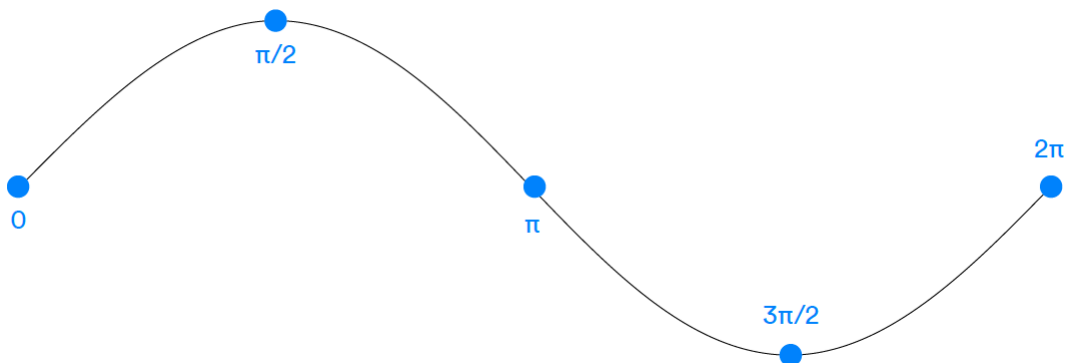


Abbildung 8: Bedeutung der Phase in einem Wellenzyklus<sup>8</sup>

**Verwendung der Phase zur Bestimmung der Signalrichtung:**

Wenn ein Sender ein Signal aussendet, so breitet es sich mit Lichtgeschwindigkeit in drei Dimensionen vom Sender aus. Dies gilt aber nur unter der Annahme, dass das Signal weder von Hindernissen noch von anderen Faktoren behindert wird. Ihre Bahn folgt einer sich ausdehnenden Kugel, und die Wellenfront auf der Oberfläche dieser Kugel hat eine Amplitude, die stetig abnimmt, wenn die in der Übertragung enthaltene Energie über eine immer größere Fläche verteilt wird. Während sich die Kugel ausdehnt, entfernt sie sich immer weiter vom Sender. Um diese Idee zu verallgemeinern, ist es einfacher, in zwei statt in drei Dimensionen zu denken. Dabei beschreibt das Signal eine kreisförmige Bahn, ähnlich wie die Wellen, die in einem Wasserbecken entstehen, wenn ein Stein hineingeworfen wird. Betrachtet man eine Antenne, welche man in den Signalweg platziert, so ändert sich die Phase der Welle beim Passieren der Antenne kontinuierlich von  $0^\circ$  bis  $360^\circ$ . Wird die Phase dann zu einem bestimmten Zeitpunkt ( $t$ ) gemessen, so hat diese einen Wert, der als  $p_1$  bezeichnet wird. Wenn eine weitere Antenne in den Signalweg an beliebiger Stelle auf den Umfang des Kreises gestellt wird, so haben beide Antennen den gleichen Abstand zum Sender. Unter der Voraussetzung, dass die Welle beide Antennen mit der gleichen Frequenz und damit der gleichen Wellenlänge passiert, so ist die Phase  $p_2$  bei der zweiten Antenne gleich

<sup>8</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

der Phase  $p_1$  der ersten, wenn diese zum selben Zeitpunkt ( $t$ ) gemessen wird. Dieser Zusammenhang ist in Abbildung 9 veranschaulicht. [10]

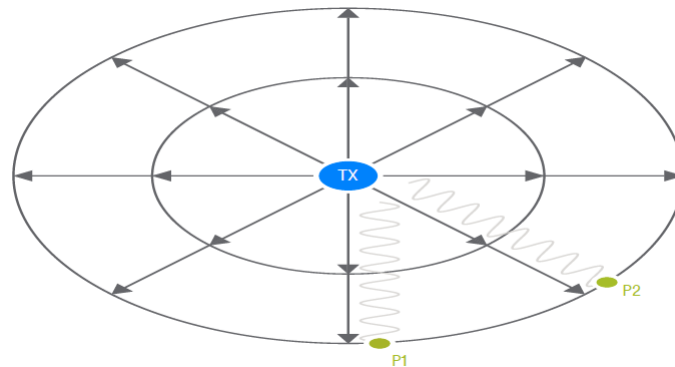


Abbildung 9: Gleiche Phasenwerte bei gleichem Abstand zum Sender<sup>9</sup>

Wenn nun die zweite Antenne so verschoben wird, dass sie sich näher am Sender befindet, wobei darauf geachtet werden muss, dass die Differenz zwischen dem Abstand von Antenne 1 zum Sender und der Abstand von Antenne 2 zum Sender kein exaktes Vielfaches der Wellenlänge ist, dann erhält man bei der Messung von  $p_1$  und  $p_2$  zum Zeitpunkt ( $t$ ) unterschiedliche Phasenwerte. Der Zusammenhang wird in Abbildung 10 nochmal gezeigt. [10]

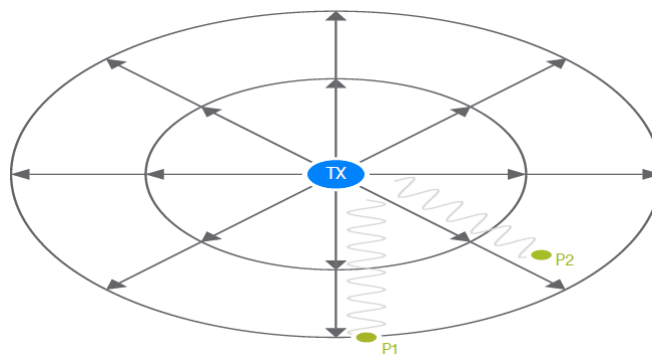


Abbildung 10: Ungleiche Phasenwerte bei unterschiedlichen Entfernungen zum Sender<sup>11</sup>

<sup>9</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

<sup>11</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

Wenn der Abstand zwischen den beiden Antennen (in einer geraden Linie), die Phasendifferenz ( $p_2 - p_1$ ) und die Wellenlänge des Signals bekannt sind, kann mit einfacher Trigonometrie der Richtungswinkel des Signals für die AoA-Methode wie in Abbildung 11 gezeigt oder für die AoD-Methode wie in Abbildung 12 dargestellt, berechnet werden. [10]

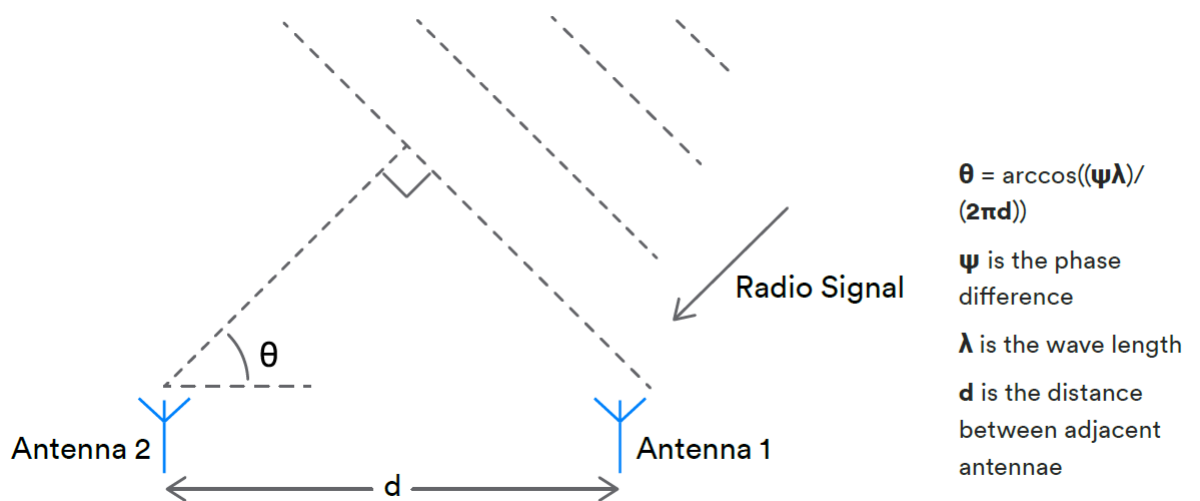


Abbildung 11: Verwendung der Phasendifferenz zur Berechnung des Ankunfts-winkels<sup>13</sup>

