



Hochschule Magdeburg-Stendal
Fachbereich Ingenieurwissenschaften und
Industriedesign (IWID)
Institut für Elektrotechnik

Bachelorarbeit

zur Erlangung des Grades eines

„Bachelor of Engineering“

im Studiengang Elektrotechnik

Thema: Exploration und Schnittstellenprogrammierung eines GNSS-Empfängers zur Richtungs- und Positionsbestimmung von HF-Messdrohnen

Eingereicht von: Toni Prietz

Angefertigt für: Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahn

Matrikel: 2019

Ausgabetermin: 29.02.2024

Abgabetermin: 17.04.2024

Erstprüfer: Herr Prof. Dr. techn. Sebastian Hantscher

Zweitprüfer: Herr Richard Rauwolf M.Eng

.....
1. Prüfer

.....
2. Prüfer

Eidesstattliche Erklärung

Hiermit bestätige ich, Toni Prietz 20192196, dass ich die vorliegende Arbeit mit dem Titel „Exploration und Schnittstellenprogrammierung eines GNSS-Empfängers zur Richtungs- und Positionsbestimmung von HF-Messdrohnen“ selbstständig und ohne die Hilfe anderer Personen angefertigt habe.

Ich habe nur die konkret angegebenen Quellen und Hilfsmittel und diese nur in der angegebenen Form verwendet.

Aus fremden Werken und Quellen entnommene Inhalte, wörtliche Zitate oder sinngemäße Inhalte, z. B. der Argumentation nach, und IT-/KI-generierte Inhalte habe ich an der jeweiligen Stelle unter Angabe der konkreten Quellen gekennzeichnet.

Mir ist bewusst, dass bei dem Versuch, das Ergebnis einer Prüfungsleistung durch Täuschung oder Benutzung nicht zugelassener Hilfsmittel zu beeinflussen, die betreffende Prüfungsleistung mit „nicht ausreichend“ / „nicht erfolgreich abgeschlossen“ zu bewerten ist, beziehungsweise die betreffende Prüfungsleistung als mit „nicht ausreichend“/“nicht erfolgreich abgeschlossen“ bewertet gilt (Muster-SPO der Hochschule Magdeburg-Stendal vom 23.03.2023 § 35 Abs. 3 Satz 1).

Ich bestätige ausdrücklich, dass diese Arbeit weder vollständig noch teilweise einer anderen Prüfungsbehörde vorgelegt oder veröffentlicht worden ist.

Ich stimme zu, dass die Arbeit in eine Datenbank zur Plagiats- bzw. Hilfsmittelnutzungsprüfung eingestellt und gespeichert wird.

Toni Prietz, den 17.04.2024

Inhalt

Einleitung	1
Motivation	1
1 Theoretische Grundlagen	3
1.1 GNSS	3
1.1.1 Codemessung	4
1.1.2 Phasenmessung.....	5
1.1.3 Fehlereinflüsse	6
1.2 Differentielles GNSS	8
1.2.1 Absolute und relative Positionsbestimmung	9
1.2.2 Echtzeitkinematik.....	10
1.2.3 Korrekturverfahren.....	12
1.3 Koordinatensysteme	13
1.3.1 ECEF-Koordinaten.....	13
1.3.2 Ellipsoidische Koordinaten	13
1.3.3 NED-Koordinaten	14
2 Konzept	15
2.1 GNSS-Kompass.....	15
2.2 Positionierung mit eigener Basisstation	16
2.3 Positionierung ohne eigene Basisstation	18
2.3.1 SBAS.....	18
2.3.2 SAPOS	18
2.3.3 Point-Perfect.....	19
3 Equipment	20
3.1 Empfänger und Antennen.....	20
3.2 Übertragungsprotokolle	23
3.2.1 NMEA-Standard.....	23

3.2.2	RTCM-Standard	24
3.2.3	UBX-Protokoll	24
3.3	Konfiguration und Datenaustausch	26
3.3.1	Konfiguration für Positionsbestimmungen	28
3.3.2	Konfiguration des GNSS-Kompasses	29
3.4	Programmierschnittstelle	30
4	Praktische Erprobung	33
4.1	Test des GNSS-Kompasses	33
4.1.1	Messaufbau.....	33
4.1.2	Test im Stillstand.....	36
4.1.3	Test in Fahrt	37
4.1.4	Test bei hoher Winkelgeschwindigkeit	40
4.2	Test der Positionsgenauigkeit	42
4.2.1	Messaufbau.....	42
4.2.2	Einzelpunktmessung.....	43
4.2.3	Einmessung der Referenzstation	46
4.2.4	Korrekturdienstgestützte RTK-Messung.....	50
4.2.5	Berechnung des Drohnenmittelpunktes.....	52
5	Fazit und Ausblick	54
6	Abkürzungsverzeichnis	57
7	Literaturverzeichnis.....	59
8	Anhang	62
8.1	Empfängerkonfiguration GNSS-Kompass	62
8.1.1	Moving-Base-Empfänger 5 Hz Messrate	62
8.1.2	Rover-Empfänger 5 Hz Messrate	63
8.2	Python-Code GNSS-Kompass.....	64

Einleitung

Der Prüf- und Messdienst (PMD) der Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahn überprüft bundesweit in Deutschland Sendefunkanlagen u. a. auf die Einhaltung von Frequenzuteilungen, Versorgungsverpflichtungen und EMV-Personenschutzgrenzwerten. Zudem leistet der PMD eine Störungsbearbeitung, wobei Funkstörungen und Störquellen jeglicher Art ermittelt und ggf. beseitigt werden.

Im Normalfall werden diese Aufgaben mithilfe von Dienstmessfahrzeugen erfüllt, welche mit entsprechender Messtechnik ausgerüstet sind. Insbesondere dann, wenn die Erreichbarkeit von vorgegebenen Messstandorten bei der Vermessung einer Sendefunkanlage und die Empfangbarkeit einer Funkstörung aufgrund von fehlenden Straßen, Bewaldung, Abschattung durch Gebäude oder Berge oder einer zu geringen Messhöhe mittels Fahrzeugen nicht gegeben ist, können Drohnenmessungen aus der Luft dazu beitragen, Überprüfungen und Störungsbearbeitungen erheblich zu erleichtern. Ein mögliches Szenario stellen Störungen im Flugfunk- oder Richtfunkbereich dar, welche am Boden nur schwer oder gar nicht empfangen werden können, insbesondere dann, wenn die Störungspeilung im Bereich der Hauptstrahlrichtung der gestörten Antenne erfolgen muss. Des Weiteren können Drohnenmessungen beispielsweise die Ermittlung von Strahlungsdiagrammen großer Rundfunksender erleichtern, um ein korrektes Abstrahlverhalten der Antennenanlage und die Einhaltung des zugeteilten Strahlungsdiagrammes zu überprüfen.

Zurzeit werden für Drohnenmessungen externe Unternehmen beauftragt, was zu Inflexibilität und zusätzlichen Kosten führt, weshalb künftig eigene Messdrohnen im PMD zum Einsatz kommen sollen. Für die Navigation, Peilungen bei der Störungssuche und die Auswertbarkeit von Messdaten müssen die Drohnen mit einer möglichst hochgenauen Richtungs- und Positionsbestimmung ausgerüstet sein.

Motivation

In einer vorangegangenen Arbeit wurden bereits verschiedene Möglichkeiten für die Richtungsbestimmung analysiert. Konventionelle Methoden wie Magnetkompass haben sich aufgrund von Magnetfeldeinflüssen der Drohnenmotoren auf den Kompass als unbrauchbar erwiesen. Auch Methoden der ungestützten inertialen Messsysteme (Trägheitssensorik) für die Richtungs- und Positionsbestimmung sind für diesen Anwendungsbereich nicht geeignet, weil diese

u.a. eine mit der Zeit abnehmende Genauigkeit aufweisen, da sich entstandene Fehler akkumulieren. Zudem benötigen diese Systeme eine anfängliche Initialisierung für absolute Richtungs- und Positionsergebnisse, da diese nur relative Werte bezogen auf die Ausgangssituation der Messung liefern. [1, pp. 20-21]

Aufgrund dessen hat man sich für eine Lösung mittels Global Navigation Satellite System (GNSS) entschieden. Mithilfe eines Messaufbaus bestehend aus zwei Empfängern und Antennen ist es möglich, zu jedem Zeitpunkt und unabhängig von Bewegungen der Drohne sowohl eine Richtungs- als auch eine Positionsbestimmung zu realisieren. In der vorangegangenen Arbeit wurde bereits ein GNSS-Empfänger der Firma „Septentrio“ untersucht und erprobt, welcher jedoch eine unzureichende Positionsgenauigkeit aufwies. [1, p. 34] Zudem hat sich gezeigt, dass die Richtungsbestimmung des Empfängers bei dynamischen Testmessungen drastische Fehler hervorruft, was eine Anwendung auf einer Drohne unmöglich macht.

Für die Lösung des bestehenden Problems soll in dieser Arbeit ein Konzept zur Richtungs- und Positionsbestimmung mit GNSS-Empfängern der Firma u-blox untersucht und erprobt werden. Schwerpunkte werden Messungen zur Bestimmung der Positions- und Richtungsgenauigkeit bei verschiedenen Szenarien sein, um einschätzen zu können, ob eine Anwendung der zu untersuchenden Empfänger auf einer Drohne und ggf. auch auf Messfahrzeugen des PMD realisierbar ist. Zudem wird eine Übersicht über die Empfängerkonfigurationen mithilfe der Evaluierungssoftware „u-center“ gegeben. Für die Anwendung der GNSS-Empfänger auf einer Messdrohne muss eine Konfigurations- und Empfangsschnittstelle geschaffen werden, um zum einen die Empfänger über ein proprietäres Übertragungsprotokoll ohne Evaluierungssoftware konfigurieren– und zum anderen die Empfängerdaten fernauslesen und weiterverarbeiten zu können.

1 Theoretische Grundlagen

Die technischen Aspekte der Satellitenpositionierung sind äußerst vielfältig und umfangreich. Dieses Kapitel enthält eine kurze Erläuterung des Grundprinzips der GNSS-Systeme, der damit verbundenen Fehlereinflüsse und Möglichkeiten für deren Korrektur sowie die für diese Arbeit relevanten Begriffsklärungen.

1.1 GNSS

Das Global Navigation Satellite System ist ein Sammelbegriff für die derzeit existierenden Positionierungsdienste, darunter u.a. die Systeme NAVSTAR GPS (USA), GLONASS (Russland), Galileo (Europa) und BeiDou (China). Alle Systeme verwenden dabei das gleiche Prinzip für die Positionsbestimmung: ein GNSS-Empfänger empfängt Satellitensignale und misst mithilfe von Signalzeitmarken deren Laufzeit vom Aussenden des Signals bis zum Empfangszeitpunkt. Die Satellitensignale breiten sich mit Lichtgeschwindigkeit aus, dementsprechend kann mit der gemessenen Laufzeit auf den Abstand zwischen Satellit und Empfänger geschlossen werden. Stellt man ein Gleichungssystem zur Bestimmung einer Position im dreidimensionalen Raum mithilfe der berechneten Distanzen auf, erhält man drei Unbekannte für jede Dimension. Man benötigt also theoretisch die Distanzen zu drei Satelliten, um das Gleichungssystem lösen zu können. Im Gegensatz zu den Satelliten besitzen die Empfänger jedoch keine hochgenauen und mit den Satelliten synchronisierten Atomuhren, weshalb sich bei den Laufzeitmessungen ein Zeitfehler ergibt. Die berechneten Entfernungen aus den gemessenen Signallaufzeiten entsprechen also nicht den tatsächlichen Entfernungen zwischen Satellit und Empfänger, weshalb man auch von Pseudoentfernungen spricht. Da jedoch alle Satellitenuhren untereinander synchronisiert sind und dementsprechend der Zeitfehler in allen Laufzeitmessungen gleich groß ist, kann dieser durch eine Erweiterung des Gleichungssystems um eine weitere Unbekannte berechnet werden. Demzufolge müssen für eine Positionsbestimmung die Signallaufzeiten von mindestens *vier* verschiedenen Satelliten gemessen werden. [2, pp. 30-35]

Die geometrische Entfernung ρ zwischen Satellit und Empfänger wird durch die Bildung des Betrags des Entfernungsvektors in Gleichung (1) berechnet, welcher sich durch die Differenz zwischen Empfänger- und Satellitenortsvektor bildet. Als Bezugspunkt dient der Koordinatenursprung des ECEF-Koordinatensystems, welches in einem späteren Abschnitt erläutert wird. Die Koordinaten des Satellitenortsvektors (X_{SV}, Y_{SV}, Z_{SV}) werden vom Empfänger mithilfe der

im Satellitensignal enthaltenen Bahnparameter berechnet. Die Gleichung enthält drei Unbekannte (X_E, Y_E, Z_E) für die Position des Empfängers.

$$\rho = \sqrt{(X_E - X_{SV})^2 + (Y_E - Y_{SV})^2 + (Z_E - Z_{SV})^2} \quad (1)$$

Da bei der Laufzeitmessung des Satellitensignals zur Berechnung der geometrischen Entfernung ein Zeitfehler durch die Empfängeruhrenabweichung entsteht, wird dieser in der Beobachtungsgleichung (2) berücksichtigt.

$$C = \rho + c_0(dt + dT) + \varepsilon(C) \quad (2)$$

Dabei ist dt die Empfängeruhrenabweichung und dT die Satellitenuhrenkorrektur, welche ebenfalls im Satellitensignal übertragen wird. Diese wird benötigt, da auch die Atomuhren der Satelliten einer Ungenauigkeit unterliegen. Der Parameter ε dient zur Einbeziehung von Messrauschen und Mehrwegeeffekten. [3, p. 29] Setzt man Gleichung (1) in die Beobachtungsgleichung ein, erhält man vier Unbekannte, welche mithilfe der gemessenen Signallaufzeiten zu mindestens vier Satelliten gelöst werden.

1.1.1 Codemessung

Die Laufzeitbestimmung der Signale durch Codemessung (auch Pseudoentfernungsmessung) erfolgt im Empfänger bei der Dekodierung der Satellitensignale. Die eigentlichen übertragenen Daten wie u.a. Satellitenzeit, Ephemeriden (Satellitenlaufbahn) und Zeitkorrekturinformationen werden mit einer Datenrate von $50 \frac{\text{Bit}}{\text{s}}$ übertragen. Da alle Satelliten auf der gleichen Frequenz senden, die Signale jedoch im Empfänger wieder getrennt werden müssen, werden die Datenbits im Satelliten mit einem hochfrequenten PRN-Code (Pseudo Random Noise) überlagert, was zu einer Spreizung der Bandbreite führt. Dieses Verfahren wird als Code Division Multiple Access (CDMA) bezeichnet, wodurch Daten von verschiedenen Sendern zeitgleich und auf derselben Frequenz übertragen werden können. Der PRN-Code besteht aus einer scheinbar zufälligen Folge von Einsen und Nullen, wobei die Codes untereinander eine hohe Orthogonalität aufweisen, sodass diese nicht oder kaum miteinander korrelieren und die Signale im Empfänger einfacher getrennt werden können. Jeder Satellit besitzt einen spezifischen PRN-Code, durch welchen die vermischten Signale im Empfänger zum jeweiligen Satelliten zugeordnet werden können. Dies geschieht, indem die empfangenen Signale mit den im Empfänger generierten PRN-Codes korreliert werden. Zusätzlich zur Identifikation der Satelliten wird

durch die Korrelation auch die Laufzeit der Signale bestimmt, da beim Synchronisieren der selbst erzeugte PRN-Code schrittweise über den empfangenen Code geschoben wird, bis ein Korrelationsmaximum entsteht. Dieses Prinzip wird in Abbildung 1 gezeigt. Bezogen auf eine Zeitreferenz der Empfängeruhr kann durch diese Verschiebung auf eine Ankunftszeit des Signals geschlossen werden. Mit dem im Datensignal angebenen Sendezeitpunkt des ersten Bits einer Präambel, kann dann eine (Pseudo)-Signallaufzeit ermittelt werden, welche jedoch noch durch den Zeitfehler verfälscht ist. Dieser kann dann mathematisch eliminiert werden, da er in allen gemessenen Signallaufzeiten gleich groß ist. [4, pp. 49, 87] Allein mit der unkorrigierten Positionsmessung durch die Laufzeitbestimmung mithilfe von Codemessungen sind mit modernen GNSS-Empfängern Positionsgenauigkeiten von ca. 10 Metern möglich. [4, p. 4]

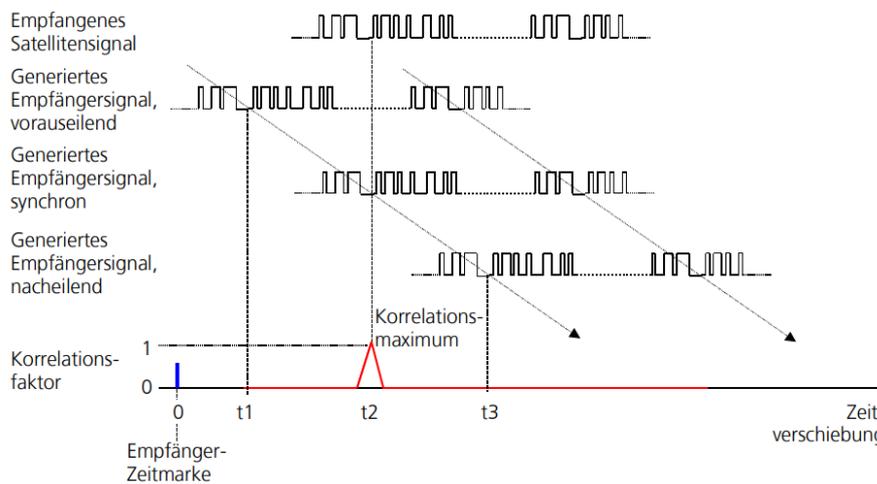


Abbildung 1 Prinzip der Codeverschiebung für die Korrelation [4, p. 87]

1.1.2 Phasenmessung

Für eine erhebliche Steigerung der Positionsgenauigkeit kann die Trägerphase der Satellitensignale ausgewertet werden. Im Gegensatz zu Codemessungen werden dabei die Abstände zwischen Empfänger und Satellit nicht über Signallaufzeiten bestimmt, sondern direkt über die Wellenlängen des Trägersignals. Die verwendeten Trägerfrequenzen liegen im L-Band zwischen 1 und 2 GHz, also deutlich höher als die Übertragungsrate des Daten- oder PRN-Codes. Die Wellenlängen betragen demzufolge nur ca. 19 cm. Durch die Messung der Phase mit der das Signal beim Empfänger eintrifft, kann man die Distanz mit einer deutlich höheren Genauigkeit bestimmen. Abbildung 2 zeigt das Prinzip der Trägerphasenmessung. Die Distanz D wird durch eine Anzahl an vollen Wellenzyklen $N \cdot \lambda$ sowie einen Bruchteil des letzten Zyklus $\frac{\varphi \cdot \lambda}{2\pi}$ berechnet, wobei λ die Wellenlänge und φ die Phase darstellt. Dieser letzte Bruchteil wird durch

eine Phasendifferenz zwischen dem Empfangssignal und der Frequenz eines Empfängeroszillators bestimmt und ist ausschlaggebend für die Zentimeter- bzw. Millimetergenauigkeit, die mit diesem Verfahren erreicht werden kann. [5, pp. 181-182] Problematisch ist dabei die unbekannte Anzahl an vollen Zyklen N der Trägerwelle zwischen Satellit und Empfänger, was zu einer Mehrdeutigkeit (Ambiguität) führt. „Durch Beobachtung mehrerer Satelliten zu verschiedenen Zeiten und durch ununterbrochenen Vergleich zwischen Anwender- und Referenzempfänger (während oder nach der Messung) kann nach dem Lösen von umfangreichen Gleichungssystemen die Position im Bereich von einigen Millimetern bestimmt werden.“ [4, p. 108] Der im Zitat erwähnte Referenzempfänger ist Teil der DGNSS-Technik und wird in Abschnitt 1.2 erläutert.