<sup>13</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

<sup>15</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

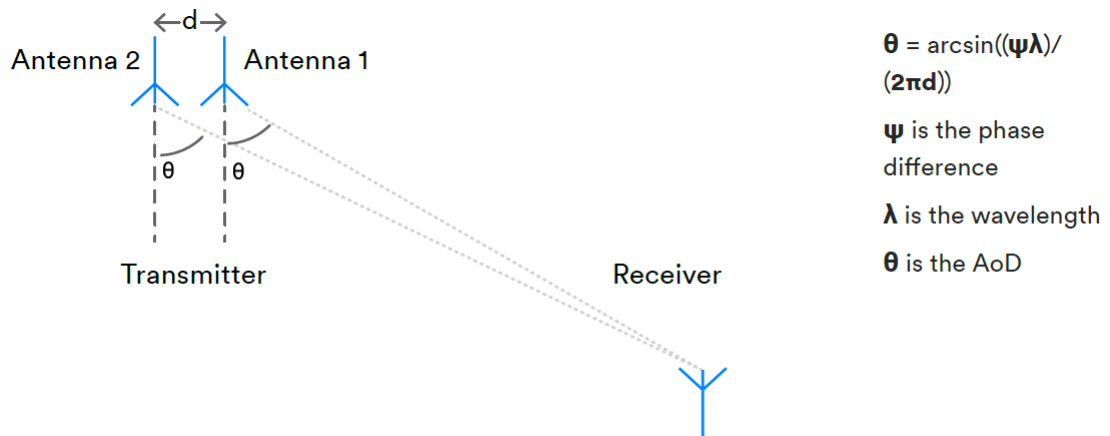


Abbildung 12: Verwendung der Phasendifferenz zur Berechnung des Abflugwinkels<sup>15</sup>

#### Abtastung:

Die Bluetooth-Richtungsbestimmung benutzt speziell entwickelte Richtungssignale. Ein Gerät, das eines dieser Signale empfängt, nimmt eine Reihe von Phasen- und Amplitudenmessungen in präzisen Intervallen in einem Prozess vor, der als Inphase- und Quadraturabtastung (IQ Abtastung) bekannt ist. Eine einzelne IQ Abtastung besteht aus der Amplitude und dem Phasenwinkel der Welle, dargestellt als ein Paar von kartesischen Koordinaten (siehe Abbildung 13).

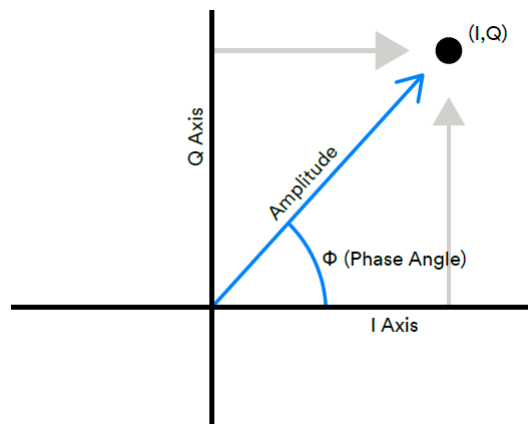


Abbildung 13: Phasenwinkel und Amplitude als (I,Q) kartesische Koordinaten<sup>16</sup>

<sup>16</sup>[https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf)

Anwendungen können diese kartesische Darstellung in entsprechende Polarkoordinaten umwandeln, die dem Phasenwinkel und dem Amplitudenwert entspricht. Bei der Durchführung von IQ Abtastung in einem Gerät mit einer Antennengruppe muss jede der erfassten Abtastungen einer bestimmten Antenne in der Gruppe zugewiesen werden. Im Falle von AoA enthält der Empfänger die Antennengruppe und führt eine IQ Abtastung von jeder Antenne in einer geeigneten Reihenfolge durch. Im Fall von AoD ist es der Sender, der die Antennengruppe enthält, aber es ist immer noch der Empfänger, der die IQ Abtastung durchführt. Dies geschieht, indem er Messungen von seiner einzelnen Antenne vornimmt und Details über das Design der Antennengruppe im Sender verwendet, um die Messungen bestimmter Antennen in der Gruppe des Senders und in Richtungsberechnungen zuzuordnen. Damit AoD funktionieren kann, muss es also einen Weg geben, dem Empfänger die Details der Antennengruppe des Senders mitzuteilen. Die Bluetooth SIG hat dafür Profile veröffentlicht, die festlegen, wie dies zu bewerkstelligen ist. [10]



## 5 XPLR-AOA-1 kit

Nachdem in Kapitel 4 der theoretische Hintergrund für die Richtungsbestimmung mit der Bluetooth-Technologie erläutert worden ist, befasst sich dieses Kapitel mit dem Lokalisierungssystem, das in dieser Arbeit evaluiert wird. Das System ist das *XPLR-AOA-1* der Firma ublox. Es basiert auf der AoA-Methode, die in Kapitel 4 beschrieben ist. Das System besteht hardwareseitig aus einem Multi-Antennen-Anker (auch genannt C211) und einem Tag (auch genannt C209). Der Anker enthält das Bluetooth LE-Modul NINA-B411 und der Tag das Bluetooth LE-Modul NINA-B406. Ebenfalls wird von der Firma Software zur Verfügung gestellt, um die Funktionsweise der AoA-Methode, auf dem das System basiert, zu testen. Der grundlegende Aufbau des Systems ist der, dass der Tag ein spezielles Richtungssignal periodisch aussendet und dieses von dem Multi-Antennen-Anker empfangen wird. Die Firmware, die auf dem eingebetteten Mikrocontroller in NINA-B411 läuft, berechnet dann den Ankunftsinkel des empfangenen Signals. Mit Hilfe des Programms *s-center*, das von ublox bereitgestellt wird, können die berechneten Winkel des Ankers in einer Graphikanzeige visuell betrachtet werden. Die graphische Darstellung repräsentiert den Ankunftsinkel des ankommenden Signals zu einem Bezugspunkt, wie in diesem Fall auf den Anker. Der Multi-Antennen-Anker und der Tag sind in den Abbildungen 14 und 15 zu sehen [5]. Im Folgenden geht es darum, die Firmware auf den Anker und den Tag zu installieren.

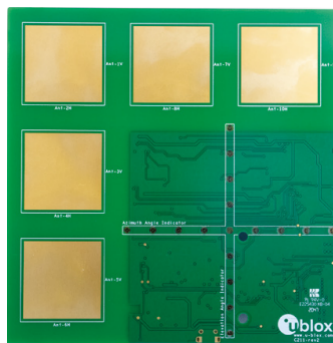


Abbildung 14: Multi-Antennen-Anker (C211)<sup>17</sup>

<sup>17</sup>[https://content.u-blox.com/sites/default/files/XPLR-AOA-1\\_ProductSummary\\_UBX-21015378.pdf](https://content.u-blox.com/sites/default/files/XPLR-AOA-1_ProductSummary_UBX-21015378.pdf)

<sup>18</sup>[https://content.u-blox.com/sites/default/files/XPLR-AOA-1\\_ProductSummary\\_UBX-21015378.pdf](https://content.u-blox.com/sites/default/files/XPLR-AOA-1_ProductSummary_UBX-21015378.pdf)



Abbildung 15: Tag (C209)<sup>18</sup>

## 5.1 Anker-Setup

Der Anker sowie der Tag werden abgesehen vom Bootloader ohne Firmware ausgeliefert. In den kommenden Schritten wird gezeigt, wie diese installiert wird.

Zuerst wird der Anker mit einem USB-A- auf Micro-B-Kabel an den Computer angeschlossen. Die Treiber für den eingebauten FTDI USB-zu-UART-Wandler werden normalerweise automatisch installiert. Andernfalls können diese direkt von der FTDI-Website (<https://ftdichip.com/drivers/>) heruntergeladen werden. Die serielle Schnittstelle ist nach erfolgreicher Installation des Treibers zum Beispiel auf Windows im Gerätemanager sichtbar (siehe Abbildung 16).

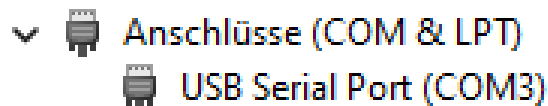


Abbildung 16: Anzeige der seriellen Schnittstelle im Gerätemanager<sup>19</sup>

Im nächsten Schritt wird die Firmware namens *u-connectLocate* von der ublox-Website (<https://www.u-blox.com/en/product/s-center>) heruntergeladen. Die Firmware kann jetzt auf zweierlei Arten installiert werden. Zum einen kann das Tool namens *newtmgr* benutzt werden, welches für die Plattformen Linux, Windows und MAC OS zur Verfügung steht, aber standardmäßig nur für Windows mitgeliefert wird, um die Firmware zu installieren. Zum anderen kann das bereits angesprochene Programm *s-center* verwendet werden. Da *s-center* nur unter Windows läuft wird im folgendem Schritt das Programm *newtmgr* genutzt, um Plattformunabhängig zu bleiben. Nun wird *newtmgr* mit den passenden Parametern gestartet und die Firmware wird installiert.

---

<sup>19</sup>Eigene Darstellung

Der Befehl zum Installieren der Firmware schaut wie folgt aus:

```
newtmgr --conntype=serial --connstring="COMXX,baud=115200"  
  image upload <binary image>
```

An dieser Stelle muss nur noch *COMXX* mit dem passendem COM-Port ausgetauscht werden und für *<binary image>* die zuvor heruntergeladene Binärdatei ersetzt werden. Nach erfolgreicher Installation muss der Anker einmal neugestartet werden, um funktionsfähig zu sein. Dies wird mit folgendem Befehl oder durch Drücken des Reset-Buttons auf der Ankerplatine erreicht. Auch hier muss *COMXX* passend ersetzt werden.

```
newtmgr --conntype=serial --connstring="COMXX,baud=115200"  
  reset
```

## 5.2 Tag-Setup

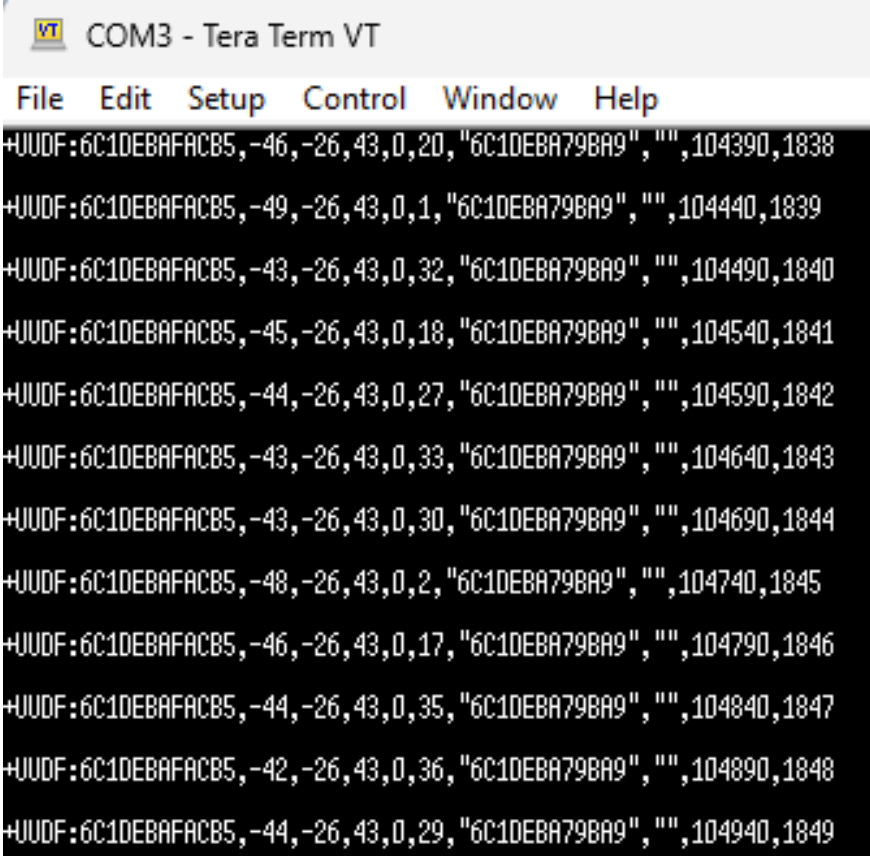
Zu Beginn wird der Tag mit Hilfe eines USB-A- auf Micro-B-Kabel mit dem Computer verbunden. Wie schon bei dem Anker-Setup werden die Treiber für den eingebauten FTDI USB-zu-UART-Wandler normalerweise automatisch installiert. Ist dies nicht der Fall, so müssen diese selbstständig heruntergeladen werden. Auch hier gibt es die Möglichkeit, die Firmware über zwei Wege zu installieren. Das Programm *s-center* kann dafür genutzt werden, aber wie bereits oben beschrieben, wird auch hier der alternative Weg gegangen. Da der Bootloader des Tags ein anderer als der Bootloader des Ankers ist, wird das *nrfutil* Tool (<https://github.com/NordicSemiconductor/pc-nrfutil/releases>) von Nordic benötigt, um Software via UART zu installieren. Die Software für den Tag ist in einem Repository (<https://github.com/u-blox/c209-aoa-tag/releases>) von ublox zu finden. Nachdem das Tool *nrfutil* und die Software für den Tag heruntergeladen worden sind, kann die Firmware installiert werden. Hierfür wird der SW2-Button des Tags gedrückt gehalten, während der Tag neugestartet wird, um den Bootloader in einen benötigten Modus zu versetzen. Ein Neustart kann durch das Einstecken des USB-Kabels oder das Drücken des Reset-Buttons erzwungen werden. Nun startet man das Tool mit dem folgenden Befehl,

```
nrfutil dfu serial -pkg app.zip -p COMXX -b 115200 -fc 1
```

um die Firmware zu installieren. Auch hier muss der *COMXX* durch den passenden COM-Port ersetzt werden und *app.zip* mit der zuvor heruntergeladene Datei ausgetauscht werden. Nachdem die Firmware aufgespielt ist, kann der Tag fortan mit einer CR2032-Batterie betrieben werden.

### 5.3 Anwendung

Nach erfolgreicher Inbetriebnahme der Hardware kann mit Hilfe eines Terminal-Emulators (beispielsweise Tera Term) der COM-Port passend konfiguriert werden, um die berechneten Ankunftszeitpunkte, sowie weitere Informationen, welche vom Anker zur Verfügung gestellt werden, anzuzeigen. Das Datenblatt für das System gibt Auskunft über die benötigte Konfiguration des Ports. Hier ist zu entnehmen, dass eine Baudrate von 115200 kbps, 8 Datenbits, kein Paritätsbit, 1 Stopbit und zusätzlich die Flusskontrolle durch die Nutzung von RTS/CTS verwendet werden müssen. Nachdem die Einstellungen übernommen worden sind, werden, wie in Abbildung 17 gezeigt, die ankommenden Daten des Ankers über die serielle Schnittstelle sichtbar.



```
VT COM3 - Tera Term VT
File Edit Setup Control Window Help
+UUDF:6C1DEBAFACB5,-46,-26,43,0,20,"6C1DEBA79BA9", "",104390,1838
+UUDF:6C1DEBAFACB5,-49,-26,43,0,1,"6C1DEBA79BA9", "",104440,1839
+UUDF:6C1DEBAFACB5,-43,-26,43,0,32,"6C1DEBA79BA9", "",104490,1840
+UUDF:6C1DEBAFACB5,-45,-26,43,0,18,"6C1DEBA79BA9", "",104540,1841
+UUDF:6C1DEBAFACB5,-44,-26,43,0,27,"6C1DEBA79BA9", "",104590,1842
+UUDF:6C1DEBAFACB5,-43,-26,43,0,33,"6C1DEBA79BA9", "",104640,1843
+UUDF:6C1DEBAFACB5,-43,-26,43,0,30,"6C1DEBA79BA9", "",104690,1844
+UUDF:6C1DEBAFACB5,-48,-26,43,0,2,"6C1DEBA79BA9", "",104740,1845
+UUDF:6C1DEBAFACB5,-46,-26,43,0,17,"6C1DEBA79BA9", "",104790,1846
+UUDF:6C1DEBAFACB5,-44,-26,43,0,35,"6C1DEBA79BA9", "",104840,1847
+UUDF:6C1DEBAFACB5,-42,-26,43,0,36,"6C1DEBA79BA9", "",104890,1848
+UUDF:6C1DEBAFACB5,-44,-26,43,0,29,"6C1DEBA79BA9", "",104940,1849
```

Abbildung 17: Datenübertragung des Ankers über die serielle Schnittstelle<sup>20</sup>