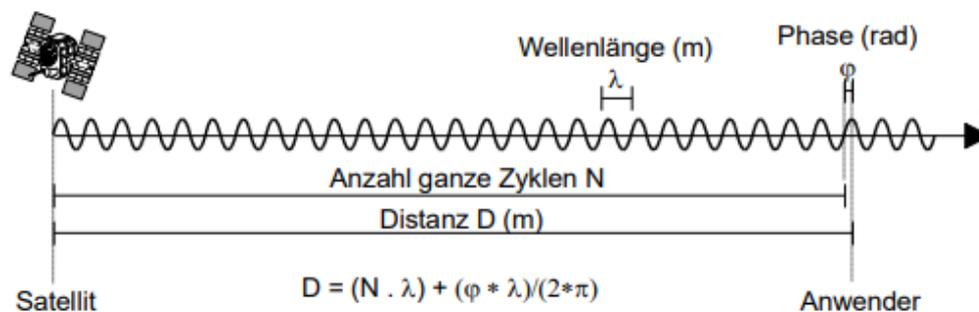


Abbildung 2 Trägerphasenmessung [4, p. 108]

1.1.3 Fehlereinflüsse

Die Standortbestimmung mittels GNSS-Signalen unterliegt sowohl bei Code- als auch bei Phasemessungen einer Vielzahl von Messfehlern, ohne deren Korrektur eine präzise Positionsmessung auch mit Auswertung der Trägerphase nicht möglich ist. Die schwerwiegendsten Fehlereinflüsse werden im Folgenden kurz erläutert, um den Nutzen der Korrekturmethode zur Erhöhung der Genauigkeit nachvollziehen zu können, welche im späteren Verlauf der Arbeit relevant sind.

Einen besonders großen Einfluss auf die Positionsgenauigkeit haben dispersive Effekte in der Ionosphäre, also eine frequenzabhängige Ausbreitungsgeschwindigkeit der Signale. Die resultierenden Fehler werden dabei vom Ionisationsgrad der Ionosphäre durch die Sonneneinstrahlung beeinflusst und variieren demzufolge tages- und jahreszeitenabhängig. Moderne Zweifrequenzempfänger können diesen Fehler größtenteils eliminieren. Auch in der Troposphäre, wo sämtliche Wettererscheinungen stattfinden, wirken sich Luftdruck, Temperatur und Wassergehalt auf den Brechungsindex und damit auf die Ausbreitungsgeschwindigkeit aus. Zwar sind

diese Effekte im Gegensatz zur Ionosphäre nichtdispersiv, können aber durch sich schnell wechselnde Bedingungen nur schwer als Fehlermodell modelliert werden. Durch differentielle Verfahren, welche im nächsten Kapitel erläutert werden, können wetterbedingte Messfehler jedoch stark reduziert werden. [5, pp. 129-140]

Weitere Fehlergrößen stellen Positionsfehler der Satelliten dar, die von ihren geplanten Umlaufbahnen (Ephemeriden) abweichen. Zudem unterliegen auch die hochgenauen Atomuhren in den Satelliten einer Drift, was trotz regelmäßiger Synchronisation durch Bodenstationen Zeitfehler im Bereich von einigen Nanosekunden verursachen kann. Weiterhin entstehen im Satelliten instrumentelle Signalverzögerungen (Code- und Phasenbiases). Fehlereinflüsse dieser Art können durch die Übertragung von präzisen Korrekturdaten als Modellierungen im Zustandsraum an den Empfänger berücksichtigt werden, was in einem späteren Kapitel erläutert wird. [6, p. 299]

Zuletzt sind einige zufällige Fehler zu nennen, deren Auswirkung zum Teil durch das Empfängerequipment bzw. den Messstandort beeinflusst werden. Zum einen stellt der Mehrwegeempfang eine große Gefahr für Messfehler dar. Durch Signalreflektionen an Objekten, welche sich in der Nähe des Empfängers befinden, können störende Interferenzen an der Antenne entstehen. Zwar sind GNSS-Signale durch das Frequenzspreizverfahren gegen Mehrwegeempfang gut geschützt, jedoch sollte für eine präzise Positionsbestimmung stets ein Standort mit ausreichendem Abstand zu Objekten mit glatten Oberflächen (z.B. Hausfassaden, Fahrzeuge, Gewässer) gewählt werden. [5, p. 142] Moderne Präzisionsantennen, wie beispielsweise Choke-Ring-Antennen, bieten zusätzlichen Schutz gegen Mehrwegeeffekte. Bei der Auswahl des Messstandortes sollte zudem darauf geachtet werden, Signalabschattungen durch große Objekte zu vermeiden. Die Genauigkeit der Positionsbestimmung ist u.a. abhängig von der aktuellen Satellitenkonstellation. Die empfangenen Satelliten sollten sich möglichst weit auseinander und auf einem niedrigen Elevationswinkel befinden. Abbildung 3 zeigt links eine bessere Satellitengeometrie als im rechten Teil. Durch Signalabschattungen wird eine optimale Satellitenkonstellation verhindert. Da moderne GNSS-Empfänger Satelliten von mehreren Positionierungssystemen gleichzeitig empfangen können, wird dieses Problem durch die generell größere Verfügbarkeit an Satelliten gemindert.

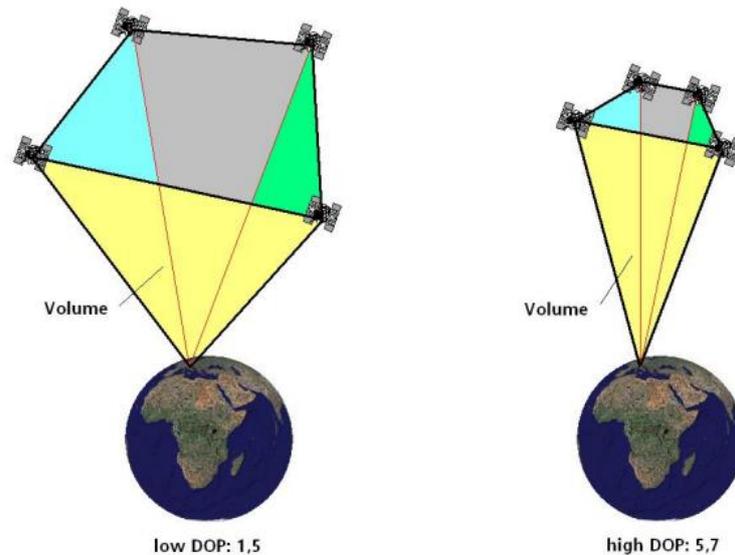


Abbildung 3 Gute (links) und schlechte (rechts) Satellitenkonstellation [4, p. 93]

1.2 Differentielles GNSS

Das Konzept des differentiellen GNSS dient zur Genauigkeitssteigerung von Positionsmessungen durch die präzise Bestimmung relativer Abstände zwischen GNSS-Empfängern und ist ein Ergebnis aus der Zeit der Selective Availability (SA), welches eine Abwehrmaßnahme der US-Regierung von 1990 bis 2000 bezeichnet. Mit der SA wurden die L1-Signale der GPS-Satelliten damals künstlich verfälscht, um eine Bedrohung durch einen militärischen Einsatz des Systems von Feinden zu verhindern. Somit wurde die Positionsgenauigkeit des zivilen L1-Bands auf ca. 100 Meter begrenzt. Um diese künstliche Verfälschung bei zivilen Anwendungen zu umgehen, wurden ortsfeste GPS-Referenzstationen hochgenau vermessen. „Durch Vergleich mit dem tatsächlichen und dem von den Referenzempfängern errechneten Standort kann man auf den Positionsfehler schließen. Diesem Fehler unterliegen auch die errechneten Positionen anderer Empfänger in der Umgebung der Referenzstationen. Wenn man diesen Fehler an die umliegenden Empfänger kommuniziert, dann können diese den Fehler aus ihren Ergebnissen herausrechnen und somit selbst Positionen mit einer sehr hohen Genauigkeit liefern.“ [2, p. 36] Auch wenn die künstliche Verschlechterung im Jahr 2000 wieder abgeschaltet wurde, kann die DGNSS-Technik genutzt werden, um einige der zuvor genannten Messfehler zu reduzieren.

Die Übermittlung der von einer Referenzstation ermittelten Messfehler in Form von Korrekturdaten kann über verschiedene Kommunikationswege erfolgen. Die heute am gebräuchlichste Form ist die Übertragung mittels Networked Transport of RTCM via Internet Protocol (Ntrip),

wobei die in einer RTCM-Nachricht (Erläuterung in Abschnitt 3.2.2) verpackten Korrekturdaten über das Internet übertragen werden. Zudem werden die Daten in vielen Anwendungen auch per Satellit auf dem L-Band übertragen.

1.2.1 Absolute und relative Positionsbestimmung

Zum besseren Verständnis der DGNSS-Technik muss zwischen absoluter und relativer Positionsbestimmung unterschieden werden. Im Grunde ist jede Positionierung relativ, da man stets ein Bezugssystem braucht, auf das sich die gemessenen Koordinaten beziehen. Jedoch spricht man von absoluter Positionierung, wenn der Standort eines *einzelnen* Empfängers direkt anhand der empfangenen Satellitensignale auf Basis von Code- oder Phasenbeobachtungen berechnet wird. Die errechnete Position wird dann in einem Koordinatensystem mit dem Erdmittelpunkt als Bezugspunkt angegeben, sodass der Anwender weiß, wo genau sich der Empfänger auf der Erde befindet. Dabei ist die Verwendung von Korrekturdaten für eine Genauigkeitssteigerung möglich, jedoch nicht zwingend erforderlich. Anwendungsbeispiele dafür können Positionsbestimmungen für Fahrzeuge, Schiffe oder Smartphones sein.

Bei der relativen Positionsmessung sind mindestens *zwei* GNSS-Empfänger erforderlich, wobei jeweils einer als – in den meisten Anwendungen – ortsfeste Referenzstation dient und Korrekturdaten durch eine Differenzbildung zwischen tatsächlicher und gemessener Position bereitstellt. Der andere Empfänger, Rover genannt, berechnet anhand dieser Korrekturdaten dessen Position bezogen auf den Standort des Referenzempfängers in Form eines Basislinienvektors. Durch die Addition dieses Vektors zu den bekannten Koordinaten der Referenzposition ergibt sich dann die absolute Position des Rovers. Auch bei der relativen Positionierung können sowohl Codemessungen als auch Phasenmessungen verwendet werden. [3, p. 30] Dabei ist zu beachten, dass zwar die relative Position zwischen beiden Empfängern besonders mithilfe der Phasenmessung durch die Differenzbildung stets genau ermittelt werden kann, sich jedoch dementsprechend ein Fehler in der absoluten Position des Referenzempfängers im selben Maße auf die Genauigkeit der absoluten Position des Rovers auswirkt. Bei einigen Anwendungen ist die absolute Position – also die Koordinaten auf einer Karte – gar nicht von Interesse, sondern nur der relative Basislinienvektor zwischen zwei Empfängern. Ein Beispiel dafür ist im folgenden Kapitel beschrieben.

1.2.2 Echtzeitkinematik

Eine Vielzahl moderner GNSS-Empfänger verfügen über die sogenannte Real-Time-Kinematic-Technik (RTK), welche auf dem zuvor beschriebenen Prinzip des differentiellen GNSS basiert. Dabei findet mithilfe der Korrekturdaten des Referenzempfängers (Basis) eine Trägerphasenauswertung im Rover in Echtzeit statt, sodass ein Basislinienvektor zwischen den Empfängern im Genauigkeitsbereich von wenigen Zentimetern berechnet werden kann. Durch die Lösung der Phasenmehrdeutigkeiten in Echtzeit kann sich der Rover-Empfänger auf einer mobilen Plattform wie beispielsweise einer Drohne befinden. Im RTK-Betrieb kann zwischen zwei Betriebsmodi- bzw. Status unterschieden werden: RTK-float und RTK-fixed. Befindet sich der Empfänger im RTK-fixed-Modus, konnten die Phasenmehrdeutigkeiten ganzzählig gelöst werden, sodass eine Zentimetergenauigkeit erreicht werden kann. Im RTK-float-Modus werden die Ambiguitäten nur angenähert geschätzt, sodass nur eine Genauigkeit im Bereich von 20 cm oder mehr erreicht werden kann. Kann der RTK-fixed-Status nicht erreicht werden, ist dies oftmals auf eine zu geringe Anzahl an verfügbaren Satelliten durch Signalabschattungen zurückzuführen.

Die relative Positionierung beruht mathematisch betrachtet auf der Differenzbildung der Code- oder Phasenbeobachtung zweier GNSS-Empfänger zu einem gemeinsamen Satelliten. „Bei der relativen Positionierung mit Phasendaten kann zwischen einfachen Differenzen und doppelten Differenzen unterschieden werden. Einfache Phasendifferenzen [...] bilden sich, wie schon bei den Codebeobachtungen, aus den von zwei Empfängern simultan beobachteten Signalen eines Satelliten. Bei den doppelten Phasendifferenzen [...] werden aus den einfachen Empfänger-differenzen die Differenzen zu zwei Satelliten gebildet. Durch diese doppelte Differenzbildung wird zusätzlich die Empfängeruhrabweichung eliminiert. Dadurch gelingt es, dass die Mehrdeutigkeiten auf ganze Zahlen festgesetzt werden (*fixed*-Lösung).“ [3, pp. 30-31] Beide Empfänger, Basis und Rover, müssen für den differentiellen Betrieb dementsprechend freie Sicht auf dieselben Satelliten haben. Abbildung 4 und Abbildung 5 veranschaulichen das Prinzip der einfachen und doppelten Differenzierung, mit welcher es möglich ist, den Satelliten- und Empfängeruhrenfehler rechnerisch zu eliminieren.

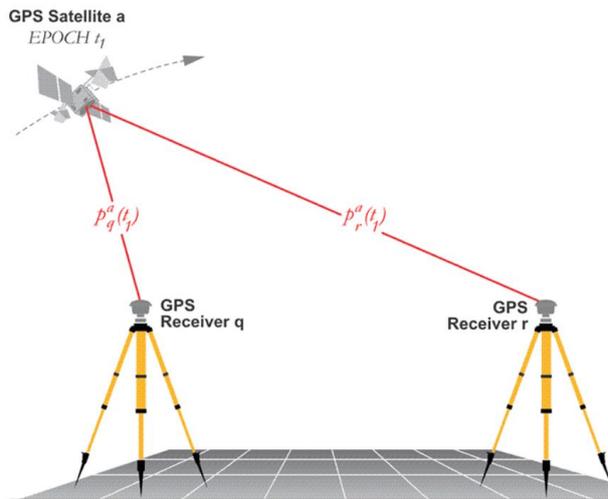


Abbildung 4 Einfach Differenzierung [7]

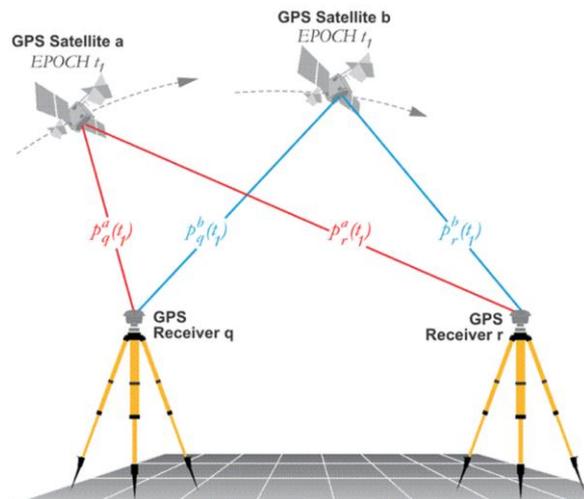


Abbildung 5 Doppelte Differenzierung [7]

Zu unterscheiden sind zwei Konzepte der relativen Positionierung. Das ursprüngliche Prinzip des DGNSS, dass ein Referenzempfänger fest an einem Standort mit *bekannt* absoluten Koordinaten verbleibt und somit die Gemessene mit der tatsächlichen Position vergleichen kann, dient dazu, den mobilen Empfängern ebenfalls die Berechnung einer absoluten Position mit hoher Genauigkeit zu ermöglichen.

Es ist jedoch auch möglich, mithilfe der doppelten Differenzierung präzise relative Positionen zwischen zwei Empfängern zu bestimmen, wenn die absolute Position des Referenzempfängers nicht mit hoher Genauigkeit bekannt ist. Ein Beispiel dafür sind GNSS-gestützte Landeanflüge von Flugobjekten auf bewegliche Landeplattformen, wobei sich der Referenzempfänger sogar in Bewegung befindet. Die absolute Position der Landeplattform ist dabei nicht bekannt, sondern lediglich der relative Basislinienvektor zwischen Referenz und Rover. [8] Dieses Konzept wird später für die GNSS-Richtungsbestimmung der Messdrohne relevant. Ein weiteres Beispiel, bei dem der Referenzempfänger zwar fixiert ist, die absoluten Positionen beider Empfänger jedoch irrelevant sind, sind satellitengesteuerte Mähroboter, wobei eine eigene GNSS-Referenzstation in der Nähe der Rasenfläche fest installiert wird und mit dem mobilen Empfänger auf dem Mähroboter kommuniziert. Dabei speichert der Roboter lediglich die relativen Positionen zur Referenzstation ab, die beim händischen Abfahren der Rasenflächengrenzen gemessen werden.

1.2.3 Korrekturverfahren

Erfordert eine Anwendung die Ermittlung von absoluten Positionen, bei denen die Genauigkeit einer unkorrigierten Einzelpunktbestimmung nicht ausreicht, können Korrekturdaten mithilfe verschiedener Konzepte zugeführt werden, um die zuvor erwähnten Messfehler zu minimieren.