---

<sup>20</sup>Eigene Darstellung

Hier folgt nun eine kurze Beschreibung, wie die ankommenden Daten zu interpretieren sind. Das *+UUDF*: gibt an, dass es sich um ein Winkelberechnungs-Event handelt. Der zweite Parameter ist die Eddystone-Instanz-ID. Eddystone ist ein Bluetooth LE-Protokoll, das von Google entwickelt wurde. Die ID ist eine eindeutige Kennung, die von einem Gerät, welches das zuvor genannte Protokoll nutzt, gesendet wird und dient dazu, ein spezifisches Gerät zu identifizieren. Der dritte Parameter ist der RSSI. Er gibt die Signalstärke an, wie beispielsweise in der ersten Zeile den Wert *-46*. Die Parameter vier und fünf sind berechnete Winkel. Der Anker berechnet nicht nur einen Winkel, sondern zwei. Die geometrischen Beziehungen der beiden Winkel im Bezug auf die Ankerplatine sind in Abbildung 18 dargestellt. Der sechste Parameter wird von der Software aktuell nicht genutzt, deshalb steht dort standardmäßig eine *0* drin. Parameter sieben gibt den Kanal an, von dem das Paket berechnet wird. Der achte Parameter repräsentiert die MAC-Adresse des Ankers. Parameter neun ist eine benutzerdefinierte Zeichenfolge. Der vorletzte Parameter gibt die vergangene Zeit, seit Systemstart an. Der letzte Parameter ist ein periodischer Ereigniszähler von einem periodischen Nachrichtenverbreitungspaket, das von dem Tag kommt. [7]

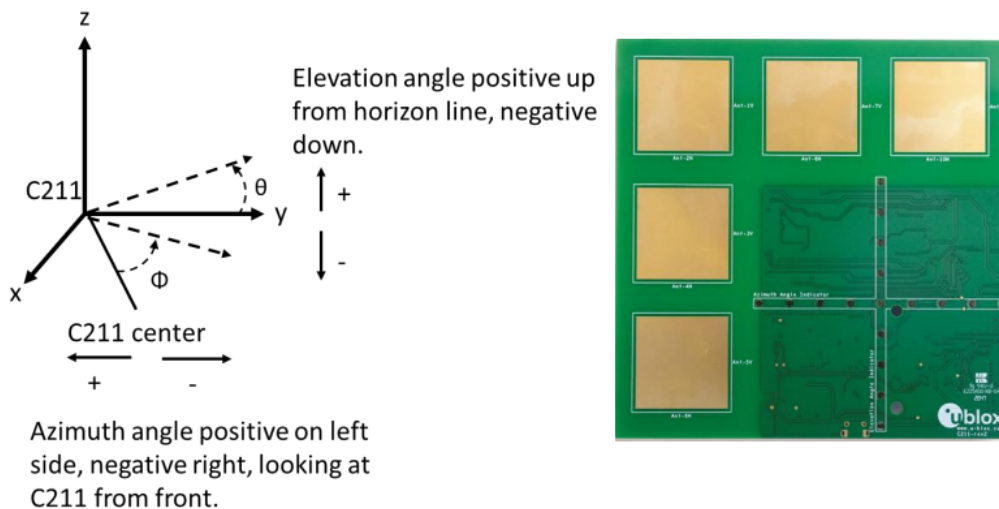


Abbildung 18: Winkelberechnung ausgehend vom Anker<sup>21</sup>

<sup>21</sup>[https://content.u-blox.com/sites/default/files/XPLR-AOA-Explorer-kits\\_UserGuide\\_UBX-21004616.pdf](https://content.u-blox.com/sites/default/files/XPLR-AOA-Explorer-kits_UserGuide_UBX-21004616.pdf)

Soll eine bessere Darstellung der Daten erreicht werden, so kann dies durch Nutzung des bereits erwähnten Programms *s-center* erfolgen. Hier folgt ein kurzes Beispiel, wie die Evaluierung mit Hilfe des graphischen Programms aussieht.

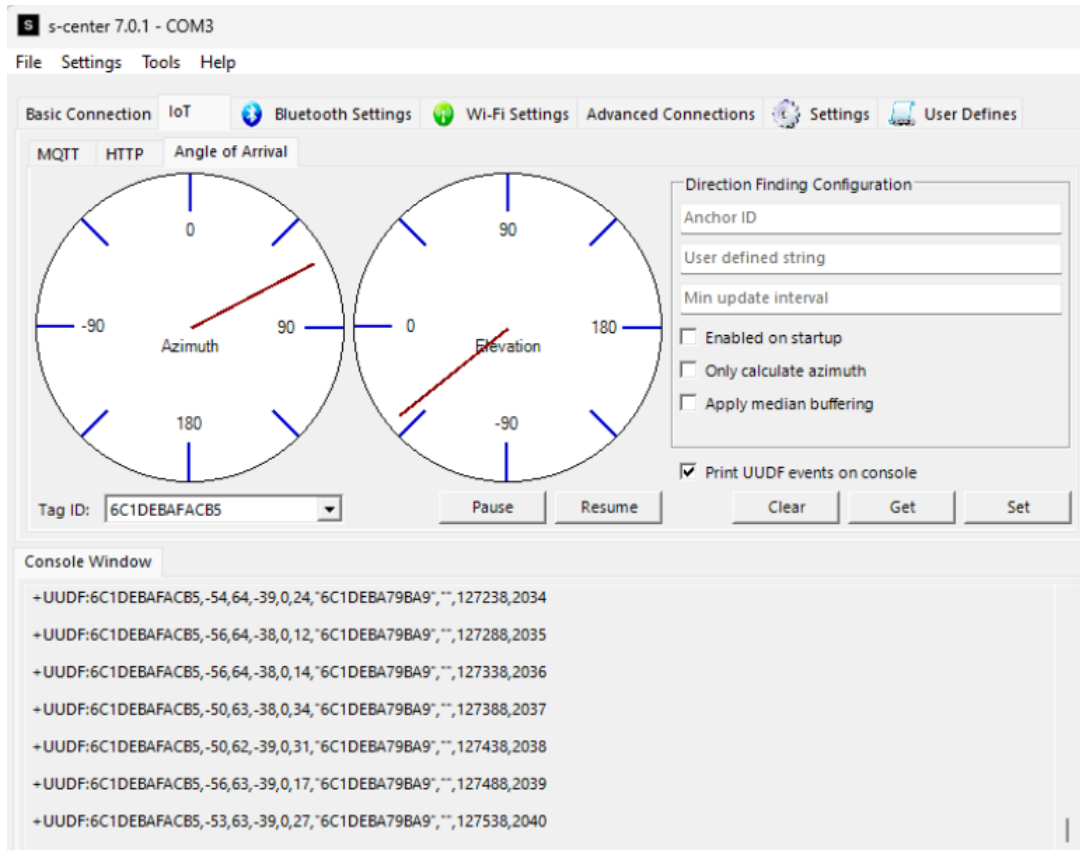


Abbildung 19: Darstellung der Daten mit s-center<sup>22</sup>

In der Graphik werden die beiden berechneten Winkel des Ankers in separaten Winkelanzeigen dargestellt. Darunter sind die zur Verfügung gestellten Daten der Ankerplatte in einem Konsolenfenster zu sehen. Diese können angezeigt werden, indem man die Checkbox für das Ausgeben der Winkelberechnungs-Event anklickt.

<sup>22</sup>Eigene Darstellung

## 6 Evaluierung des Systems

Im Kapitel 6 wird die *Bluetooth-Richtungsbestimmung* Technologie durch unterschiedliche Anwendungsszenarien genauer untersucht. Dabei wird vor allem auf die Faktoren Leistungsfähigkeit und Genauigkeit eingegangen.

### 6.1 Objekt oder Person steht statisch im Raum

In diesem Abschnitt wird untersucht, wie gut das System eine Person oder ein Objekt, welche sich an einem festen Punkt im Raum befindet, lokalisieren kann. Der Versuchsaufbau wird, wie in Abbildung 20 dargestellt, evaluiert.

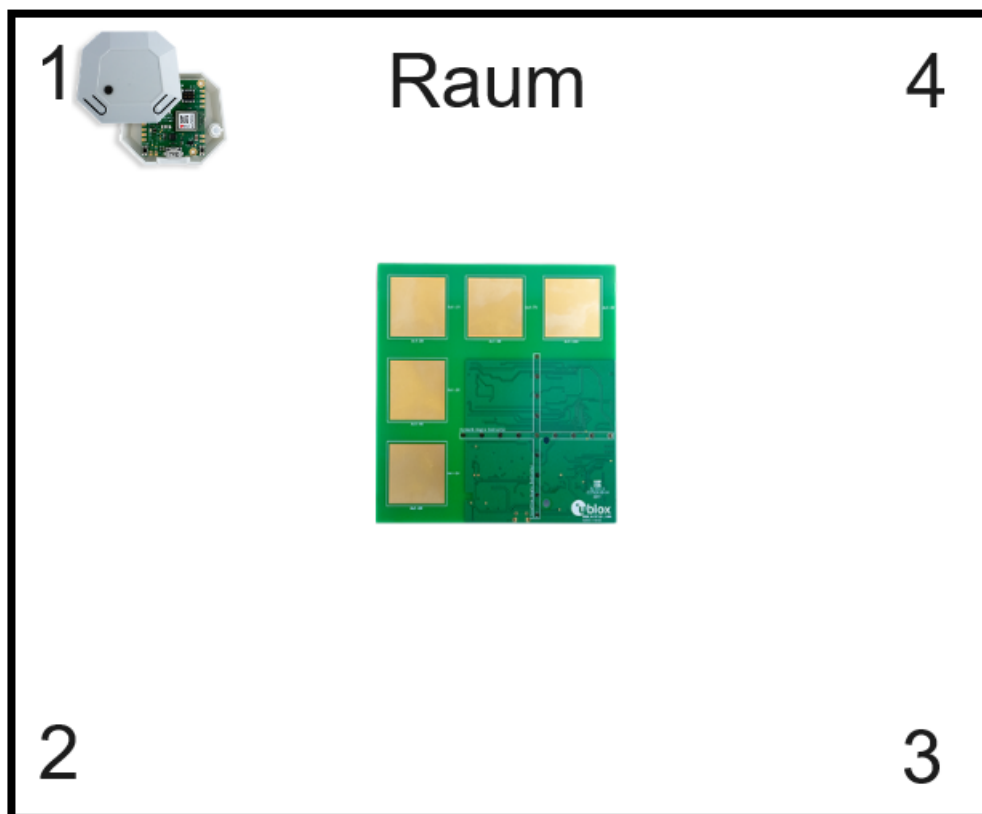


Abbildung 20: Person oder Objekt steht statisch<sup>23</sup>

<sup>23</sup>Eigene Darstellung, <https://www.u-blox.com/en/product/xplr-aoa-1-kit>

In diesem Beispiel befindet sich die zu lokalisierende Person beziehungsweise das zu lokalisierende Objekt wie in Abbildung 20 gezeigt in der linken oberen Ecke, gekennzeichnet durch die *1*, eines Raums. Mit Kenntnis der geometrischen Beziehungen, wie in Abbildung 18 zu sehen, werden Vorüberlegungen auf die resultierenden Azimuth- und Elevation-Winkel hin getroffen.

Die folgenden Werte sind zu erwarten für die ...

- ... Ecke 1: positiver Azimuth- und Elevation-Winkel
- ... Ecke 2: positiver Azimuth- aber negativer Elevation-Winkel
- ... Ecke 3: negativer Azimuth- und Elevation-Winkel
- ... Ecke 4: negativer Azimuth- aber positiver Elevation-Winkel

Die Abbildung 21 zeigt folgende berechnete Werte des Ankers für die Ecke 1.

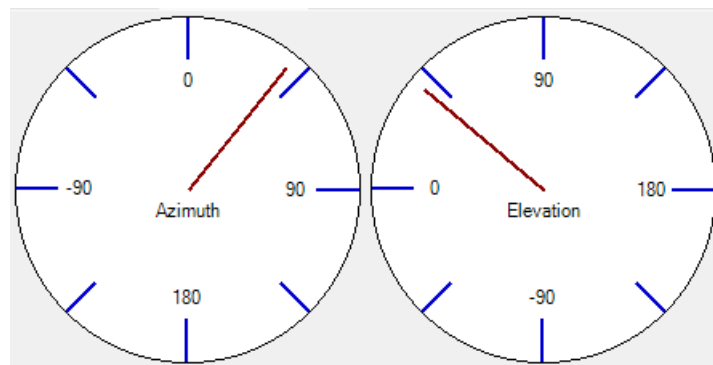


Abbildung 21: Messung in der linken oberen Ecke<sup>24</sup>

Wird das Ergebnis mit den Vorüberlegung verglichen, so ist erkennbar, dass das Ergebnis diese bestätigt.

---

<sup>24</sup>Eigene Darstellung



Der gleiche Versuch wird nun in Ecke 2 durchgeführt und erneut verglichen.

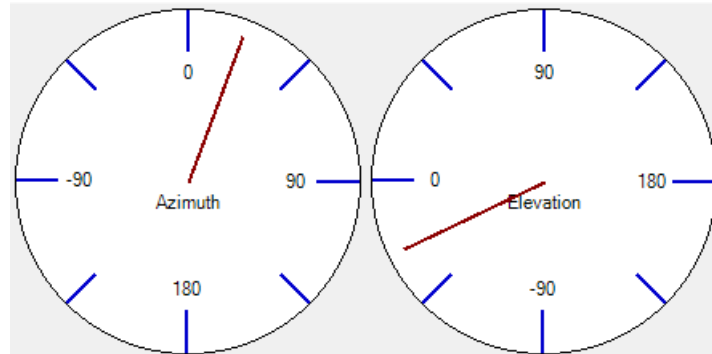


Abbildung 22: Messung in der linken unteren Ecke<sup>25</sup>

In Abbildung 22 ist das Ergebnis zu sehen. Auch hier stimmt das Resultat mit der Annahme überein. Die erwarteten Ergebnisse für den gleichen Versuch in den Ecken 3 und 4 sind ebenfalls zu bestätigen.

Der Versuch hat gezeigt, dass die Lokalisierung einer Person oder eines Objekts, die sich statisch in einem Raum befinden, mit dem System realisierbar ist. Die Genauigkeit des Systems ist dabei sehr gut. Es konnte zu jedem Zeitpunkt im Versuchsaufbau die korrekte Lokalisierung wiedergegeben werden.

---

<sup>25</sup>Eigene Darstellung

## 6.2 Objekt oder Person bewegt sich im Raum

Im folgenden Teil wird evaluiert, wie gut sich ein bewegendes Objekt oder eine sich bewegende Person lokalisieren lässt. Das Objekt oder die Person bewegt sich dabei in einer horizontalen Linie. Die Abbildung 23 zeigt den Aufbau des Versuchs.

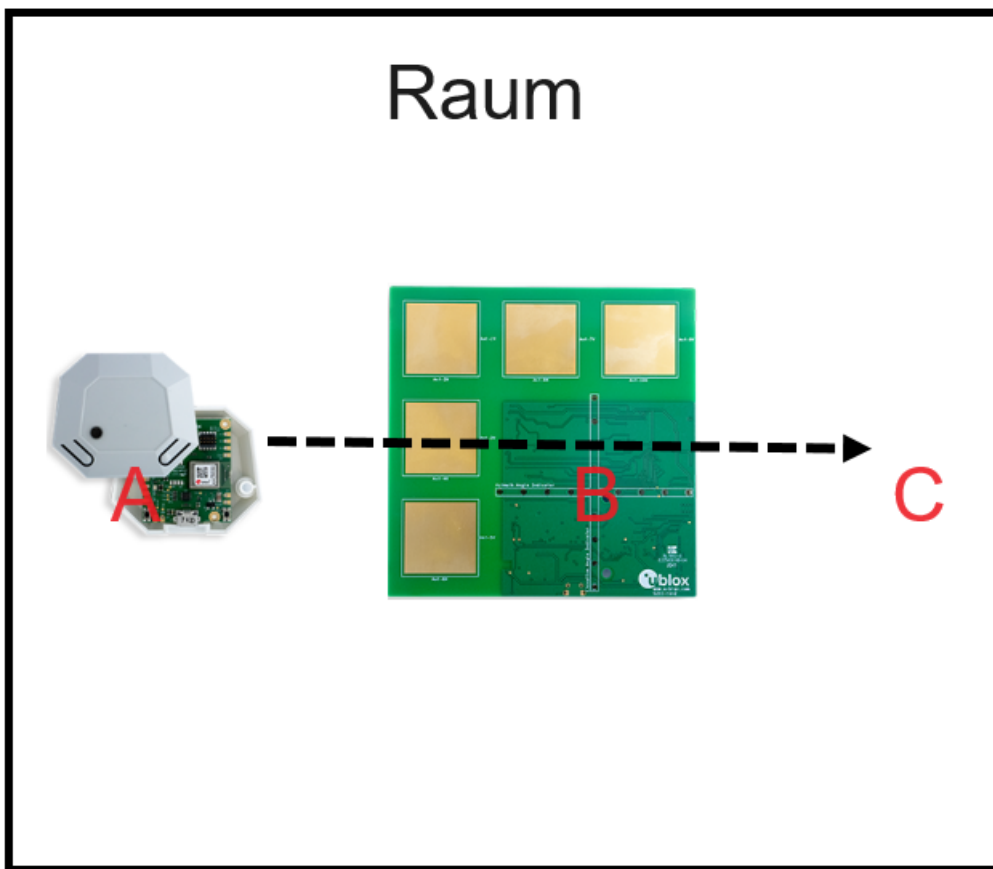


Abbildung 23: Person oder Objekt bewegt sich im Raum<sup>26</sup>

<sup>26</sup>Eigene Darstellung, <https://www.u-blox.com/en/product/xplr-aoa-1-kit>