Zum einen können die Daten mit dem bereits erläuterten differentiellen Verfahren durch Differenzbildungen erzeugt und dem Empfänger bereitgestellt werden. Dabei spricht man von der Modellierung im Beobachtungsraum (OSR - Observation Space Representation). Differentielle Korrekturdaten können sowohl von einer selbst aufgestellten Referenzstation, als auch von externen Korrekturdiensten erzeugt werden. Diese Korrekturdienste verfügen zumeist über ein Netz von Referenzstationen, um große Einsatzgebiete zu ermöglichen. Die Übertragung kann einerseits terrestrisch beispielsweise über Rund- oder Mobilfunk erfolgen, und andererseits auch über geostationäre Satelliten auf dem L-Band. Diese Dienste ermöglichen u. a. die korrigierte GNSS-Positionierung in Echtzeit. [4, pp. 109-111]

Zum anderen spricht man bei GNSS-Korrekturen von einer Modellierung im Zustandsraum (SSR – State Space Representation), wenn es sich um eine „autonome Einzelpunktbestimmung des Rovers“ [6, p. 299] handelt. Dabei werden dem Empfänger u.a. sehr präzise Ephemeriden- und Uhrenkorrekturen, sowie Modellierungen von ionosphärischen und troposphärischen Einflüssen übermittelt. Es bedarf bei dieser Methode prinzipiell also keiner zusätzlichen Referenzstation, jedoch werden bei Echtzeitanwendungen oftmals dennoch Referenzstationsnetze hinzugezogen. [6, p. 299] Die Technik, die dieses Verfahren nutzt, wird als Precise Point Positioning (PPP) bezeichnet. Im Gegensatz zur Differenzbildung beim DGNSS werden Fehlerquellen dabei auf Grundlage von Beobachtungen global verteilter Referenzstationen einzeln modelliert. [9, p. 242]

1.3 Koordinatensysteme

Die Angabe von Positionsdaten kann in verschiedenen Formaten und Bezugssystemen erfolgen. Die in dieser Arbeit verwendeten Formate werden nachfolgend erläutert, um die Messergebnisse nachvollziehen zu können.

1.3.1 ECEF-Koordinaten

Das ECEF-Format (Earth Centered Earth Fixed) ist ein geozentrisches, kartesisches Koordinatensystem, in welchem die Positionen von GPS-Satelliten und GPS-Empfängern bestimmt werden. Der Koordinatenursprung befindet sich im Massezentrum der Erde bzw. im Zentrum des jeweiligen verwendeten Referenzellipsoids. Dabei ist die Z-Achse zum Nordpol gerichtet, die X-Achse befindet sich auf der Schnittgeraden von Äquatorebene und Nullmeridian und die Y-Achse steht rechtwinklig zur X-Achse und Z-Achse. [14] Die Positionsangabe erfolgt dann jeweils mithilfe eines Ortsvektors, dessen Elemente in Meter angegeben werden. In GNSS-Systemen wird standardmäßig der WGS84-Referenzellipsoid zur Berechnung verwendet. Dieses weltweite mathematische Bezugssystem beschreibt die Form der Erde durch ein an den Polen abgeplattetes Rotationsellipsoid, welches durch Drehung der Meridianellipse (große Halbachse) um die kleine Halbachse entsteht. [10] Für jedes Bezugssystem bzw. Referenzellipsoid bestehen verschiedene Werte für die Längen der großen und kleinen Halbachse, mithilfe derer sich die verschiedenen Formate der Positionsangabe ineinander umrechnen lassen.

1.3.2 Ellipsoidische Koordinaten

Ein weiteres Darstellungsformat von Navigationsdaten sind ellipsoidische- bzw. geografische Koordinaten, welche in geografischer Länge λ und geografischer Breite ϕ angegeben werden. Dabei ist die geografische Länge der Winkel zwischen einem Punkt auf der Erdoberfläche und der Nullmeridianebene, und die geografische Breite der Winkel zwischen einem Punkt und der Äquatorebene. Die Winkel können im Format Grad-Minute-Dezimalsekunde (z.B. $11^\circ 33' 11,180''$) oder nur in Dezimalgrad (z.B. $11,55310556^\circ$) erfolgen, mit dem jeweiligen Richtungszusatz Nord oder Süd und Ost oder West, da die Gradangabe nur bis 180° ausgehend von Äquatorebene und Nullmeridian erfolgt. [11] In GNSS-Empfängern werden dann südliche oder westliche Positionen oft mit einem negativen Dezimalgrad dargestellt. Für die Angabe der Höhe gibt es verschiedene Möglichkeiten, jedoch ist in dieser Arbeit nur die ellipsoidische Höhe –

also der Abstand zwischen einer Position und der mathematischen Oberfläche des Referenzellipsoides – von Bedeutung. Kartesische und Ellipsoidische Koordinaten können ineinander umgerechnet werden. Eine Beispielrechnung wird in Abschnitt 4.2.3 durchgeführt.

1.3.3 NED-Koordinaten

NED-Koordinaten (North, East, Down) werden benötigt, wenn Geschwindigkeiten und relative Positionen in einem lokalen Koordinatensystem ausgedrückt werden sollen. Dabei stellt die aktuelle Position eines Referenzempfängers den Koordinatenursprung dar. Die Abwärts-Achse zeigt dabei in Richtung Erdmittelpunkt. Nord- und Ost-Achse spannen dementsprechend eine Tangentialebene zur Erdoberfläche auf, wenn sich die Position auf der Erdoberfläche befindet. Abbildung 6 veranschaulicht den Aufbau der Koordinatensysteme. [12] Die grünen Pfeile stellen das NED-Koordinatensystem dar.

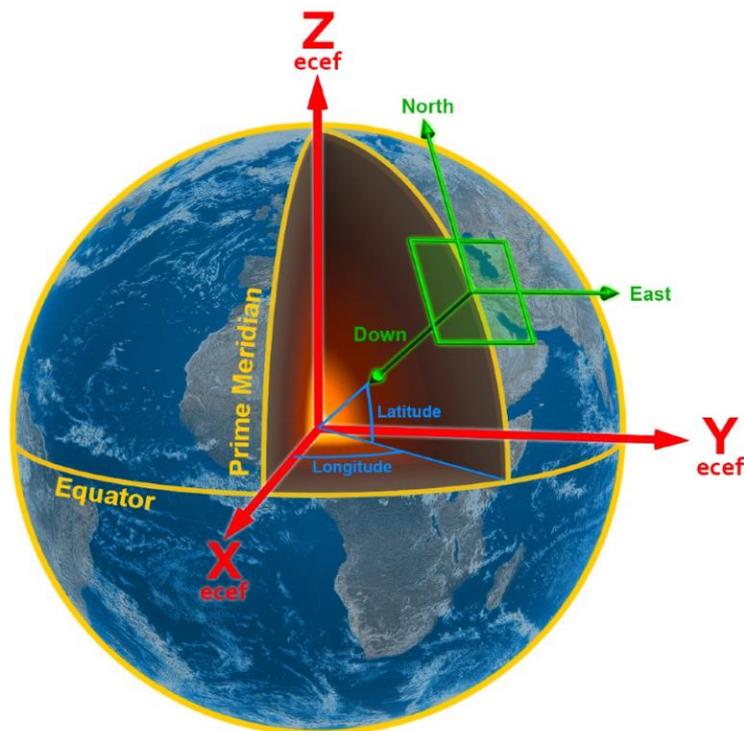


Abbildung 6 Darstellung der verschiedenen Koordinatensysteme [12]

2 Konzept

In diesem Kapitel werden die in dieser Arbeit zu untersuchenden Konzepte für die GNSS-Positions- und Richtungsbestimmung der HF-Messdrohnen auf Grundlage der Erläuterungen in den theoretischen Grundlagen vorgestellt.

2.1 GNSS-Kompass

Das Prinzip des differentiellen GNSS kann wie bereits beschrieben mit RTK-Technologie genutzt werden, um die relativen Positionen unter Bildung eines Basislinienvektors zwischen zwei GNSS-Antennen präzise in Echtzeit zu ermitteln. Standardmäßig verbleibt der Referenzempfänger (Basis) beim DGNSS an einem festen Punkt mit bekannten Koordinaten und sendet einen kontinuierlichen Korrekturdatenstrom mit Code- oder Phasenbeobachtungen an den mobilen Empfänger (Rover). Ziel dieses Konzeptes ist die präzise Bestimmung der absoluten Position des Rovers mit Referenz zur absoluten Position der Basis. Mit dem relativen Basisvektor von der Basis-Antenne zur Rover-Antenne kann jedoch auch auf eine Himmelsrichtung, bezogen auf geografisch Nord, geschlossen werden, was anders ausgedrückt einem fixierten Kompass entsprechen würde.

Für die Bestimmung der „Blickrichtung“ der Messdrohnen müssen sich jedoch beide Empfänger auf der Drohne befinden. Auf einen feststehenden Basisempfänger muss demzufolge verzichtet werden. Da dann kein Abgleich zwischen gemessener und tatsächlicher Position mehr stattfinden kann, geht die Zentimetergenauigkeit in der absoluten Position der Empfänger zunächst verloren und fällt auf die Genauigkeit von ca. 5-10 Metern einer unkorrigierten Einzelpunktbestimmung zurück. Die präzise Berechnung eines Basislinienvektors ist jedoch mithilfe des Differenzbildungsverfahrens auch mit einem beweglichen Referenzempfänger möglich. Dementsprechend lässt sich durch das Anbringen zwei miteinander verbundener GNSS-Empfänger- und Antennen auf der Drohne ein mobiler GNSS-Kompass zur Kursmessung umsetzen. [13, p. 5] Dabei ist die Genauigkeit der Richtungsangabe sowohl vom Abstand der Antennen als auch von der verfügbaren Satellitenkonstellation abhängig. Je größer der Abstand (Basislinie) zwischen den Antennen ist, desto weniger Einfluss hat ein Fehler in der relativen Position auf die berechnete Richtung.

Abbildung 7 zeigt den Antennen-Messaufbau eines GNSS-Kompasses für die Fahrtrichtungsbestimmung eines Autos. Dabei befindet sich die Antenne der beweglichen Basis (Referenz) am Heck- und die Rover-Antenne an der Front des Fahrzeugs auf einer horizontalen Achse

längs zur Fahrtrichtung. Die Empfänger sind miteinander verbunden, sodass der Referenzempfänger RTCM-Korrekturdaten an den Rover senden kann. Mit den horizontalen Anteilen des Basislinienvektors (roter Pfeil) kann ausgehend von der Position der Referenzantenne eine Richtung berechnet werden.

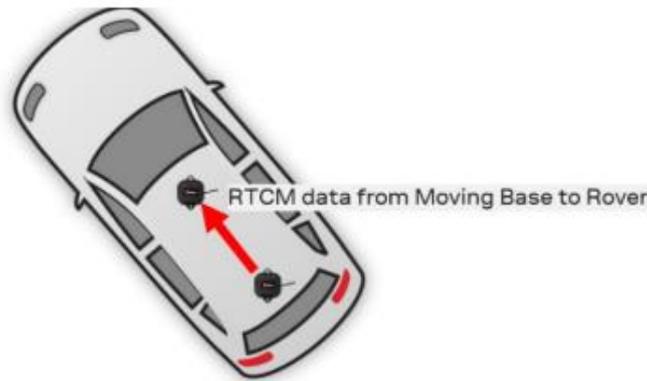


Abbildung 7 Antennen-Messaufbau für eine Richtungsbestimmung auf einem Autodach [13, p. 7]

Auf der Drohne würden die Antennen dementsprechend befestigt werden, dass die Achse auf der sich die Antennen befinden, horizontal und längs zur Blickrichtung der Messantenne ausgerichtet ist. Zudem ist darauf zu achten, die Antennen möglichst am höchsten Punkt der Drohne zu positionieren und ggf. ein Metallblech unter den Antennen anzubringen, um diese vor indirekten Signalen zu schützen, welche unter den Antennen reflektiert wurden. [13, pp. 9-10]

2.2 Positionierung mit eigener Basisstation

Mit dem zuvor beschriebenen Konzept des GNSS-Kompasses lässt sich ohne Weiteres eine absolute Positionierung nur mit der Standardgenauigkeit der GNSS-Systeme erreichen. Sollte diese Genauigkeit für die Messanwendungen der Drohnen im PMD nicht ausreichen, kann der bewegliche Referenzempfänger auf der Drohne ebenfalls mit Korrekturdaten versorgt werden. Da die GNSS-Empfänger für die Kompassanwendung ohnehin RTK-fähig sein müssen, ist es naheliegend, ebenfalls differentielle Korrekturen für den Referenzempfänger zu verwenden. In Abbildung 8 ist das Konzept für eine präzise Richtungs- und Positionsbestimmung dargestellt, wobei das Konzept aus Abbildung 7 um einen weiteren Korrekturdatenstrom erweitert wurde.

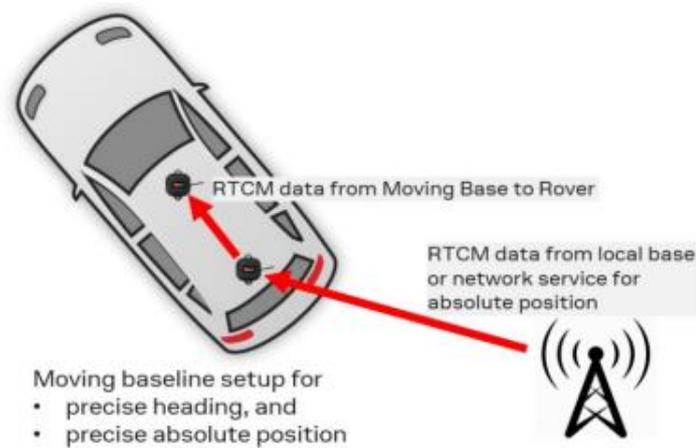


Abbildung 8 GNSS-Kompass mit präziser absoluter Positionierung [13, p. 5]

Die Korrekturdaten für die absolute Positionierung kann über verschiedene Wege erzeugt bzw. abgerufen werden. Die erste Methode wurde bereits in einer vorangegangenen Arbeit erprobt. Dabei wird eine eigene Referenzstation aufgestellt, welche dann als fester Bezugspunkt dient und differentielle Korrekturdaten erzeugt und bereitstellt. Der Datenstrom vom festen Referenzempfänger zur beweglichen Basis auf der Drohne erfolgt über einen geeigneten Funkstandard wie Bluetooth, WLAN oder LoRa. [1, p. 25]

Dieses Konzept vermeidet die Notwendigkeit eines externen Korrekturdienstes, welche oft kostenpflichtig sind. Zudem werden die Korrekturdaten dieser Dienste zumeist per Ntrip-Protokoll über das Internet übertragen, sodass stets eine Mobilfunkverbindung zur Drohne bestehen muss. Bei einer schlechten Netzabdeckung könnte dies zu Problemen führen. Die Schwierigkeit eines eigenen Referenzempfängers für die absolute Positionierung liegt darin, die exakten Koordinaten des Antennenstandortes der Referenzstation zu ermitteln, da es sich dabei wiederum um eine Einzelpunktbestimmung handelt. Die Genauigkeit dieser Einzelpunktbestimmung ist wiederum ausschlaggebend für die Positionsgenauigkeit der Drohne. Auch wenn die relative Position zwischen Drohne und festem Referenzempfänger im Genauigkeitsbereich von wenigen Zentimetern liegt, wirkt sich ein Positionsfehler des Referenzempfängers im gleichen Maße auf die Position der Drohne aus. Eine Möglichkeit, die Referenzstation einzumessen, indem Positionswerte aus Codemessungen über einen längeren Zeitraum gesammelt und gemittelt werden, wird in Abschnitt 4.2.3 untersucht.

2.3 Positionierung ohne eigene Basisstation

Die zweite Methode, den beweglichen Referenzempfänger auf der Drohne mit Korrekturdaten zu versorgen, besteht in der Nutzung eines Korrekturdienstes. Die verschiedenen Korrekturverfahren wurden bereits in Abschnitt 1.2.3 kurz erläutert. Im Folgenden werden die in der Erprobung verwendeten Korrekturdienste vorgestellt.

2.3.1 SBAS

Bei den Satellite Based Augmentation Systems (dt.: satellitengestützte Erweiterungssysteme) werden differentielle Korrekturdaten von geostationären Satelliten auf dem L1-Band ausgesendet, welche von GNSS-Empfängern für eine erhöhte Positionsgenauigkeit genutzt werden können. Die Korrekturdaten werden durch ein großflächig angelegtes Netz aus Bodenstationen erzeugt und an die geostationären Satelliten gesendet, welche die Daten über einen Transponder wieder zur Erde zurücksenden. SBAS ist ein Sammelbegriff für mehrere existierende Systeme, welche jeweils ein bestimmtes Gebiet der Erde abdecken. Das für Europa verfügbare System heißt EGNOS, ist kostenfrei verfügbar und kann durch jeden L1-Empfänger empfangen werden. Mit den Daten werden Information für die Korrektur der Ionosphären-, Ephemeriden- und Uhrenfehler übermittelt. Zusätzlich verfügen die Dienste über eine Integritätsprüfung für sicherheitskritische Anwendungen. Dabei werden die Signale der GNSS-Satelliten überwacht und im Falle einer Störung wird dies den Empfängern innerhalb von sechs Sekunden mitgeteilt. Nach [4, p. 122] liegt der typische horizontale Fehler mit dem EGNOS-System bei 1,84 m (2σ). Der Vorteil dieses Dienstes ist, dass die Korrekturdaten mit modernen GNSS-Empfänger ohne zusätzliche Hardware empfangen werden können und durch die Übertragung per Satellit auch eine hohe Verfügbarkeit besteht. Die Genauigkeit der absoluten 3D-Position ist jedoch auf mehrere Meter beschränkt. [4, pp. 113-122]

2.3.2 SAPOS

SAPOS ist ein Korrekturdienst der deutschen Landesvermessung, welcher differentielle Korrekturdaten von bundesweit verteilten Referenzstationen bereitstellt. Es sind mehrere Dienste mit unterschiedlichen Genauigkeitsstufen verfügbar. Für den hochpräzisen Echtzeit Positionierungs-Service (HEPS) wird eine Genauigkeit von 1-2 cm in der Ebene und 2-3 cm in der Höhe angegeben. Die Daten werden im RTCM-Format mit Ntrip über das Internet übertragen. Mittlerweile können alle verfügbaren Systeme (GPS, Galileo, GLONASS und BeiDou) verwendet

werden. Leider werden die Dienste auf Länderebene verwaltet, sodass keine einheitlichen Nutzungsbedingungen existieren. Bis auf Bayern, Rheinland-Pfalz, Saarland und Schleswig-Holstein steht der HEPS-Service kostenfrei zu Verfügung (Stand Januar 2024). Für den Bedarf einer bundesweiten Nutzung gibt es eine zentrale SAPOS-Stelle der Landesvermessung Niedersachsen, wobei diese in jedem Fall gebührenpflichtig ist. [14]

Nachteilig ist die Voraussetzung, dass die Drohne stets eine Verbindung zum Internet haben muss. In Gebieten mit schlechter Netzabdeckung könnte dies problematisch sein.

2.3.3 Point-Perfect

Point-Perfect ist ein Korrekturdienst der Firma u-blox. Dieser basiert auf dem noch recht neuen Verfahren des PPP-RTK, was eine Kombination aus differentiellen Korrekturen (OSR) und Fehlermodellierungen (SSR) darstellt. Der Nachteil des HEPS von SAPOS ist, dass eine bidirektionale Verbindung zwischen Korrekturserver und Empfänger bestehen muss, was für zukünftige Massenmarktanwendung, wie beispielsweise autonomes Fahren, ungeeignet ist. Im Gegensatz dazu können die Korrekturdaten der PPP-Dienste per unidirektionaler Broadcastübertragung übermittelt werden und damit auch mit Übertragungsmedien wie zum Beispiel DAB+ oder Satellitenübertragung. Mit PPP-Korrekturen allein können aktuell jedoch nicht die Konvergenzzeiten der DGNSS-Verfahren erreicht werden. PPP-RTK-Dienste vereinen dementsprechend die Vorteile der hohen Genauigkeit und der dateneffizienten Broadcastübertragung. [9, p. 244]

Point-Perfect ist grundsätzlich kostenpflichtig, jedoch bestand durch die Beschaffung der u-blox-Empfänger die Möglichkeit, den Dienst kostenfrei zu testen. Das Unternehmen wirbt mit einer horizontalen Positionsgenauigkeit von 3-6 cm in Echtzeit. Zudem können die Daten unter der Verwendung eines zusätzlichen Empfängers über das L-Band übertragen werden. Wenn sich die angegebenen Genauigkeiten in der Erprobung bestätigen lassen, wäre dieser Dienst für die Drohnenmessungen gut geeignet, da das Problem einer unzureichenden Netzabdeckung umgangen wird.

3 Equipment

In den folgenden Abschnitten werden alle für die praktische Erprobung relevanten Informationen bzgl. der verwendeten Hardware und Möglichkeiten zum Konfigurieren und Auslesen der GNSS-Empfänger behandelt. Das Kapitel beinhaltet eine Kurzanleitung für die Evaluationssoftware u-center sowie eine Beschreibung des für die Testmessungen verwendeten Python-Codes.

3.1 Empfänger und Antennen

Der in dieser Arbeit erprobte ZED-F9P-Chip der Firma u-blox ist ein low-cost Multiband GNSS-Präzisionsempfänger. Mit diesem ist es möglich, die Satellitensysteme NAVSTAR GPS, GLONASS, Galileo und Beidou gleichzeitig und auf verschiedenen Frequenzbändern zu empfangen. Dazu verfügt der Empfänger über 184 Kanäle, in welchen die Satellitensignale dekodiert werden. Zusätzlich kann der Empfänger unterschiedliche Korrekturdaten aus Modellierungen sowohl im Beobachtungsraum (OSR), als auch im Zustandsraum (SSR) empfangen und verarbeiten. [15, p. 7] Sämtliche Berechnungen von Navigationsdaten (z. B. Position, Richtung, Geschwindigkeit) aus den Entfernungsrohdaten werden vom F9P selbst getätigt, es bedarf dafür also keiner externen Hard- oder Software. Lediglich für den korrekturgestützten Betrieb mit NTRIP wurde die Evaluationssoftware u-center verwendet, um eine Verbindung zwischen Empfänger und NTRIP-Server herzustellen.

Der ZED-F9P verfügt zudem über einen „Moving-Base-Modus“, durch welchen es möglich ist, die relative Position zwischen einem Roverempfänger und einem Basisempfänger zentimetergenau zu berechnen. Dabei können sich beide Empfänger – auch relativ zueinander – in Bewegung befinden. Diese Funktion ist beispielsweise nützlich für Anwendungen, bei denen eine Drohne einem anderen Objekt folgen soll, wobei der Abstand zwischen Basis und Rover also variabel ist. Eine weitere Anwendungsmöglichkeit ist der in dieser Arbeit zu untersuchende GNSS-Kompass zur Richtungsbestimmung, bei dem sich Rover und Basis auf einem Fahrzeug und in festem Abstand zueinander befinden. Dessen Funktionsweise wurde in Abschnitt 2.1 erläutert. Darüber hinaus können auch mehrere Rover gleichzeitig verwendet werden, um die Orientierung eines Luftfahrzeugs (Roll, Pitch, Heading) zu messen. [15, p. 7]

Für die Testmessungen wurden drei EVK-F9P-Evaluation-Kits der Firma u-blox beschafft. Diese enthalten zum einen eine in ein Metallgehäuse eingebaute Testplatine (Evaluationsboard), auf welche der F9P-Empfängerchip integriert ist (siehe Abbildung 9). Auf der Platine

sind verschiedene Schnittstellenanschlüsse und Komponenten verbaut, welche in Abbildung 10 dargestellt sind. Aufgrund der verschiedenen Anschlussmöglichkeiten, welche die Platine flexibel für unterschiedliche Testmessungen macht, und wegen des robusten Gehäuses wurde anstatt einer kompakten Platine eines anderen Herstellers zunächst dieses Evaluationsboard verwendet, auch wenn dieses für einen Anbau an eine Messdrohne aufgrund der Größe zunächst ungeeignet ist.

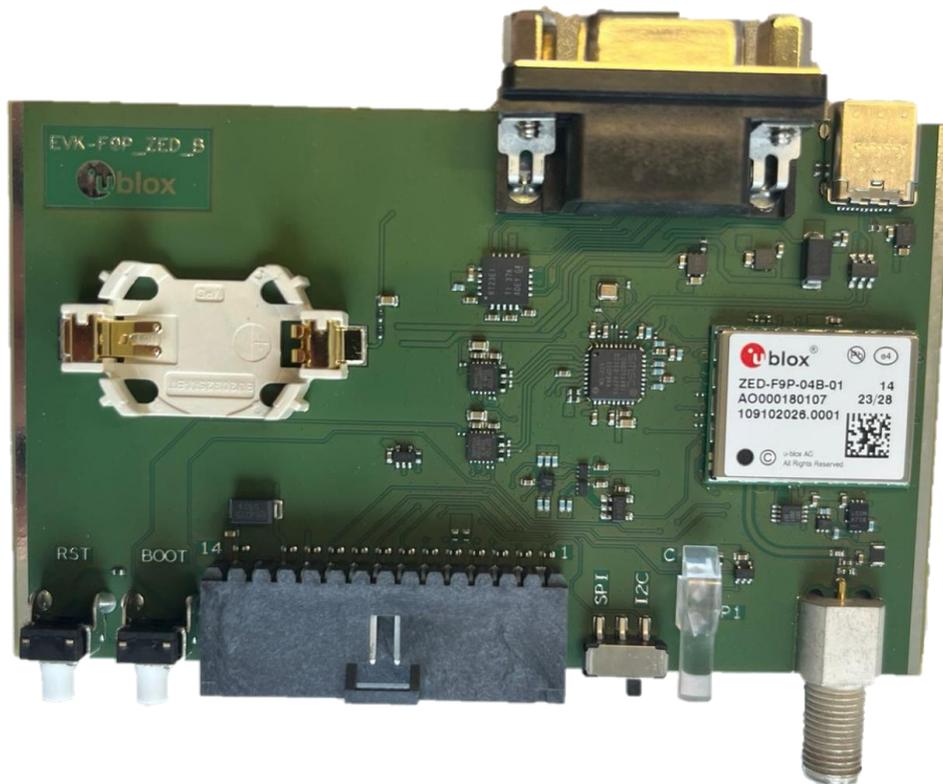


Abbildung 9 Platine des Evaluation-Kits

Die verwendeten Anschlüsse für die Erprobung der Empfänger in dieser Arbeit sind die seriellen Schnittstellen bzw. Übertragungsstandards UART1, UART2 und USB. Datenaustausch und Stromversorgung finden über ein USB-Kabel statt, welches die Verbindung zwischen Computer und Empfänger bildet. Durch den verbauten Hub können die UART-Ports auch über das USB-Kabel erreicht werden.

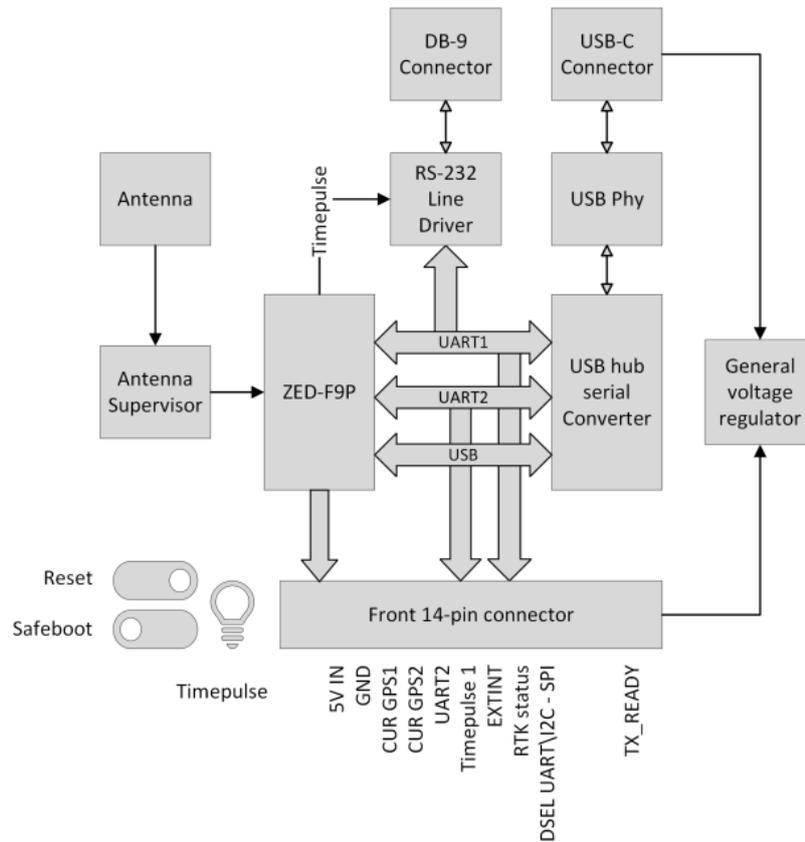


Abbildung 10 Schnittstellen und Verbindungen des Evaluation Boards [16]

Als Antennen dienen die im Evaluation Kit enthaltenen low-cost Multiband-Patchantennen ANN-MB-00 mit SMA-Anschluss. Diese unterscheidet sich von teuren Hochleistungsantennen im Wesentlichen in der höheren Anfälligkeit für Mehrwegeempfang und einer geringeren Empfindlichkeit des Signalempfangs. Bei der ANN-MB-Serie handelt es sich um eine aktive Antenne, welche mit SAW-Filtern zur Vorfilterung und rauscharmen Verstärkern (LNA) ausgestattet ist. LNAs sind vor allem dann von Vorteil, wenn sich zwischen Antenne und Empfänger ein dämpfendes Glied (z.B. ein Koaxialkabel) befindet, weil sich das Rauschen des ersten Bauteils am stärksten auf die Gesamtrauschleistung auswirkt. [17] Die SAW-Filter vermeidet die Aufnahme von Störsignalen und begrenzen die Empfangsbandbreite um die Aufnahme von zusätzlichem Rauschen zu verhindern. Gerade bei GNSS-Signalen, deren Signalpegel durch die Frequenzspreizung unterhalb des thermischen Rauschpegels liegen, ist eine Vorfilterung und rauscharme Vorverstärkung sinnvoll. [2, p. 29] In Abbildung 11 ist ein Blockschaltbild der ANN-MB-Antenne dargestellt. Da die GNSS-Signale zirkular polarisiert sind, werden dualgespeiste Patches verwendet, wobei die Zuleitungen dann über einen Hybridkoppler zusammengeführt werden.

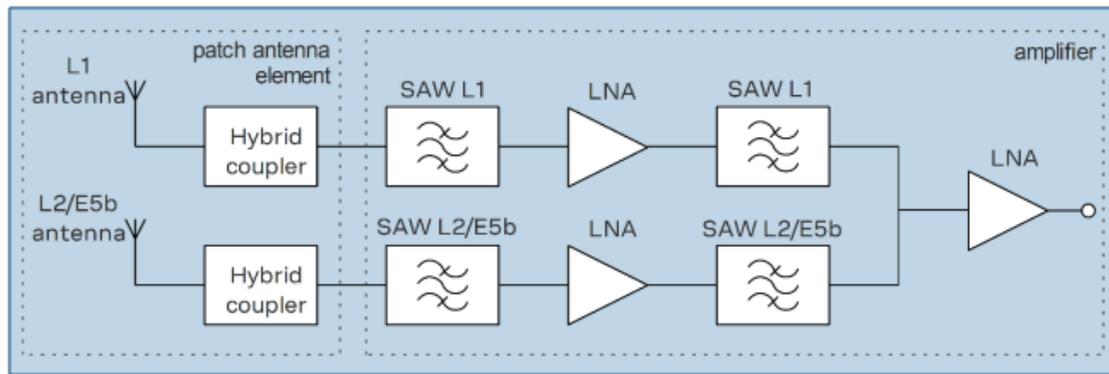


Abbildung 11 Vereinfachtes Blockdiagramm der verwendeten GNSS-Antenne [18, p. 7]

Die Antenne ist in einem robusten und wasserdichten Gehäuse verbaut, welches zudem über einen Magneten an der Gehäuseunterseite zur Montage auf metallischen Objekten verfügt.

Bei Verwendung von Patchantennen empfiehlt der Hersteller die Anbringung einer Metallfläche (Groundplane) unter der Antenne. Dies soll die Verschiebung des Antennen-Phasenzentrums vom Antennen-Referenzpunkt verhindern. Der Antennen-Referenzpunkt ist ein mechanischer Bezugspunkt am untersten Punkt eines Antennenkörpers, auf welchen sich die gemessene Position beziehen soll. Die tatsächlich gemessene Position wird allerdings vom Antennen-Phasenzentrum bestimmt, wo das Signal von der Antenne aufgenommen wird. Der Wegunterschied zwischen Referenzpunkt und Phasenzentrum ist das Phasenzentrum-Offset und ist abhängig von Einstrahlrichtung und Frequenz des Satellitensignals, was zu Messfehlern führen kann. Hochgenaue GNSS-Antennen für exakte Positionsmessungen (z.B. Choke-Ring-Antennen) müssen aus diesem Grund kalibriert werden. [13, p. 9], [19, p. 2] Darüber hinaus sollen die Metallflächen unter der Antenne Signalreflexionen vom Boden reflektieren und somit Mehrwegeempfang verringern. Da in den Testmessungen des GNSS-Kompasses die Antennen auf einem Autodach platziert wurden, dient dieses als Groundplane.

3.2 Übertragungsprotokolle

Die GNSS-Daten stellt der F9P-Empfänger in verschiedenen Übertragungsprotokollen zur Verfügung, welche im Folgenden vorgestellt werden.

3.2.1 NMEA-Standard

Das NMEA-Standardprotokoll ist schon deutlich früher entstanden als das GPS-Satellitensystem. Die National Marine Electronics Association (NMEA) ist eine Vereinigung von Elektro-

nikhändlern, die 1957 in den USA gegründet wurde und die Standardisierung bei der Übertragung von Navigationsdaten im maritimen Bereich vorantrieb. Später wurde der Standard auch bei GNSS-Systemen eingesetzt. Die Daten werden dabei im 8-Bit-ASCII-Format übertragen. Das Protokoll enthält eine Vielzahl verschiedener NMEA-Nachrichten, mit variablen Informationen und Informationsgehalt, was es flexibel für GNSS-Empfänger unterschiedlicher Hersteller macht. Jede Nachricht beginnt mit einem \$-Zeichen, folgend von zwei Buchstaben für das jeweilige Satellitensystem (z.B. GP für GPS) und drei Buchstaben für die Art der Nachricht. Danach folgen die Navigationsdaten, die jeweils durch ein Leerzeichen getrennt sind. [20]

Der F9P-Empfänger kann eine Vielzahl von NMEA-Nachrichten ausgeben, jedoch ist die Nachricht \$GxHDT, welche die für den GNSS-Kompass benötigten Richtungsdaten enthält, nicht verfügbar, weshalb es in den Testmessungen nicht verwendet wurde. Als Ersatz dafür dient das UBX-Protokoll, welches in Abschnitt 3.2.3 erläutert wird.

3.2.2 RTCM-Standard

Der RTCM-Standard der Radio Technical Commission for Maritime Services dient bei DGNSS-Anwendungen der Übertragung von Korrekturdaten in Form von Code- und Trägerphasenkorrekturen, um u.a. die Phasenmehrdeutigkeiten zu lösen. In den ersten Versionen war der Standard nur für die Satellitensysteme GPS und GLONASS verfügbar. Mit der Einführung der RTCM Multiple Signal Messages (MSM) kann der Übertragungsstandard für alle Systeme verwendet werden. [21] Auch dieses Protokoll verfügt über eine Vielzahl verschiedener Gruppen und Nachrichten. Für den GNSS-Kompass werden Nachrichten der Gruppe RTCM MSM 4 verwendet.

3.2.3 UBX-Protokoll

Das UBX-Protokoll ist ein u-blox-proprietäres Übertragungsprotokoll, bei dem die Daten binär mit 8 Bit übertragen werden. Proprietäre Datenschnittstellen werden von Herstellern verwendet, wenn u.a. bei der Verwendung von bestehenden Standardschnittstellen – wie der NMEA-Standard – eine Übertragung von Nachrichten mit höherem Datenumfang nicht effektiv möglich ist. Das ist beispielsweise bei hohen Messraten der Fall, wenn die Datendichte der Schnittstelle erhöht werden muss. Auch zu Konfigurationszwecken der Empfänger sind oft herstellereigene Übertragungsprotokolle erforderlich. Zudem verringern binäre Daten den Verarbeitungsaufwand, da „das Umwandeln von numerischen Daten im ASCII-Format in ein internes Binärformat entfällt.“ [4, p. 157]

Die verschiedenen UBX-Nachrichten sind in Gruppen eingeteilt. Die für die Testmessungen relevanten Gruppen sind UBX-CFG und UBX-NAV. Erstere enthält Nachrichten für die Konfiguration des Empfängers, letztere stellt alle berechneten Navigationsdaten und Satelliteninformationen bereit. Die jeweilige Bezeichnung einer Nachricht besteht jeweils aus „UBX“ - „Bezeichnung der Gruppe“ – „Bezeichnung der Nachricht“ (z. B. UBX-NAV-RELPOSNED). In Abbildung 12 ist der Aufbau einer UBX-Nachricht dargestellt. Jeder Rahmen beginnt mit einer zwei Byte langen Präambel mit den Synchronisationsbytes 0xb5 und 0x62. Danach folgen zwei Bytes für die Bezeichnung der Nachrichtengruppe und der Nachrichten-ID und zwei Bytes für die Angabe der Anzahl an Nutzbytes in der jeweiligen Nachricht. Anschließend folgen die Nutzinformationen. Jeder Rahmen wird durch zwei Prüfsummen mit jeweils einem Byte abgeschlossen. [22, p. 53]

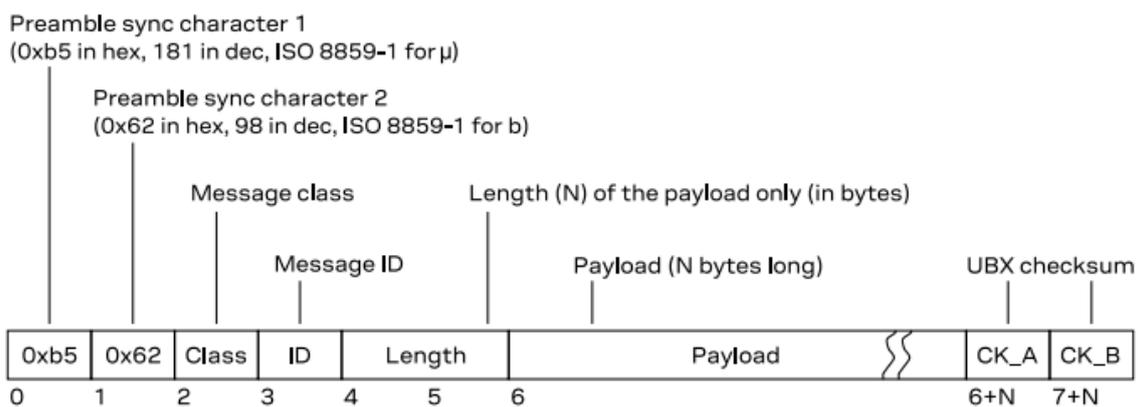


Abbildung 12 Aufbau einer UBX-Nachricht [22, p. 53]

Alle verfügbaren UBX-Nachrichten sind in der u-blox-Interface Description ab Seite 58 aufgelistet.