Die Abbildung 24 zeigt das Ergebnis der Messung für die Position A im Raum. Zu erkennen ist, dass das Elevation-Winkel nahezu 0 ist und der Azimuth-Winkel einen positiven Wert besitzt.

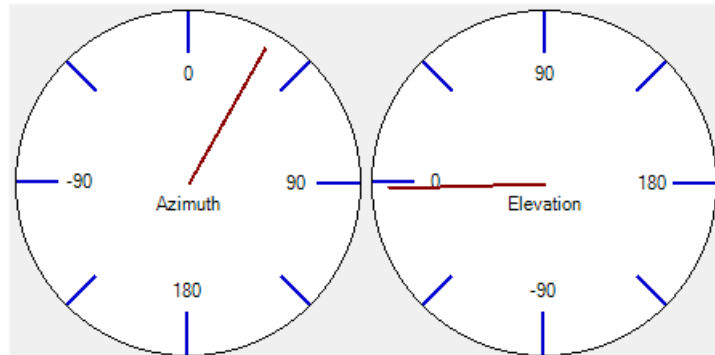


Abbildung 24: Position A im Raum<sup>27</sup>

In der Abbildung 25 befindet sich der Standort des Objekts oder der Person an Position B. In der Theorie nehmen beide Winkel den Wert 0 an. Das Ergebnis gibt dies auch wieder.

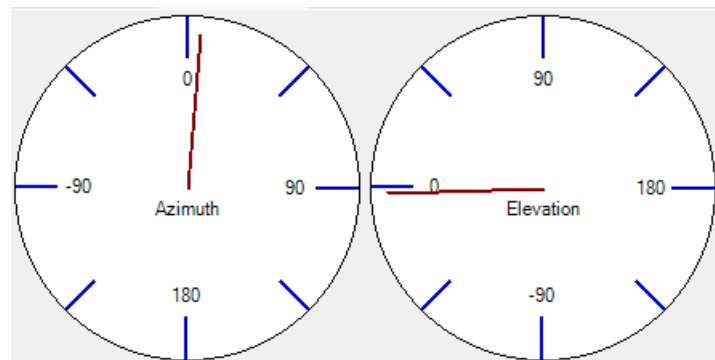


Abbildung 25: Position B im Raum<sup>28</sup>

---

<sup>27</sup>Eigene Darstellung

<sup>28</sup>Eigene Darstellung

Die Abbildung 26 visualisiert die beiden Winkel an Position C.

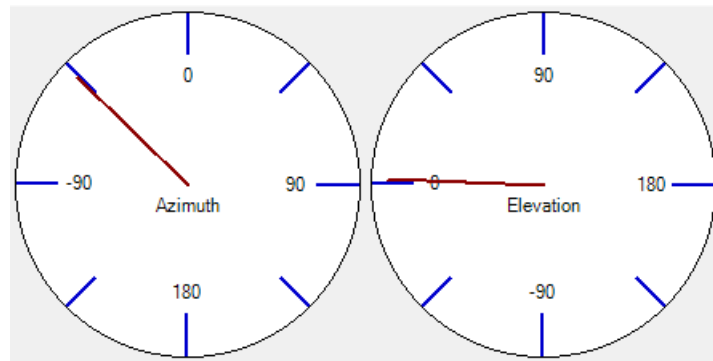


Abbildung 26: Position C im Raum<sup>29</sup>

Es werden die drei obigen Abbildungen genauer angeschaut. Dabei ist zu erkennen, dass der Elevation-Winkel nahezu konstant 0 beträgt, aber sich der Azimuth-Winkel von dem Wert 40 hin zu 0 und danach -45 verändert. Dies beschreibt eine horizontale Bewegung von links nach rechts.

Das Ergebnis zeigt, dass das System mächtig ist, ein sich bewegendes Objekt oder eine sich bewegende Person, in einem Raum zu lokalisieren. Zur Genauigkeit kann gesagt werden, dass das System keine Probleme hat, wenn es um die Lokalisierung von bewegenden Personen/Objekten geht. Die Winkeldarstellungen in der Software haben mit dem Verlauf der Bewegung sehr gut übereingestimmt.

---

<sup>29</sup>Eigene Darstellung

### 6.3 Seitenzuordnung einer Person oder eines Objekts

Mit diesem Anwendungsszenario wird beispielsweise untersucht, inwiefern das System erkennen kann, auf welcher Seite sich eine Person oder ein Objekt befindet. Die Abbildung 27 zeigt den Aufbau des Versuchs.

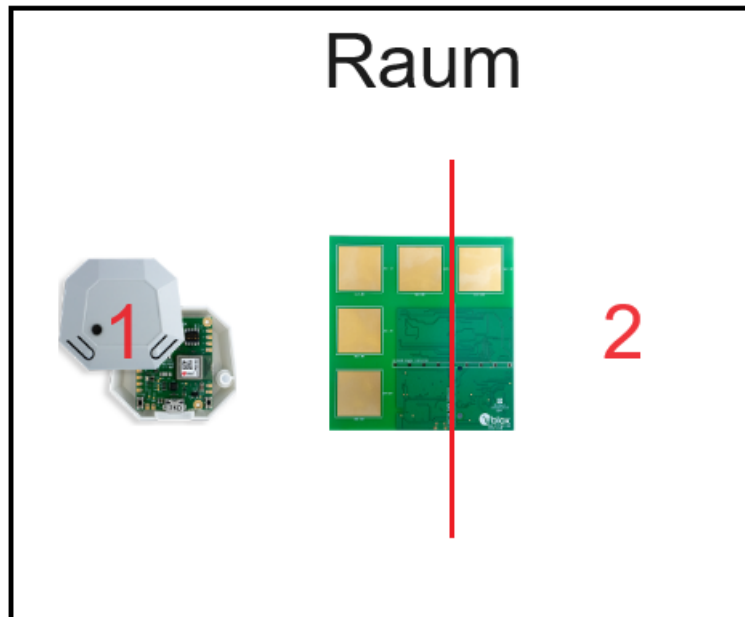


Abbildung 27: Zuordnung der Seite<sup>30</sup>

Befindet sich das Objekt beziehungsweise die Person an Position 1 wie in der obigen Abbildung zu sehen ist, so sollte der Azimuth-Winkel einen positiven Wert annehmen und der Elevation-Winkel bei dem Wert 0 liegen. An Position 2 bleibt der Elevation-Winkel bei dem Wert 0 und der Azimuth-Winkel sollte einen negativen Wert annehmen.

<sup>30</sup>Eigene Darstellung, <https://www.u-blox.com/en/product/xplr-aoa-1-kit>

In Abbildung 28 wird der Fall an Position 1 beschrieben. Abbildung 29 beschreibt den Fall in Bezug auf Position 2.

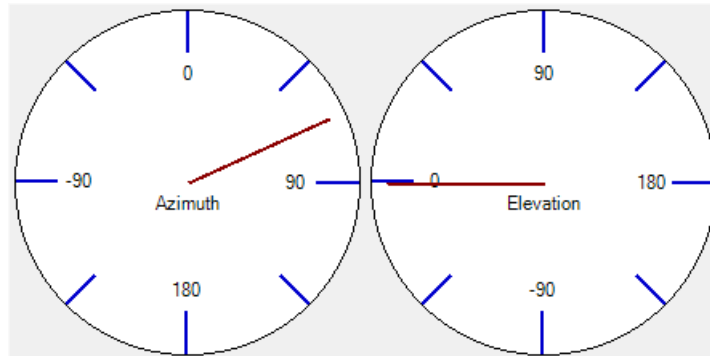


Abbildung 28: Seitenzuordnung Links<sup>31</sup>



Abbildung 29: Seitenzuordnung Rechts<sup>32</sup>

Beobachtet werden kann, dass das System in der Lage ist eine korrekte Lokalisierung, im Bezug auf eine Seitenzuordnung bei einem gegebenen Bezugspunkt, treffen kann. Die Genauigkeit ist wie schon bei den beiden vorangegangenen Versuchen sehr gut.

---

<sup>31</sup>Eigene Darstellung

<sup>32</sup>Eigene Darstellung

## 6.4 Maximale Reichweite des Systems

Im letzten Versuch geht es darum, herauszufinden, ab welcher Distanz das System keine Winkelberechnungen mehr durchführen kann und somit nicht mehr funktioniert. Der Versuchsaufbau wird wie in Abbildung 30 realisiert.

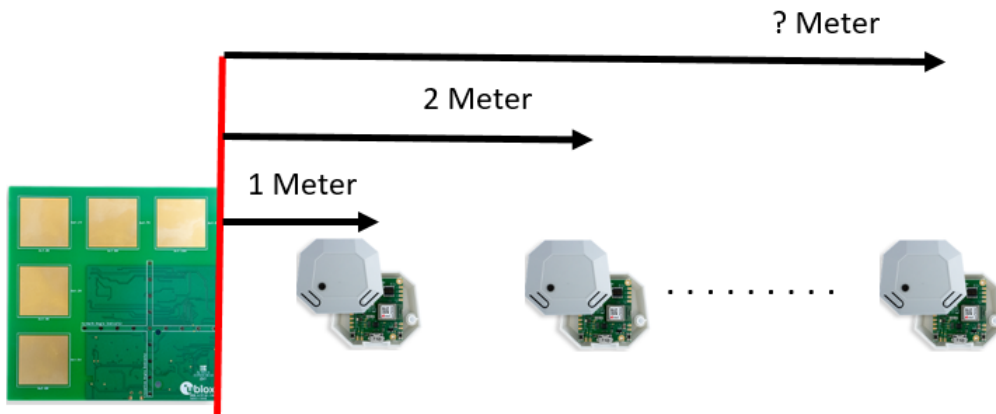


Abbildung 30: Maximale Reichweite des Systems<sup>33</sup>

Der Tag wird beginnend mit einem Abstand von 1 Meter zum Anker platziert. Dann sind zehn Messungen des RSSI bei einem Abstand von 1 Meter gemacht worden. Als nächstes wurde der Tag in einem Abstand von 2 Metern zur Ankerplatine platziert. Es wurden wieder zehn Messungen des RSSI gemacht. Dieser Prozess wird so lange durchgeführt, bis die Ankerplatine keine Daten mehr über die serielle Schnittstelle liefern kann. Die folgende Abbildung 31 zeigt die gemessenen Werte in gekürzter Form.

Distanz (Meter)	2	4	6	8	10	12	14	16	18	20	22	24	26
RSSI (dBm)	-76	-72	-88	-75	-81	-80	-81	-78	-82	-87	-84	-84	-88
	-74	-70	-74	-86	-78	-82	-81	-87	-89	-88	-79	-86	-88
	-73	-80	-79	-72	-86	-79	-85	-87	-84	-87	-84	-86	-87
	-69	-79	-73	-77	-77	-81	-83	-86	-80	-85	-81	-85	-88
	-71	-75	-76	-76	-78	-81	-82	-87	-84	-90	-85	-83	-87
	-72	-70	-85	-80	-88	-81	-84	-82	-86	-87	-85	-86	-87
	-71	-71	-80	-75	-86	-80	-88	-80	-84	-83	-85	-88	-87
	-74	-81	-75	-74	-86	-79	-82	-83	-86	-82	-80	-85	-87
	-72	-74	-79	-78	-78	-81	-87	-82	-84	-85	-81	-84	-87
	-71	-73	-77	-77	-77	-81	-89	-80	-83	-86	-88	-86	-86
Mittelwert	-72,3	-74,5	-78,6	-77	-81,5	-80,5	-84,2	-83,2	-84,2	-86	-83,2	-85,3	-87,3

Abbildung 31: Gemessene RSSI Werte<sup>34</sup>

<sup>33</sup>Eigene Darstellung, <https://www.u-blox.com/en/product/xplr-aoa-1-kit>

<sup>34</sup>Eigene Darstellung, aus Platzgründen ist die Tabelle gekürzt worden

Das Resultat dieses Versuches ist, dass das System ab einer Distanz von 26 Metern die Verbindung verliert und somit keine Lokalisierung des Tags mehr erfolgen kann. Die Abbildung 32 bildet die einzelnen Mittelwerte der jeweiligen Messungen über die zugehörige Distanz ab.

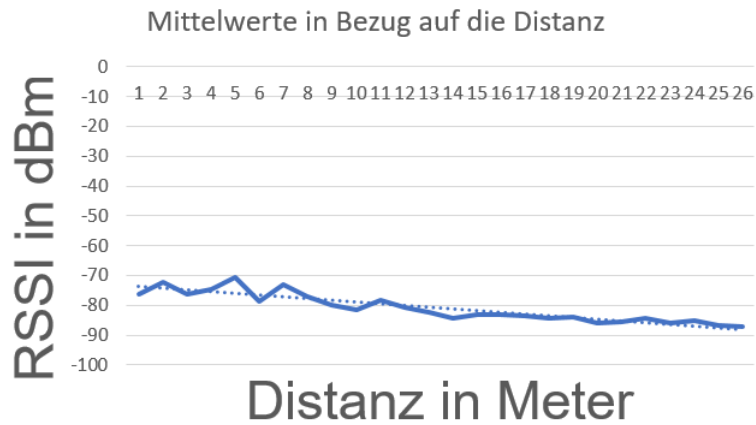


Abbildung 32: Mittelwerte in Bezug auf die Distanz<sup>35</sup>

Im obigen Diagramm wird folgender Zusammenhang erkennbar: Je größer die Distanz zwischen der Ankerplatine und dem Tag, desto geringer die Signalstärke.

---

<sup>35</sup>Eigene Darstellung



## 7 Selbst entwickelte GUI

Die Firma ublox stellt in ihren Explorer-Kits neben der Hardware auch Software zur Verfügung. Vor allem die Software *s-center* ist hierbei zu erwähnen. Mit Hilfe der Software ist es unter anderem möglich, die Funktionalität der Bluetooth-Richtungsbestimmung graphisch zu veranschaulichen. Jedoch steht das Tool gegenwärtig nur für die Betriebssystemplattform Windows zur Verfügung. Somit sind Nutzer, welche eine andere Betriebssystemplattform benutzen, erheblich eingeschränkt bei der Nutzung des Explorer-Kits. Aus diesem Grund wird in dieser Arbeit ein Programm entwickelt, welches die graphische Auswertung der Bluetooth-Richtungsbestimmung plattformunabhängig zur Verfügung stellt. Im Folgenden wird der Aufbau des Programms genauer analysiert. Der Quellcode ist in der Datei *Angles.py* zu finden, die sich im Abgabeordner der Arbeit befindet.

### 7.1 Aufbau des Programms

In den Zeilen 2 bis 5 werden Pakete eingebunden, um spezielle Funktionen im späteren Verlauf des Programms nutzen zu können. Hierbei sind kurz die Pakete *serial* und *t-kinter* zu erwähnen. Das Paket *serial* wird genutzt, um den Zugriff auf die serielle Schnittstelle zur Verfügung zu stellen. Über diese Schnittstelle werden die Daten, welche die Ankerplatine zur Verfügung stellt abgegriffen. Das Paket *t-kinter* wird genutzt, um dem Programm eine graphische Oberfläche zur Verfügung zu stellen.

Die darauffolgenden Zeilen 8 bis 11 erzeugen ein Objekt für die serielle Schnittstelle und konfigurieren es so, um die Daten von der Ankerplatine zu erhalten.

Mit der Zeile 16 wird dann versucht die angelegte serielle Schnittstelle zu öffnen. In der Zeile 17 wird eine Fehlerroutine eingebaut, falls das Öffnen der Schnittstelle nicht erfolgreich war.