Zum Auslesen der Positionsdaten wurde die Nachricht UBX-NAV-PVT (Position, Velocity, Time) verwendet. Diese enthält neben den Positionsdaten Breite, Länge und Höhe eine Vielzahl weiterer Informationen über Geschwindigkeit, Zeit und Korrekturdaten. Die Richtungsdaten des GNSS-Kompasses sind in der Nachricht UBX-NAV-RELPOSNED (Relative Position NED) enthalten. In dieser wird der Vektor von der Basis zum Rover durch ein NED-Frame mit den Variablen „relPosN“, „relPosE“ und „relPosD“ beschrieben. Das NED-Frame wurde im Abschnitt 1.3.3 erläutert. Zusätzlich werden die Länge des Vektors mit der Variable „relPosLength“ und die aus dem Vektor resultierende Richtung in Bezug auf geografisch Nord mit der Variable „relPosHeading“ angegeben.

3.3 Konfiguration und Datenaustausch

Der F9P-Empfänger kann durch die UBX-Konfigurationsnachrichten (UBX-CFG) für unterschiedliche Anwendungen vielfältig eingestellt werden. Dazu müssen die Konfigurationsnachrichten mit den entsprechenden Parametern (Items) und den dazugehörigen Werten binär über eine der verfügbaren seriellen Schnittstellen an den Empfänger gesendet werden. Für die ersten Erprobungen wurde dazu die Evaluationssoftware u-center verwendet, welche von u-blox kostenlos zur Verfügung gestellt wird. Das Programm dient zur Übersetzung und Darstellung der vom Empfänger ausgegebenen Daten – es ist keine Messsoftware, in der zusätzliche Berechnungen getätigt werden.

Im folgenden Abschnitt wird die Empfängerkonfiguration des GNSS-Kompasses von Basis- und Roverempfänger mithilfe von u-center erläutert.

In Abbildung 13 ist das Konfigurationsfenster dargestellt, welches über *View* → *Generation 9 Configuration View* → *Advanced Configuration* erreicht werden kann. Dort sind alle Konfigurationsnachrichten aufgelistet, welche durch Doppelklick auf die Bezeichnung aufgeklappt werden können. Danach erscheinen in grün alle der Nachricht zugehörigen Konfigurationsparameter (Items). Mit dem Auswählen eines dieser Items kann diesem im rechten Teil des Fensters ein Wert zugewiesen werden. Nach Eingabe des entsprechenden Wertes muss ausgewählt werden, in welchen Speicher die Einstellung geschrieben werden soll. Bei Auswahl von *Set in RAM* wird die Konfiguration nach Betätigen von *Send config changes* direkt ausgeführt, geht aber nach Abschalten des Empfängers verloren. Für eine dauerhafte Speicherung muss die Einstellung zusätzlich im Flashspeicher abgelegt werden. Nach der Auswahl der Speicherebenen werden die ausgewählten Konfigurationen in der Tabelle *Items to set* aufgelistet und gespeichert, sodass weitere Einstellungen vorgenommen werden können. Wurden alle Einstellungen eingegeben, müssen die aufgelisteten Konfigurationsnachrichten mit *Send config changes* an den Empfänger gesendet werden.

Um die Einstellungen zu einem späteren Zeitpunkt nochmals verwenden zu können, kann die Konfigurationstabelle mit *Save to file* in einer Textdatei gespeichert werden. Mit *Load from file* wird die Textdatei dann wieder in die Tabelle *Items to set* eingefügt, sodass die Konfigurationen wieder an den Empfänger gesendet werden können. Die in den Testmessungen verwendeten Konfigurationstextdateien von Rover und Basis des GNSS-Kompasses sind in Anhang 8.1 beigefügt.

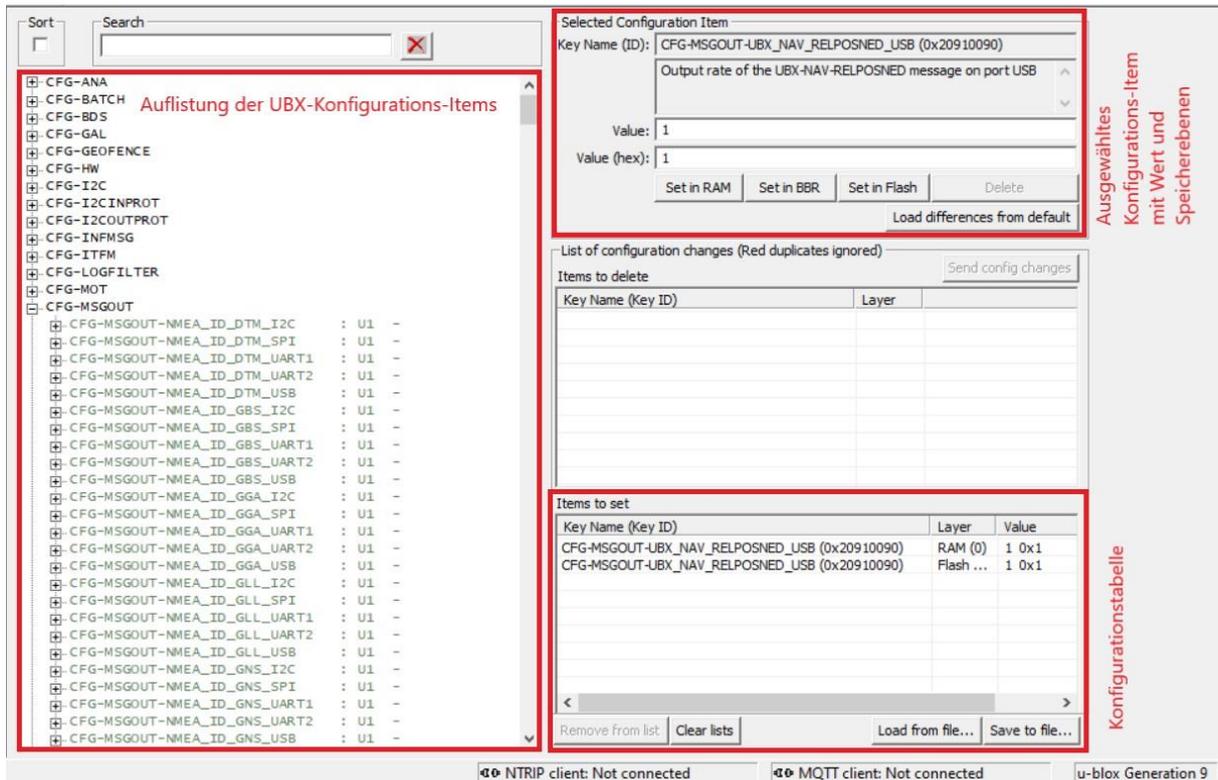


Abbildung 13 Konfigurationsfenster in u-center

Zum Auslesen der Empfängerdaten muss in u-center über *View* → *Messages View* zu den UBX-Nachrichten UBX-NAV-RELPOSNED für die Richtung und UBX-NAV-PVT für die Position navigiert werden. Abbildung 14 zeigt die Darstellung der Informationen der NAV-RELPOSNED-Nachricht, mit Nord- Ost- und Abwärtskomponente der relativen Position des Rovers, sowie Länge und Richtung des Vektors. Die Positionsdaten sind auch in anderen UBX-Nachrichten oder in den NMEA-Nachrichten verfügbar. In den Testversuchen wurde jedoch nur NAV-PVT verwendet.

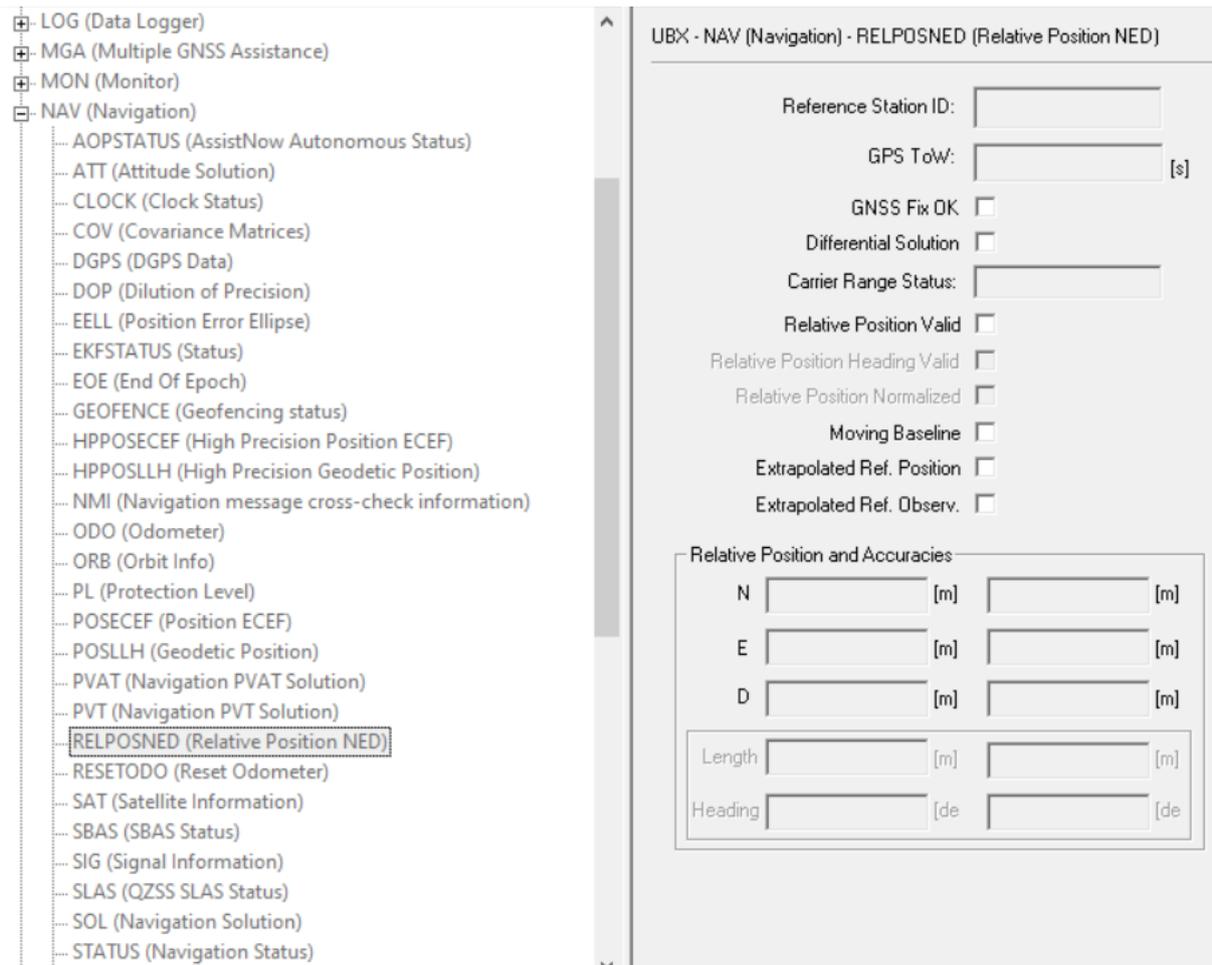


Abbildung 14 Auslesen der Richtungsdaten (relative Position) über den Messages-View

Nach dem Anschließen des EVK-Boards an einen Computer (Windows), muss über den Gerätemanager die jeweilige Bezeichnung des COM-Ports ermittelt werden. In u-center wird dieser dann über *Receiver* → *Connection* ausgewählt. Bei UART-COM-Ports muss zusätzlich die Symbolrate (Baudrate) über *Receiver* → *Baudrate* eingestellt werden, die in der Standardkonfiguration auf 38400 baud festgelegt ist.

3.3.1 Konfiguration für Positionsbestimmungen

Für die Positionsmessungen wurden stets die Standardeinstellungen des F9P-Empfängers verwendet. Diese können in u-center über *Receiver* → *Action* → *Revert config* wiederhergestellt werden. Bei der Verwendung von Korrekturdiensten wurden die in u-center integrierten NTRIP- und MQTT-clients verwendet, welche ebenfalls über den Reiter *Receiver* erreicht werden. Nach der Herstellung einer Verbindung zum Korrekturdienst wurden keine weiteren Einstellungen vorgenommen.

3.3.2 Konfiguration des GNSS-Kompasses

Für die Konfiguration des beweglichen Referenzempfängers (folgend auch als „Moving-Base-Empfänger“ bezeichnet) müssen folgende Einstellungen vorgenommen werden:

- Messrate: 200 ms (CFG-RATE)
- Baudrate UART1 und UART2: 460800 baud (CFG-UART1-BAUDRATE)
- Freigabe und Sperrung der Übertragungsprotokolle auf UART1 und UART2
 - UART1: Freigabe für UBX (In und Out)
 - UART2: Freigabe von RTCM3 (Out)
 - Sperrung der restlichen Protokolle auf UART1, UART2 und USB
- Ausgabe der erforderlichen RTCM3-Nachrichten auf UART2 (CFG-MSGOUT)
 - RTCM3.x Type: 1074, 1084, 1094, 1124, 1230, 4072_0
- Ausgabe der UBX-Nachricht UBX-NAV-PVT auf UART1 für die Positionsdaten (CFG-MSGOUT)

Die Freigabe und Sperrung der Übertragungsprotokolle auf UART1 und UART2 verhindert eine mögliche Überlastung der Schnittstellenpuffer, da nur die nötigsten Nachrichten freigegeben sind. Zum einen ist die Übertragung der RTCM-Korrekturdaten auf UART2 zeitkritisch und zum anderen sind auch NMEA-Nachrichten in der Standardkonfiguration auf UART1 freigegeben, was bei Messraten größer als 1 Hz zu einem Überlaufen des Schnittstellenpuffers führt, da auch die Nachricht UBX-NAV-PVT für die Positionsangabe über UART1 übertragen werden muss. Auch eine Erhöhung der Symbolrate ist erforderlich, um den hohen Datendurchsatz zu ermöglichen. Dies ist aber aufgrund der kurzen Verbindungsleitung zwischen den Empfängern unkritisch.

Für die Konfiguration des Rover-Empfängers müssen folgende Einstellungen vorgenommen werden:

- Messrate: 200 ms (CFG-RATE)
- Baudrate UART1 und UART2: 460800 baud
- Freigabe und Sperrung der Übertragungsprotokolle auf UART1 und UART2

- UART1: Freigabe für UBX (In und Out)
- UART2: Freigabe von RTCM3 (In)
- Sperrung der restlichen Protokolle auf UART1, UART2 und USB
- Ausgabe der UBX-Nachricht UBX-NAV-RELPOSED auf UART1 für die Richtungsdaten (CFG-MSGOUT)
- Zeitbegrenzung des DGNSS-Timeouts: 2 s (NAVSPG-CONSTR-DGNSSSTO)

Die Zeitbegrenzung des DGNSS-Timeouts dient der Überwachung des RTK-Status. Wenn die Korrekturdaten älter als zwei Sekunden sind, verlässt der Empfänger den RTK-Modus und die Richtungsdaten sind ungültig.

3.4 Programmierschnittstelle

Um die u-blox-GNSS-Empfänger auf der Drohne auslesen und fernsteuern zu können, müssen die UBX-Nachrichten über eine Programmierschnittstelle kodiert und dekodiert werden. Dies wurde mithilfe eines Python-Skripts und den Modulen „pySerial“ und „pyubx2“ gelöst. Beim pyubx2-Modul handelt es sich um einen Python3-Parser für das UBX-Protokoll, kann aber auch für das NMEA0183- und RTCM3-Protokoll verwendet werden. Das Modul leistet mit der „UBXReader“-Klasse und der parse()-Methode eine Übersetzung von binären Rohdaten in eine UBX-Nachricht, welche als Objekt der Klasse „UBXMessage“ zurückgegeben wird. Dadurch werden die an einem COM-Port empfangenen binären UBX-Nachrichten, welche die Navigationsdaten enthalten, automatisch ausgewertet und die Daten in die entsprechenden Variablen geschrieben. Der Zugriff auf die COM-Ports wird vom pySerial-Modul geregelt. Weiterhin kann mit der UBXMessage-Klasse eine UBX-Nachricht erstellt und mit der serialize()-Methode in ein Byte-Array umgewandelt werden. Somit lassen sich Konfigurationsnachrichten individuell erstellen, welche dann an den Empfänger gesendet werden können. [23]

Die UBX-Nachrichten werden im pyubx2-Modul in drei Gruppen eingeteilt:

- GET – Ausgabe vom Empfänger (z. B. Richtungs- und Positionsdaten)
- SET – Befehl an den Empfänger (Konfigurationsanweisungen)
- POLL – Abfrage an den Empfänger z. B. einzelner Parameter

Konfigurationsschnittstelle

Mit den Methoden `config_set()`, `config_del()` und `config_poll()` können UBX-Konfigurationsnachrichten mit je 64 Item-Wert-Paaren erstellt werden, wobei Parameter jeweils gesetzt, gelöscht oder abgefragt werden können.

Der im Programm verwendeten `config_set()`-Methode müssen drei Parameter übergeben werden:

- Layer: jeweilige Speicherebene im Empfänger (1=RAM, 2=BBR, 4=Flash, können UND-verknüpft werden → 5=RAM und Flash)
- Transaction: transaktionelle Konfiguration (0=Keine, 1=Start, 2=Laufend, 3=Festschreiben)
- Array mit Item-Wert-Paaren (Konfigurationsparameter)

Bei einer transaktionellen Konfiguration werden gesammelte Konfigurationsnachrichten überprüft, um logische Widersprüche zu vermeiden. Dies ist erforderlich, wenn Konfigurationsparameter voneinander abhängig sind, was bei dieser Anwendung aber nicht der Fall ist. [22, p. 226] Nachdem mit der `config_set()`-Methode eine UBX-Nachricht als `UBXMessage`-Objekt erstellt wurde, wird dieses mit der `serialize()`-Funktion in Byte-Array kodiert, welches an den Empfänger gesendet wird. Die Bezeichnungen der Konfigurations-Items (z.B. `CFG_RATE_MEAS`) sind in der `pyubx2`-Modulbeschreibung aufgelistet.

Empfangsschnittstelle

Für den Empfang von UBX-Nachrichten des Empfängers wertet die `read()`-Methode der `UBXReader`-Klasse die empfangenen Bytes aus. Mit der `parse()`-Methode werden die Rohdaten übersetzt und in ein `UBXMessage`-Objekt geschrieben. Danach können die benötigten Attribute in eine Text- oder Exceldatei geschrieben werden. [23]

Programmablauf

Der in Anhang 8.2 gezeigte Programmcode dient zur Empfängerkonfiguration und zum Auslesen der Messdaten des GNSS-Kompasses. Nach dem Starten des Programms wird abgefragt,

ob der Empfänger konfiguriert werden soll. Ist dies der Fall, müssen die jeweiligen Bezeichnungen der USB-COM-Ports für Rover und Moving-Base eingegeben werden. Danach werden die in den Konfigurations-Arrays gesammelten Parameter an die Empfänger gesendet.

Nach erfolgreicher Konfiguration oder bei Verneinung der Konfigurationsabfrage gelangt das Programm in eine Schleife zur Messwertaufnahme. Dazu muss die Bezeichnung des UART1-COM-Ports des Rovers und ein Dateiname für die Textdatei eingegeben werden. Anschließend werden die Werte für Messdauer, Richtung und RTK-Status tabellarisch in die Textdatei geschrieben. Durch betätigen von *Strg+C* kann die Schleife für die Messaufnahme gestoppt werden. Das Programm startet dann erneut mit der Konfigurationsabfrage.

4 Praktische Erprobung

Im folgenden Kapitel werden die Ergebnisse der Testmessungen der u-blox GNSS-Empfänger dargestellt und ausgewertet sowie die Messaufbauten mit den dazugehörigen Empfängerkonfigurationen beschrieben. Zunächst werden die Testmessungen des GNSS-Kompasses behandelt und darauffolgend die Testmessungen für die Genauigkeit der absoluten Position.

4.1 Test des GNSS-Kompasses

Für die praktische Erprobung des GNSS-Kompasses wurden Messungen sowohl im Stillstand als auch in Bewegung durchgeführt, bei denen die Antennen auf einem Autodach platziert wurden. Obwohl bei einer Anwendung des GNSS-Kompasses auf einer Drohne kaum Probleme mit Abschattungen und Mehrwegeeffekten entstehen dürften, da diese in der Luft weit entfernt von störenden und reflektierenden Objekten ist, werden auch die Messwertschwankungen der Empfänger in besiedeltem Gebiet und auf freier Fläche untersucht und verglichen, da auch eine eventuelle Anwendung des Kompasses auf Peilmessfahrzeugen des PMD vorgesehen ist.

4.1.1 Messaufbau

Für den Messaufbau der Richtungsmessung wurden zwei Evaluation Boards mit dem ZED-F9P-Chip (EVK-F9P) verwendet, sowie die mitgelieferten Patchantennen des Modells ANN-MB-00-00. Der ZED-F9H Chip, welcher eigentlich für Anwendungen zur Richtungsbestimmung vorgesehen ist, war zum Zeitpunkt der Beschaffung nicht auf einem Evaluation Board integriert verfügbar. Da dieser nur eine vereinfachte Variante des F9P ist, können auch zwei F9P-Empfänger verwendet werden. Der F9H-Empfänger kann nur zusammen mit einem F9P-Empfänger verwendet werden und berechnet keine absoluten Positionsdaten, sondern lediglich die Richtungsdaten. [24, p. 4]

Um einen Korrekturdatenstrom von der Moving-Base zum Rover zu ermöglichen, muss eine Verbindung aufgebaut werden. Dies ist sowohl kabelgebunden als auch kabellos möglich. Da Moving-Base- und Rover-Empfänger mit den zugehörigen Antennen fest auf der Drohne verbaut werden, können die Empfänger per Drahtverbindung über eine der verfügbaren seriellen Schnittstellen verbunden werden. Dabei sollte die Schnittstelle so konfiguriert werden, dass nur das RTCM-Protokoll freigegeben ist und auch nur die nötigsten RTCM-Korrekturnachrichten vom Moving-Base-Empfänger gesendet werden. Dies ist insbesondere bei Messraten größer als 1 Hz relevant, da die Datenmenge aufgrund der höheren Rate von RTCM-Nachrichten auf der

seriellen Schnittstelle steigt. In der Standardkonfiguration der Empfänger ist auf der UART2-Schnittstelle nur das RTCM-Protokoll freigegeben, weshalb empfohlen wird, diese auch zu nutzen. Dadurch können eventuelle Probleme bei einer fehlerhaften Konfiguration vorgebeugt werden. Da bei Drohnenflügen hohe Winkelgeschwindigkeiten auftreten, ist die standardmäßige 1 Hz-Messrate ungenügend. In den Testmessungen wird deswegen stets mit einer Messrate von 5 Hz gemessen. Um die hohe Datenrate der RTCM-Nachrichten bei höheren Messraten übertragen zu können, müssen die UART2-Ports beider Empfänger auf 460800 baud konfiguriert werden. [13, p. 12] Da eine unidirektionale Verbindung für die RTCM-Übertragung von Moving-Base zum Rover ausreichend ist, werden nur zwei Kabel für Daten- und Masseleitung benötigt.

Die Richtungsdaten werden vom Rover berechnet und über die UBX-Nachricht UBX-NAV-RELPOSNED ausgegeben. Diese muss über eine der seriellen Schnittstellen (außer UART2) empfangen werden. In den Testmessungen wurden die Richtungsdaten via USB-Kabel und UART1-COM-Port an einen Rechner gesendet. Auch dort ist eine Erhöhung der Symbolrate zwingend erforderlich, um ein Überlaufen der Schnittstellenpuffer zu verhindern.

Der Moving-Base-Empfänger muss so konfiguriert werden, dass er nur die erforderlichen RTCM-Nachrichten auf dem UART2-TX-Pin ausgibt, welche in Abbildung 15 aufgelistet sind. Die Daten der absoluten Position der Drohne müssen im Anwendungsfall auf der Drohne ebenfalls vom Moving-Base-Empfänger auf einer anderen Schnittstelle als UART2 ausgegeben werden. Diese können sowohl durch das NMEA-Protokoll als auch durch das UBX-Protokoll übermittelt werden. Da die Richtungsdaten des Rovers allerdings im UBX-Protokoll enthalten sind, ist es naheliegend, auch die Positionsdaten über dieses zu empfangen. Alle erforderlichen Konfigurationen für beide Empfänger sind in Anhang 8.1 aufgelistet.

Die Messwerte wurden mithilfe des in Abschnitt 3.4 beschriebenen Python-Skripts aufgenommen und in einer Textdatei gespeichert.

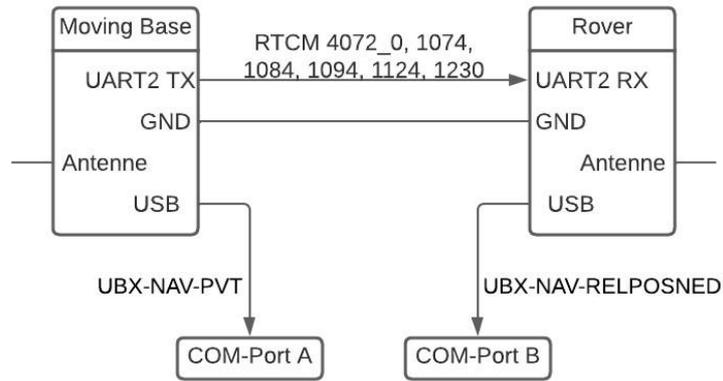


Abbildung 15 Messaufbau GNSS-Kompass Blockschaltbild

In Abbildung 16 wird die Platzierung der Antennen auf einem Autodach gezeigt. Diese befinden sich in einem Abstand von einem Meter zueinander und müssen genau mittig angeordnet werden, damit die Richtung des relativen Vektors auch der des Fahrzeugs entspricht. Die Antenne der Moving-Base ist am hinteren Teil des Fahrzeugs platziert. Der Abstand der Antennen von einem Meter ist ein vorgegebener Richtwert, welcher sich noch am ehesten bzgl. des Anbaus an die Messdrohnen realisieren lässt. Laut Darstellung aus dem u-blox F9P-Datenblatt in Abbildung 17 ist bei einer 1 m-Basislinienlänge eine Richtungsgenauigkeit von $0,4^\circ$ zu erwarten.



Abbildung 16 Platzierung der Antennen auf dem Autodach

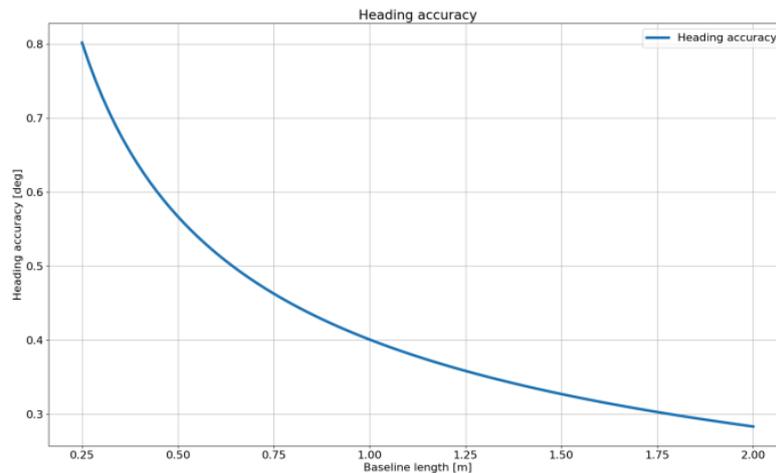


Abbildung 17 Richtungsgenauigkeit des GNSS-Kompasses in Abhängigkeit zur Basislinienlänge laut F9P-Datenblatt [24, p. 6]

4.1.2 Test im Stillstand

Für Anwendungen des GNSS-Kompasses auf den Messdrohnen haben statische Testmessungen eher eine untergeordnete Aussagefähigkeit. Für den eventuellen Einsatz der Empfänger auf Messfahrzeugen des PMD bei Funkstörungsbearbeitungen sollen in den folgenden Tests dennoch die Messwertschwankungen bei verschiedenen Empfangsbedingungen untersucht werden, um die Auswirkung von Abschattung und Mehrwegeempfang auf die Richtungsgenauigkeit festzustellen. Dazu wurden mithilfe des Messaufbaus in Abbildung 16 zwei Zeitverläufe von Richtungsdaten über 15 Minuten aufgenommen und in Abbildung 18 dargestellt. Der blaue Verlauf stellt die Messung dar, bei der das Auto zwischen zwei ca. 20 m voneinander entfernten dreistöckigen Wohnblöcken platziert wurde. Die Gebäude verursachen eine starke Abschattung, was die Anzahl von verfügbaren Satelliten extrem einschränkt. Zudem stellt der geringe Abstand zu den Häuserwänden ein großes Potential für Mehrwegeempfang dar. Beim orangen Zeitverlauf hingegen wurde das Auto auf einem freiflächigen Parkplatz abgestellt, wodurch eine annähernd optimale Satellitenkonstellation möglich war. In beiden Messungen wurde das Auto ungefähr gleich ausgerichtet. Für die Auswertung ist aber eher die Messwertschwankung von Interesse. Die Messabweichung in Bezug auf eine Sollrichtung wird in Abschnitt 4.1.3 untersucht.

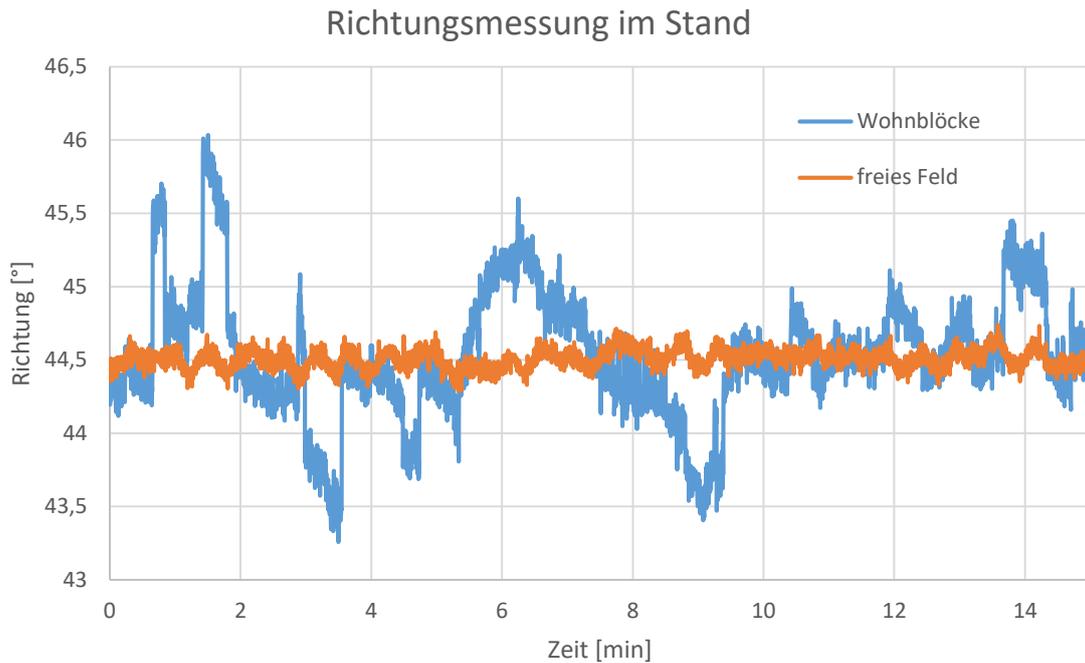


Abbildung 18 Zeitlicher Verlauf der Richtungsmesswerte einer Testmessung im Stand

Im Diagramm sind deutliche Unterschiede zwischen beiden Zeitverläufen zu erkennen. Die Messdaten des blauen Verlaufes mit schlechten Empfangsbedingungen weisen deutlich höhere Schwankungen auf. Die Differenz zwischen dem höchsten und niedrigsten Messwert liegt dabei bei $2,8^\circ$. Beim orangen Messverlauf mit günstigen Empfangsbedingungen liegt diese hingegen nur bei $0,4^\circ$. Demzufolge sollte bei Störpeilungen mit Messfahrzeugen darauf geachtet werden, sich an möglichst freiflächigen Standorten zu platzieren. Aufgrund des zufälligen Charakters der Messwertschwankungen können diese jedoch durch eine Mittelwertbildung gut ausgeglichen werden. So kommen beide Messverläufe nach der Mittelung auf einen Richtungswert von $44,5^\circ$.

4.1.3 Test in Fahrt

In der folgenden Messung soll das Verhalten des GNSS-Kompasses bei dynamischer Anwendung untersucht werden, um festzustellen, ob eine präzise Richtungsbestimmung sichergestellt werden kann, wenn sich die Drohne in Bewegung befindet. Dazu wurde eine Richtungsmessung mithilfe des Messaufbaus der Antennen auf dem Autodach durchgeführt, bei der das Auto auf einer geraden Teststrecke einmal auf $50 \frac{km}{h}$ und einmal auf $100 \frac{km}{h}$ beschleunigt wurde. Die Teststrecke befindet sich in der Nähe von Menz (Gommern) und ist nur von großen Ackerflächen umgeben, sodass die Empfangsbedingungen bzgl. Abschattung und Mehrwegeempfang

denen eines Drohnenfluges am ehesten entsprechen. Die Richtungsdaten wurden bei einer Messrate von 5 Hz in einer Messzeit von 60 s aufgenommen und anschließend in einem Diagramm mit einem Vergleich zur Sollrichtung dargestellt.

Zur Berechnung der Sollrichtung der Teststrecke wurden in Google Earth zwei Achslängen auf dem Gitternetz ausgemessen (Abbildung 19 und Abbildung 20), welche als Katheten zusammen mit der Teststrecke als Hypotenuse ein rechtwinkliges Dreieck ergeben. Durch Anwendung des Satz des Pythagoras in Gleichung (3) kann der Winkel β zwischen der Achse b (Abbildung 20) und der Teststrecke mit den ermittelten Achslängen berechnet werden. Für die Sollrichtung muss der Winkel β von 180° abgezogen werden. Das Ergebnis ist in Gleichung (4) dargestellt.

$$\beta = \arctan\left(\frac{\text{Achse } a}{\text{Achse } b}\right) = 60,9^\circ \quad (3)$$

$$\text{Sollrichtung} = 180^\circ - \beta = 119,1^\circ \quad (4)$$

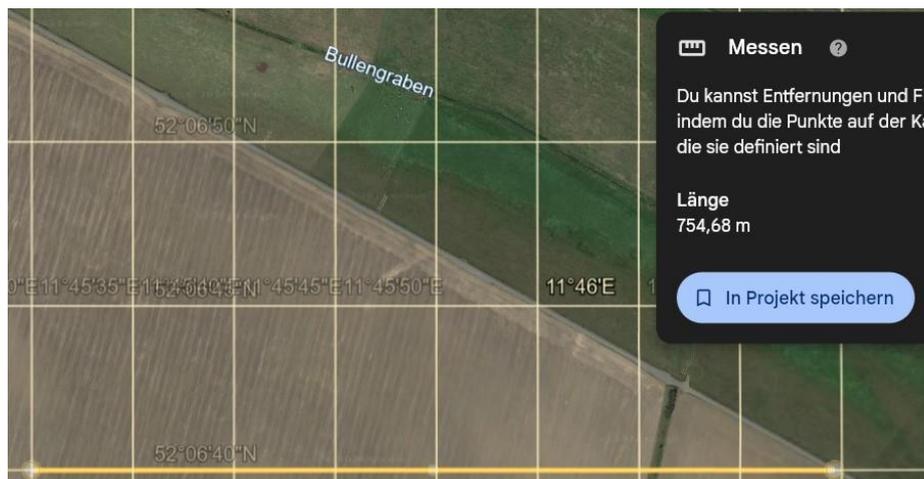


Abbildung 19 Google Earth-Teststrecke Achse a

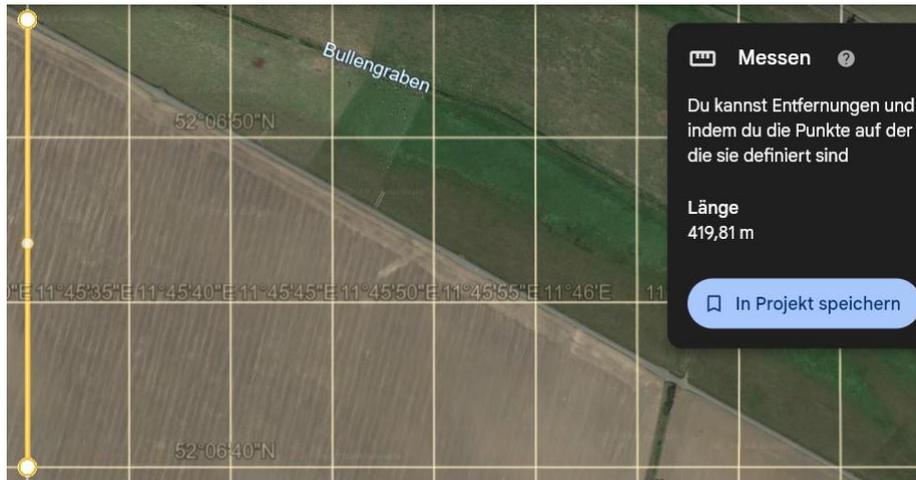


Abbildung 20 Google Earth-Teststrecke Achse b

In Abbildung 21 ist der zeitliche Verlauf der Messergebnisse der zwei Testfahrten dargestellt. Der blaue Verlauf zeigt die Richtungswerte bei einer Geschwindigkeit von $50 \frac{km}{h}$ und der orange Verlauf die Richtungswerte bei $100 \frac{km}{h}$. Die rote Linie markiert den berechneten Richtungswert von $119,1^\circ$.