Falls es zu keinem Fehler gekommen ist wird in den Zeilen 24 bis 37 eine Fensterinstanz angelegt, die Instanz konfiguriert und ein Objekt angelegt, um in das Fenster reinzeichnen zu können.

Die Zeilen 40 bis 46 legen Variablen an, die später genutzt werden, um zwei Kreise in dem Fenster darzustellen. Die Kreise sind dazu da, die berechneten Winkel der Ankerplatine graphisch zu visualisieren.

In Zeile 48 wird eine Funktion geschrieben, die später eine Linie innerhalb der

Kreise zeichnet. Die Linie repräsentiert dabei die berechneten Winkel der Ankerplatine. Die Zeile 50 definiert eine Funktion, welche Hilfslinien in die Kreise zeichnet, um die visuelle Darstellung zu verbessern.

Von der Zeile 63 bis 81 werden Textfelder angelegt und gleichzeitig im Fenster platziert. Dabei präsentieren zwei der angelegten Textfelder die berechneten Winkel durch die Ankerplatine und die restlichen Felder dienen als Hilfsanzeige für die Winkel  $0^\circ$ ,  $90^\circ$  und  $-90^\circ$ .

Die Zeilen 84 und 85 legen zwei Variablen für die beiden Winkel an.

Danach beginnt mit Zeile 87 die Hauptschleife des Programms. Mit den Zeilen 89 und 90 werden die beiden Kreise in das Fenster gezeichnet. Zeile 93 bis 101 erzeugt die vorher genannten Hilfslinien in den Kreisen.

Mit der Zeile 104 wird dann versucht, die ankommenden Daten über der seriellen Schnittstelle zu lesen. Eine kleine Fehlerbehandlung ist folgend auch implementiert, da es bei dem Systemstart manchmal zu einem Fehler kommen kann. Durch die Behandlung des Fehlers kommt das Programm aber nicht zum Absturz, sondern es wird einfach im nächsten Durchlauf wieder versucht gültige Daten zu lesen.

Werden dann gültige Daten gelesen, so sucht man mit der Zeile 110 nach einem bestimmten Muster in den Daten und bei erfolgreicher Suche extrahiert die Zeile 112 die beiden Winkel aus den Daten. Die Zeilen 113 und 114 aktualisieren die beiden Textfelder für die Winkel im jedem Durchlauf. Um später die richtige Drehrichtung der beiden Linien zu gewährleisten, wird in den Zeilen 116 und 117 eine kurze Zwischenrechnung durchgeführt. Danach wird die Funktion zum Zeichnen der Linien in den beiden Kreisen zweimal aufgerufen.

Im letzten Teil des Programms von Zeile 123 bis 130 werden Funktionen aufgerufen, um das Fenster regelmäßig zu updaten und die Zeichenfläche im Fenster regelmäßig zu löschen, damit im nächsten Durchlauf die Graphiken neu gezeichnet werden können. Wenn das Programm beendet wird, schließt die Zeile 132 die zuvor geöffnete serielle Schnittstelle wieder.

## 7.2 Anwendungsbeispiel

In Abbildung 33 ist die graphische Oberfläche des Programms zur Laufzeit zu sehen.



Abbildung 33: Visualisierung der Winkel zur Laufzeit<sup>36</sup>

Hier ist zu sehen, dass die Ankerplatine einen Azimuth-Winkel von  $38^\circ$  und einen Elevation-Winkel von  $-33^\circ$  berechnet hat. Die Winkel werden dann in die für sie vorgesehene Winkelanzeige dargestellt.

---

<sup>36</sup>Eigene Darstellung

## 8 Fazit und Ausblick

Diese Arbeit hat das Lokalisierungssystem **XPLR-AOA-1 kit** in unterschiedlichen Anwendungsszenarien untersucht. Dabei hat sich ergeben, dass das System in der Lage ist statische oder bewegende Objekte/Personen in einem Raum zu lokalisieren. Die Genauigkeit, mit der das System dies tut, ist dabei beachtlich gut. Anzumerken ist aber, dass die Winkeldarstellungen in der Software nicht immer akkurat sind. So kann es vorkommen, dass kleine Sprünge um den eigentlich betrachteten Wert angezeigt werden. Dieses Problem ist bei allen Versuchen immer mal wieder zu beobachten gewesen. Sehr positiv zu erwähnen ist, dass das System in der Lage ist zu erkennen auf welcher Seite sich ein Objekt beziehungsweise eine Person befindet. Das System ist in diesem Versuch hierbei als Tür betrachtet worden. Es konnte in diesem Beispiel sehr gut festgestellt werden, auf welcher Seite der Tür sich eine Person befand und ob diese sich der Tür angenähert hat. Abschließend ist die maximale Distanz, in der das System noch funktionsfähig ist, getestet worden. Dabei hat sich herausgestellt, dass das System erst bei einer Entfernung von mehr als 26 Metern nicht mehr funktionsfähig ist. Dieser Wert ist schon sehr beeindruckend und war zu Beginn der Arbeit nicht zu erwarten. Jedoch muss hier gesagt werden, dass der Hersteller einen optimalen Abstand zwischen der Ankerplatine und dem Tag von 15 Metern angibt. Nach den 15 Metern kann es zu leichten Abweichungen zwischen den berechneten Winkeln durch den Anker und den tatsächlichen Winkeln kommen. Abschließend ist zu sagen, dass die Bluetooth-Richtungsbestimmung mit dem XPLR-AOA-1 kit schon sehr gut funktioniert. Diese Technologie wird in Zukunft mit Sicherheit ihre Anwendung in unterschiedlichen alltäglichen Lebenssituationen finden. Das Schlusswort ist, dass auf dieser Arbeit, in Form einer Bachelorarbeit oder Masterarbeit, aufgebaut werden kann, indem mehrere dieser XPLR-AOA-1 kits genutzt werden. Somit kann ein Positionsbestimmungssystem durch die Kombination dieser Kits realisiert und untersucht werden.

## Literatur

- [1] Bluetooth Special Interest Group. Bluetooth direction finding, 2023. [Online; <https://www.bluetooth.com/learn-about-bluetooth/feature-enhancements/direction-finding/>; Stand 15.11.2023].
- [2] Bluetooth Special Interest Group. Bluetooth drahtlose technologie, 2023. [Online; <https://www.bluetooth.com/de/learn-about-bluetooth/tech-overview/>; Stand 16.11.2023].
- [3] Peter Karlsson. Bluetooth for high precision indoor positioning, 2023. [Online; <https://content.u-blox.com/sites/default/files/Indoor-positioning-Getting-started-u-blox-WhitePaper.pdf>; Stand 07.11.2023].
- [4] u-blox. Bluetooth indoor positioning, 2020. [Online; <https://www.u-blox.com/en/technologies/bluetooth-indoor-positioning>; Stand 15.11.2023].
- [5] ublox. Xplr-aoa-1, 2023. [Online; <https://www.u-blox.com/en/product/xplr-aoa-1-kit>; Stand 25.10.2023].
- [6] wikipedia. Bluetooth low energy, 2023. [Online; [https://de.wikipedia.org/wiki/Bluetooth\\_Low\\_Energy](https://de.wikipedia.org/wiki/Bluetooth_Low_Energy); Stand 01.11.2023].
- [7] wikipedia. Eddystone (google), 2023. [Online; [https://en.wikipedia.org/wiki/Eddystone\\_\(Google\)](https://en.wikipedia.org/wiki/Eddystone_(Google)); Stand 01.11.2023].
- [8] wikipedia. Received signal strength indication, 2023. [Online; [https://de.wikipedia.org/wiki/Received\\_Signal\\_Strength\\_Indication](https://de.wikipedia.org/wiki/Received_Signal_Strength_Indication); Stand 15.11.2023].
- [9] wiliot. Bluetooth tag vs. beacon, 2022. [Online; <https://www.wiliot.com/bluetooth-tag-vs-beacon>; Stand 13.11.2023].
- [10] Martin Woolley. Bluetooth direction finding - a technical overview, 2021. [Online; [https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF\\_Technical\\_Overview.pdf](https://www.bluetooth.com/wp-content/uploads/Files/developer/RDF_Technical_Overview.pdf); Stand 15.11.2023].
- [11] Torbjørn Øvrebek. Bluetooth 5 advertising extensions, 2022. [Online; <https://blog.nordicsemi.com/getconnected/bluetooth-5-advertising-extensions>; Stand 07.11.2023].