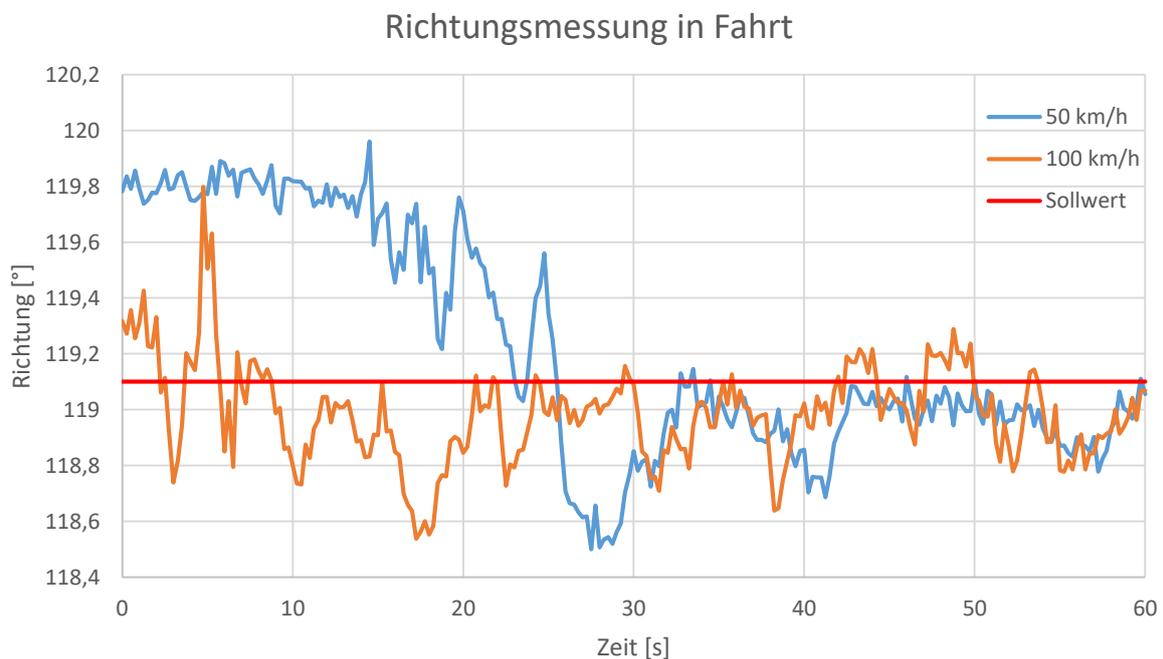


Abbildung 21 Zeitlicher Verlauf der Richtungsmesswerte einer Testfahrt bei verschiedenen Geschwindigkeiten

Abzulesen ist eine maximale Richtungsabweichung von $0,86^\circ$ im blauen Messverlauf. Allerdings können die großen Abweichungen zu Beginn der Messung auf eine Lenkkorrektur beim

Beschleunigen des Fahrzeuges zurückgeführt werden, weshalb eher die Messwerte ab Sekunde 30 zu berücksichtigen sind. In diesem Bereich beträgt die maximale Richtungsabweichung $0,46^\circ$ im orangenen Messverlauf. Da aber in der gesamten Messung leichte Verfälschungen durch Lenkkorrekturen zu erwarten sind, ist von einer Richtungsabweichung von $0,1^\circ$ bis maximal $0,5^\circ$ bei dynamischen Anwendungen auszugehen.

Während der Messung wurde zudem der Status der Mehrdeutigkeitslösung der RTK-Phasemessung überwacht. Dieser ist ebenfalls in der UBX-Nachricht UBX-NAV-RELPOSNED enthalten. Dazu wird das „carrSoln“-Flag auf die Werte 0 für „keine Lösung“, 1 für „RTK-float“ und 2 für „RTK-fixed“ gesetzt. Der Status kann genutzt werden, um die Korrektheit der Richtungsdaten einzuschätzen. Für eine präzise Richtungsmessung sollte sich der Rover stets im RTK-fixed-Status befinden. Durch die Aufzeichnung des Status während der Messung konnte festgestellt werden, dass der Empfänger auch bei einer Geschwindigkeit von $100 \frac{km}{h}$ stabil im RTK-fixed-Modus bleibt.

Ergänzend zu den Messungen in Abschnitt 4.1.2, die zur Einschätzung der Messwertschwankungen des GNSS-Kompasses bei unterschiedlichen Abschattungs- und Mehrwegeempfangsbedingungen dienten, wurde eine Messfahrt durch die Stadt Magdeburg und über eine Landstraße mit dichter angrenzender Bewaldung unternommen, bei welcher wieder der Status des carrSoln-Flag aufgezeichnet wurde. Die Fahrt dauerte etwa 30 Minuten. Von insgesamt 7897 Messwerten wurden nur bei 85 Werten der Status „keine Lösung“ und bei 49 Werten der Status „RTK-float“ aufgenommen. Grund dafür waren erwartungsgemäß Signalabbrüche unter Brücken und bei einer Tunneldurchfahrt. Beim Durchfahren von Häuserschluchten oder Bewaldung konnte der Rover stets den RTK-fixed-Status halten. Zudem konnte festgestellt werden, dass der Empfänger auch nach Signalabbrüchen durch Tunnel oder Brücken den RTK-fixed-Status schon nach ein bis zwei Sekunden nach dem Passieren der Hindernisse wieder erreichen konnte.

4.1.4 Test bei hoher Winkelgeschwindigkeit

Zusätzlich zu den Testmessungen mit dem Auto soll bei der folgenden Untersuchung des GNSS-Kompasses eine schnelle Richtungsänderung einer Drohne simuliert werden, um sicherzugehen, dass die Empfänger auch bei hohen Winkelgeschwindigkeiten, wie sie auf einer Drohne vorkommen können, verlässliche Richtungswerte ausgeben. In Abbildung 22 ist der verwendete Messaufbau gezeigt. Die Antennen sind auf einer Metallschiene im Abstand von

einem Meter befestigt, welche drehbar auf einem Stativ aufgeschraubt ist. Wie vom Hersteller empfohlen, wurden die Antennen mit gleicher Ausrichtung auf einem runden Blech mit 12 cm Durchmesser als Groundplane platziert. Für die schnelle Richtungsänderung bei der Messung wurde die Schiene im 45 °-Winkel zwischen 45 ° und 90 °, bezogen auf geografisch Nord, zügig hin und her bewegt. Das Diagramm in Abbildung 23 stellt die aufgenommenen Messwerte dar.



Abbildung 22 Messaufbau der Richtungsmessung mit drehbarer Metallschiene

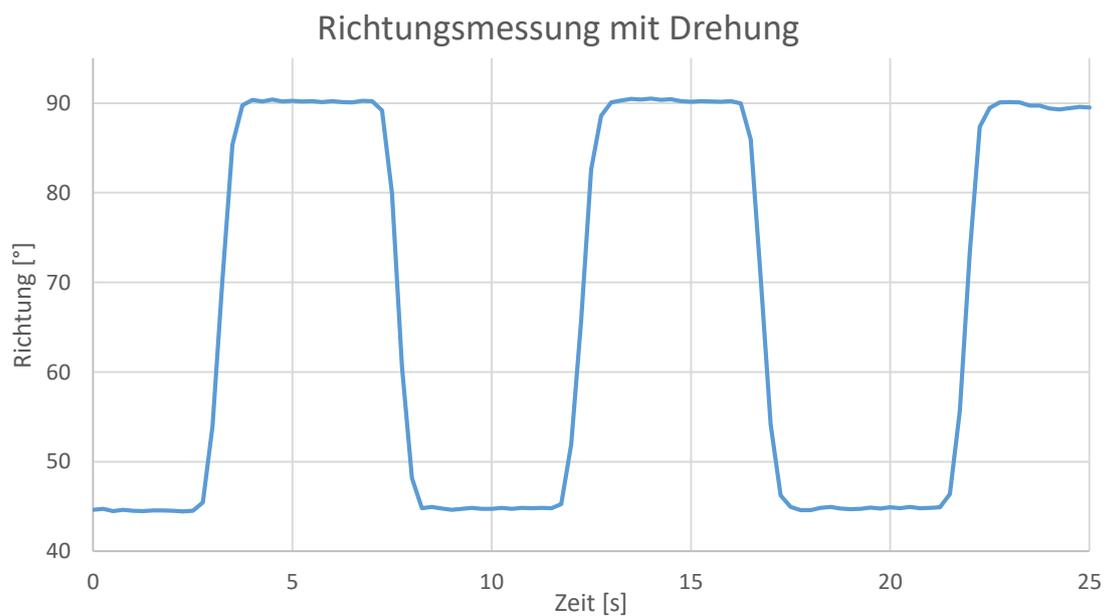


Abbildung 23 Zeitlicher Verlauf der Richtungsmesswerte bei sich schnell ändernder Richtung

Im Diagramm ist zu erkennen, dass trotz schneller Richtungsänderungen des relativen Vektors zwischen den Antennen stabile und exakte Werte berechnet wurden. Es ist kein Über- bzw. Einschwingen des Verlaufs nach der Richtungsänderung erkennbar. Auch eine signifikante Verzögerung zwischen Richtungs- und Messwertänderung konnte bei einer simultanen Darstellung der Werte mithilfe eines zusätzlichen Python-Skriptes nicht beobachtet werden.

4.2 Test der Positionsgenauigkeit

Um festzustellen, welches Konzept der absoluten Positionsbestimmung für eine Anwendung auf der Drohne geeignet ist, werden bei den folgenden Testversuchen die Genauigkeiten der absoluten Positionsmessung untersucht. Dazu wurden Positionsmessungen sowohl mit als auch ohne externe Korrekturen durchgeführt.

4.2.1 Messaufbau

Zur Evaluierung der Positionsgenauigkeit wurden die gemessenen Positionen mit den Koordinaten eines geodätischen Referenzpunktes, welcher vom Landesamt für Vermessung und Geoinformationen Sachsen-Anhalt präzise vermessen wurde, verglichen. Der Referenzstein befindet sich in Magdeburg, Stadtteil Diesdorf, am Rückhaltebecken der Schrote und wurde für Genauigkeitsüberprüfungen von Navigationssystemen für die Öffentlichkeit errichtet. [25]



Abbildung 24 Georeferenzstein mit Markierungskreuz und Koordinaten [25]



Abbildung 25 Georeferenzstein mit Messantenne

Das Kreuz auf der Metallplatte in Abbildung 24 markiert den exakten Referenzpunkt. Für die Testmessungen wurde die Antenne auf dem Kreuz platziert, wie in Abbildung 25 dargestellt.

Der GNSS-Empfänger wurde bei allen Positionsmessungen in den Standardkonfigurationen betrieben.

Für den geodätischen Referenzpunkt sind folgende Koordinaten angegeben:

- Östlicher Längengrad $\lambda = 11^\circ 33' 11,180'' = 11,55310556^\circ$
- Nördlicher Breitengrad $\Phi = 52^\circ 7' 24,503'' = 52,12347306^\circ$
- Höhe über dem WGS84-Referenzellipsoid $h = 115,3 \text{ m}$

4.2.2 Einzelpunktmessung

Im folgenden Abschnitt sollen die Positionsgenauigkeiten der u-blox-Empfänger im Betrieb ohne RTK-Korrekturdaten untersucht werden. In der Standardkonfiguration werden automatisch SBAS-Korrekturen verwendet, wenn keine RTK-Daten verfügbar sind und das Signal eines SBAS-Satelliten empfangen werden kann. Um das Verhalten des Empfängers besser zu untersuchen, wurden Positionsdaten über mehrere Minuten aufgenommen. Die Daten werden aufgeteilt in Längengrad, Breitengrad und Ellipsoidhöhe in Abbildung 27, Abbildung 28 und Abbildung 29 dargestellt. Es wurden drei Messungen durchgeführt. Vor jeder Messung wurde mithilfe der u-center-Software ein Empfängerkaltstart ausgelöst, um eine Unabhängigkeit der Messwerte zu gewährleisten. Ein Kaltstart bedeutet, dass alle im Empfänger gespeicherten Satelliteninformationen wie Ephemeriden, Almanach und Satellitenkorrekturen gelöscht werden.

Zur Einschätzung der Genauigkeit sind in den Diagrammen die Sollwerte des geodätischen Referenzpunktes markiert. Zudem stellen die gestrichelten Linien eine Abweichung von einem Meter in Länge, Breite und Höhe dar. Die Berechnung der 1 m-Positionsfehler erfolgt mithilfe der in Gleichung (5) und (6) berechneten Differenzen der Längengrade $d\lambda_{1m}$ und Breitengrade $d\phi_{1m}$. Am Äquator beträgt der Abstand zwischen Längengraden und Breitenkreisen im WGS84-Referenzellipsoid 111319 m, was jeweils einem Grad entspricht. Demzufolge können die 1 m-Entfernungsdifferenzen dy und dx ins Verhältnis zu den Längen- und Breitenkreisabständen gesetzt und mit 1° multipliziert werden. Abbildung 26 zeigt die Problematik des variablen Abstands der Längengrade. Während der Abstand zwischen den Breitenkreisen (horizontal) konstant ist, nimmt er zwischen den Längengraden (vertikal) zu den Polen hin kontinuierlich ab, was durch den Kosinus in Gleichung (5) mit dem Referenz Breitengrad beschrieben wird.

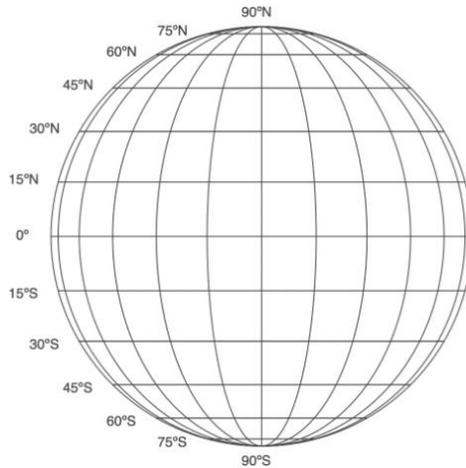


Abbildung 26 Darstellung der Längen- und Breitenkreise auf der Erdkugel [26]

Die Ergebnisse der beiden Gleichungen werden mit den Positionsdaten des geodätischen Referenzpunktes aus Abschnitt 4.2.1 jeweils addiert und subtrahiert, sodass man in jede Himmelsrichtung einen Längengrad- und Breitengradwert mit einer 1 m-Abweichung ausgehend vom Referenzpunkt erhält. Die Erdkrümmung kann bei dieser kleinen Entfernung vernachlässigt werden. [27]

$$d\lambda_{1m} = 1^\circ \cdot \frac{dy}{\cos(\phi_{ref}) \cdot 111319 m} = 0,000015^\circ \quad (5)$$

$$d\phi_{1m} = 1^\circ \cdot \frac{dx}{111319 m} = 0,000009^\circ \quad (6)$$

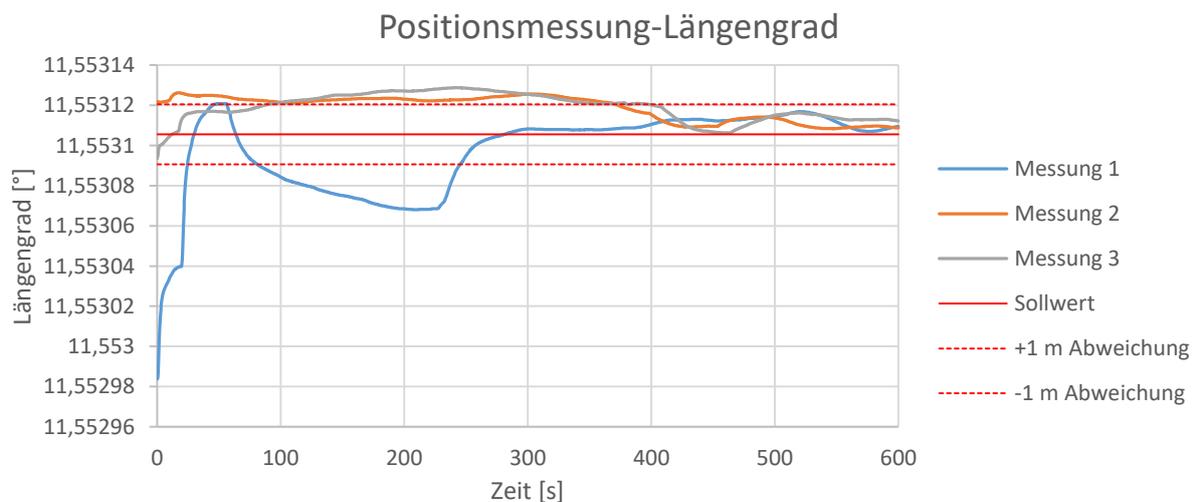


Abbildung 27 Längengrade der Positionsmessung ohne RTK-Korrekturen

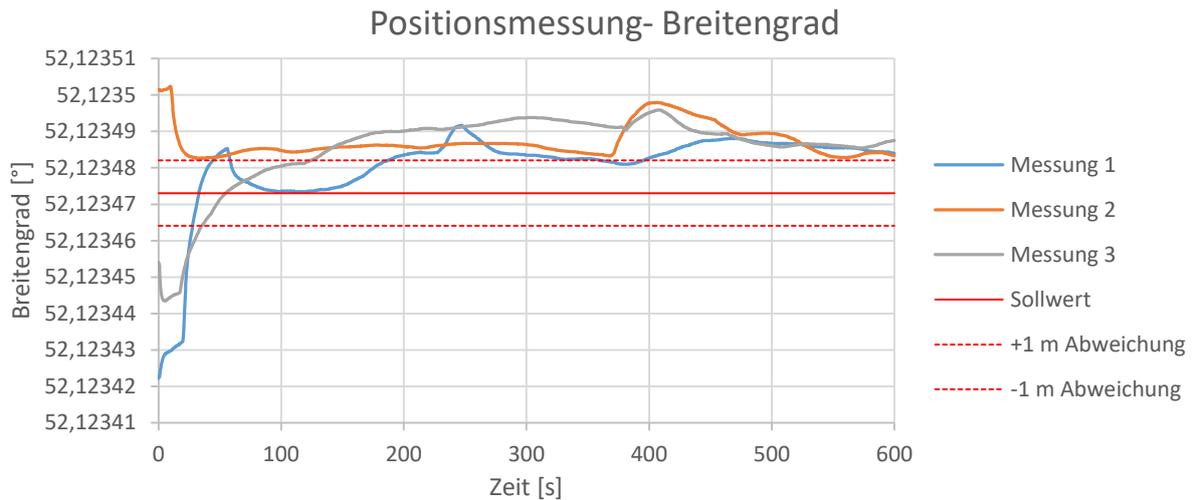


Abbildung 28 Breitenengrade der Positionsmessung ohne RTK-Korrekturen

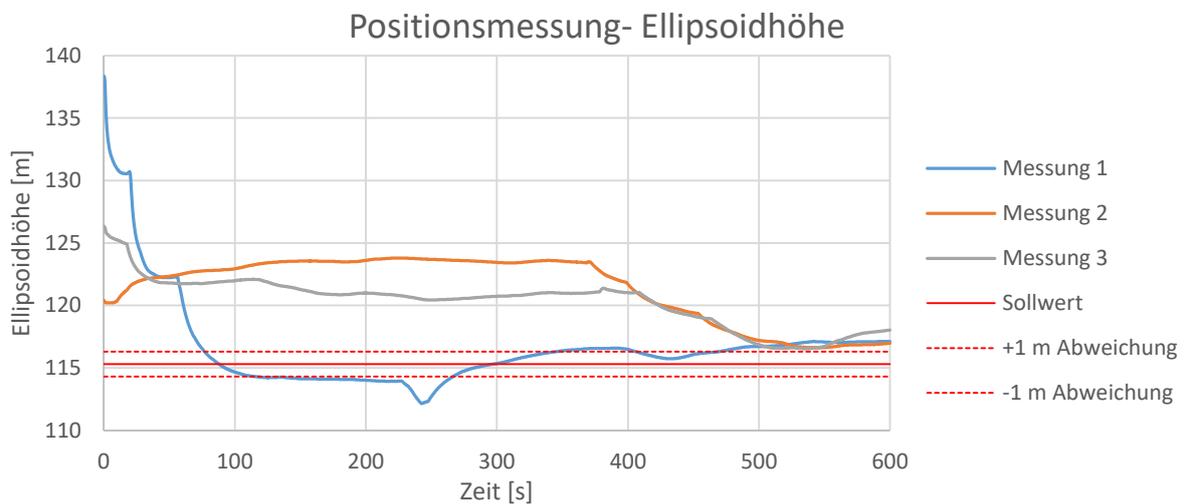


Abbildung 29 Ellipsoidhöhen der Positionsmessung ohne RTK-Korrekturen

Die anfängliche starke Abweichung im Diagramm innerhalb der ersten Minute kann vernachlässigt werden, da der Empfänger nach einem Kaltstart zunächst etwas Zeit benötigt, um grundlegende Informationen über Ephemeriden und Uhrenkorrekturen für die Positionsberechnung zu sammeln. Mit zunehmender Messdauer gleichen sich die Messreihen immer weiter an, sodass die Werte im Bereich von einem Meter streuen, was beim Betrieb mit SBAS-Korrekturen zu erwarten ist. Bezogen auf den Referenzpunkt überschreiten die Abweichungen die 1 m-Marke jedoch teilweise deutlich. Insbesondere zwei Höhenmessverläufe weisen über mehrere Minuten Abweichungen von mehr als fünf Metern auf. Einen möglichen Einfluss auf die Messung könnte die den Messstandort umgebende Vegetation vor allem in südlicher Richtung

darstellen. Dadurch wurden relevante SBAS-Satelliten, welche sich geostationär auf Äquatorhöhe befinden, möglicherweise abgeschattet. Auch Messfehler durch Mehrwegeempfang könnten eine signifikante Verschlechterung der Messverläufe verursacht haben. So stand während der Messung ein PKW in einer Entfernung von wenigen Metern zur Messposition. Da bei einer Drohnenmessung Signalabschattungen von SBAS-Satelliten bzw. von allen GNSS-Satelliten in den meisten Fällen nicht auftreten und somit auch eine generell bessere Satellitenkonstellation möglich ist, können bei Drohnenflügen bessere Ergebnisse erwartet werden. Auch der Einfluss von Signalreflektionen bzw. Mehrwegeempfang ist in der Luft deutlich geringer, da die Drohne nicht von störenden Objekten umgeben ist. Jedoch muss stets mit 3D-Standortabweichungen von ein bis drei Metern gerechnet werden. Zudem sollten die Empfänger vor jeder Messung mehrere Minuten in Betrieb verweilen, sodass dieser Zeit hat, um Satelliteninformationen zu sammeln. Im besten Fall beträgt diese Dauer mindestens 12,5 Minuten (bei GPS), da in dieser Zeit alle 25 Rahmen einer Satellitennachricht übertragen werden können und dementsprechend frühestens dann alle Informationen über den vollständigen Almanach, Ephemeriden und Korrekturangaben vorliegen. Erst dann kann mit einer maximalen Genauigkeit gerechnet werden.

4.2.3 Einmessung der Referenzstation

Die Verwendung einer eigenen mobilen Referenzstation ermöglicht einen RTK-Betrieb für eine präzise Positionsbestimmung der Drohne. Die Positionsgenauigkeit der Drohne ist dabei aber abhängig von der Genauigkeit des vermessenen Referenzpunktes. Für die Referenzstation, welche während der Drohnenmessung fest an einem Punkt auf dem Boden verweilt, besteht die Möglichkeit den F9P-Empfänger für eine längere Zeit einzumessen. Dazu versetzt man den Empfänger in einen „Survey-in“-Modus, in welchem berechnete Positionsdaten gesammelt werden und aus diesen ein gewichteter Mittelwert gebildet wird. Je länger der Empfänger Positionsdaten sammelt, desto genauer sollte die berechnete Position werden, da tageszeitabhängige Messfehler durch die Mittelwertbildung ausgeglichen werden können. Die Positionsgenauigkeit des Survey-in-Modus soll im folgenden Abschnitt untersucht werden.

Zur Konfiguration müssen dem Empfänger zwei Parameter bzw. Anforderungen übermittelt werden. Zum einen eine Mindestmesszeit in Sekunden und zum anderen ein Abstandslimit, welches bestimmt, wie weit gemessene Positionen streuen dürfen, die zur Berechnung des Mittelwerts beitragen. Letzterer Parameter sinkt mit fortlaufender Messung vom Meterbereich in

den Bereich von wenigen Zentimetern. Wenn beide Anforderungen erfüllt sind, geht der Empfänger in den TIME-Modus über und kann dann als Referenzstation genutzt werden. [15, p. 20]

Folgende UBX-Konfigurationsnachrichten müssen an den Empfänger gesendet werden:

CFG-TMODE-MODE	1 # (0-Disabled, 1-Survey in, 2-Fixed)
CFG-TMODE-SVIN_MIN_DUR	0x15180 # 86400 s
CFG-TMODE-SVIN_ACC_LIMIT	0x7d0 # 2000 (Faktor 0,1 in mm → 2000=20 cm)

Mit den Konfigurationen in diesem Beispiel sammelt der Empfänger mindestens einen Tag lang Messwerte. Da es zum einen nicht möglich war, eine Messung über 24 Stunden am geodätischen Referenzpunkt durchzuführen und es zum anderen praxisfern ist, vor einer Drohnenmessung eine Referenzstation 24 Stunden lang einzumessen, wurden zunächst nur über 2 Stunden und 30 Minuten Messdaten gesammelt und dabei in festen Zeitabschnitten die berechnete Position aufgenommen.

Der „Survey-in“-Status wird während der Messung über die UBX-Nachricht UBX-NAV-SVIN ausgegeben und beinhaltet die aktuell berechnete Position durch die Mittelwertbildung der Messwerte im kartesischen ECEF-Format. Die Positionsangabe des geodätischen Referenzpunktes ist allerdings nur in ellipsoidischen Koordinaten (Längengrad, Breitengrad, Höhe) verfügbar, weshalb vorher eine Umrechnung in kartesische Koordinaten erfolgen muss, um die Werte vergleichen zu können.

Für das WGS84-Referenzellipsoid gelten folgende Werte:

- Große Halbachse $a = 6.378.137,000$ m
- Kleine Halbachse $b = 6.356.752,315$ m [28]

Anhand dieser Werte lässt sich ein Abflachungsfaktor f mit der Gleichung (7)

$$f = \frac{a-b}{a} = 3,352810546 \cdot 10^{-3} \quad (7)$$

und die erste numerische Exzentrizität e^2 mit der Gleichung (8) berechnen. [29]

$$e^2 = \frac{a^2 - b^2}{a^2} = 2f - f^2 = 6,69437975 \cdot 10^{-3} \quad (8)$$

Aus der Exzentrizität kann nun ein Querkrümmungsradius N in Gleichung (9) berechnet werden, woraus sich dann mit den angegebenen ellipsoidischen Koordinaten des geodätischen Referenzpunktes die ECEF-Koordinaten X , Y und Z aus den Gleichungen (10), (11) und (12) berechnen lassen. [29]

$$N(\phi) = \frac{a}{\sqrt{1 - e^2 \sin^2(\phi)}} = 6391480,16 \text{ m} \quad (9)$$

$$X = (N + h) \cos(\phi) \cos(\lambda) = 3844689,75 \text{ m} \quad (10)$$

$$Y = (N + h) \cos(\phi) \sin(\lambda) = 785922,69 \text{ m} \quad (11)$$

$$Z = N(1 - e^2) + h \sin(\phi) = 5011341,05 \text{ m} \quad (12)$$

In der folgenden Tabelle sind die Messergebnisse einer „Survey-in“-Messung am geodätischen Messpunkt dargestellt. Nach je 30 Minuten wurden die gemittelten Messwerte der Grafikoberfläche von u-center entnommen. Durch eine Betragsberechnung des Vektors zwischen den gemessenen ECEF-Koordinaten und denen des Referenzpunktes mithilfe der Gleichung (13) kann eine absolute Positionsabweichung berechnet werden.

$$\Delta P = \sqrt{(X_R - X)^2 + (Y_R - Y)^2 + (Z_R - Z)^2} \quad (13)$$

	ECEF X [m]	ECEF Y [m]	ECEF Z [m]	Absoluter Fehler ΔP [m]
Referenzpunkt	3844689,75	785922,69	5011341,05	
30 min	3844690,96	785922,86	5011343,65	2,872803509
60 min	3844690,71	785922,97	5011342,92	2,120589541
90 min	3844690,7	785923,03	5011342,69	1,925538886
120 min	3844690,59	785923,07	5011342,48	1,701440567
150 min	3844690,52	785923,12	5011342,49	1,688608895

Tabelle 1 Survey-in Positionen nach je 30 Minuten mit 3D-Fehler zum Referenzpunkt

In Tabelle 1 sind die abgelesenen Positionsmesswerte in 30 Minuten-Abständen mit den dazugehörigen berechneten 3D-Fehlern in Bezug auf den geodätischen Referenzpunkt dargestellt.

Nach 30 Minuten beträgt der Messfehler fast drei Meter. Zwar sinkt dieser mit zunehmender Messdauer kontinuierlich ab, jedoch wurde der Versuch nach 150 Minuten abgebrochen, da ein weiteres Messen nicht möglich war und die Messgenauigkeit unzufrieden stellend war. Ein Positionsfehler der mobilen Referenzstation von 1,7 Metern würde bedeuten, dass die Positionsabweichung der Drohne ebenfalls bei 1,7 Metern liegt.

Auch wenn eine Messdauer der mobilen Referenzstation von mehreren Stunden für den praxisbezogenen Einsatz im PMD nur unter Umständen realisierbar ist, soll im Folgenden dennoch untersucht werden, welche Positionsgenauigkeiten bei längeren Messzeiten erreicht werden können. Da eine Messung über viele Stunden am geodätischen Referenzpunkt nicht möglich war, wurde ein eigener „Referenzpunkt“ mithilfe des in Abschnitt 4.2.4 behandelten Verfahrens vermessen.

Der Vergleich von Positionsmesswerten mit einem selbst vermessenen Standort mithilfe des verfügbaren Equipments stellt natürlich keinen beweisfesten Nachweis für die Positionsgenauigkeit des Survey-in-Verfahrens dar, da man nicht feststellen kann, inwieweit der selbst vermessene Referenzpunkt der tatsächlichen Position entspricht. Allerdings wird in Abschnitt 4.2.4 gezeigt, dass mithilfe des SAPOS-Korrekturdienstes die geringsten Positionsabweichungen erreicht werden können. Würde die Positionsgenauigkeit bei einer Vermessung einer eigenen Referenzstation durch den Survey-in-Modus im Bereich der Messungen mit SAPOS-Korrekturen liegen, könnte auf eine Verwendung von externen Korrekturdiensten verzichtet werden. Dies soll im Folgenden untersucht werden.

Zunächst wurde eine Antennenposition im RTK-Modus mithilfe des SAPOS-Korrekturdienstes vermessen. Die Antenne befand sich dabei auf einem Garagendach, welches durch umliegende Häuser mittelmäßig abgeschattet wurde. Im RTK-fixed-Modus wurden die ECEF-Koordinaten für die Referenzposition aufgenommen. Anschließend wurde eine 24-stündige Survey-in-Messung durchgeführt. Die Messergebnisse und die berechnete Abweichung sind in Tabelle 2 dargestellt.

	SAPOS-Messung	24 Stunden-Messung
ECEF X [m]	3838771,14	3838770,8
ECEF Y [m]	796429,62	796429,96
ECEF Z [m]	5014171,28	5014172,99
3D-Abweichung [m]	1,78	

Tabelle 2 SAPOS-RTK-Positionsdaten im Vergleich zu denen einer 24-Stunden-Messung

Auch nach einer 24-stündigen Messzeit ergibt sich eine Abweichung der Positionen im Meterbereich. Da die SAPOS-RTK-Messung insbesondere in der Höhe nicht fehlerfrei ist, hat die berechnete Abweichung eine gewisse Toleranz. Dennoch ist diese deutlich höher als erwartet. Demzufolge liefert auch eine mehrstündige Survey-in-Messung keine signifikante Verbesserung der Genauigkeit der absoluten Position.

4.2.4 Korrekturdienstgestützte RTK-Messung

In den letzten Abschnitten wurden Positionsmessungen ohne Verwendung von Korrekturdiensten durchgeführt. Um bei der Positionsbestimmung der Drohne eine möglichst hohe Genauigkeit zu erzielen, werden im folgenden Kapitel verschiedene Korrekturdienste in die Positionsmessung eingebunden.

PointPerfect

Zunächst wurden mehrere Messungen mit Daten des Korrekturdienstes PointPerfect von u-blox am geodätischen Referenzpunkt durchgeführt. Über die Benutzeroberfläche des MQTT-client in der u-center-Software kann sich der GNSS-Empfänger mit einem PointPerfect-Server verbinden und so Korrekturdaten empfangen. Auf diese Weise sollen Genauigkeiten im Bereich 3-6 cm in der Ebene innerhalb weniger Sekunden möglich sein.

Es wurden vier Positionen als ellipsoidische Koordinaten hintereinander jeweils nach einem Kaltstart des Empfängers aufgenommen, nachdem der Empfänger den RTK-fixed Status erreicht hat. Anschließend wurden die Werte mit den Koordinaten des Referenzpunktes verglichen. Die Fehlerberechnung erfolgt wie in Abschnitt 4.2.2. In Tabelle 3 sind alle abgelesenen Positionen in Länge, Breite und Höhe, sowie die berechneten 2D- und Höhenfehler dargestellt.

	Länge [°]	Breite [°]	Höhe [m]	2D-Fehler [m]	Höhenfehler [m]
Referenzpunkt	11,55310556	52,12347306	115,3		
1. Messung	11,55311517	52,12347467	115,34	0,352000512	0,04
2. Messung	11,5531097	52,1234693	114,27	0,438370017	1,03
3. Messung	11,5531149	52,1234783	115,56	0,653333662	0,26
4. Messung	11,55311483	52,12347767	115,55	0,590489824	0,25

Tabelle 3 Messwerte mit Korrekturdaten von PointPerfect und dazugehörige 2D- und Höhenfehler zum Referenzpunkt

Anhand der 2D- und Höhenfehler ist festzustellen, dass die erwartete Positionsgenauigkeit von wenigen Zentimetern nicht erreicht wurde. Zwar streuen die laufend berechneten Positionen im RTK-fixed Status nur noch um 2-3 cm, jedoch weichen diese im Bereich von 35 cm bis 65 cm in der Ebene von der Position des Referenzpunktes ab. Zudem gab es Unterschiede in der Zeitdauer, in der der RTK-fixed Status erreicht wurde. Zum Teil dauerte dies wenige Minuten, allerdings wurden auch einige Messungen abgebrochen, weil der Empfänger den fixed-Status überhaupt nicht erreichen konnte. Außerdem weisen die unterschiedlichen Werte der Messungen auf eine niedrige Wiederholgenauigkeit hin.

SAPOS

Da die Positionsmessung mit dem PointPerfect-Korrekturdienst nicht die gewünschte Genauigkeit ergeben hat, wurde auf den SAPOS-Korrekturdienst Sachsen-Anhalt der deutschen Landesvermessung zurückgegriffen. Zur Verbindung des SAPOS-Korrekturservers mit dem Empfänger wurden, wie in Abbildung 30 gezeigt, über die NTRIP-client-Benutzeroberfläche in der u-center-Software die entsprechenden Zugangsdaten eingegeben, welche auf der Webseite des Geodatenportals Sachsen-Anhalt zur Verfügung gestellt werden.

NTRIP client settings

NTRIP caster settings

Address: 164.133.22.132

Port: 2101

Username: user

Password: ****

NTRIP stream

Update source table [X] Request Interval (sec)

NTRIP mount point: VRS_3_4G_ST Mount point details

Use manual position

Longitude (deg): 0

Latitude (deg): 0

Altitude (m): 0

Geoid sep. (m): 0

OK Cancel

Abbildung 30 Zugriffsdaten für den SAPOS-Server im u-center NTRIP-client, Passwort: user

Wie bei den Testmessungen mit PointPerfect wurden auch hier mehrere Messungen hintereinander jeweils nach einem Kaltstart des Empfängers durchgeführt, um auch die Wiederholgenauigkeit zu überprüfen. Folgende Werte haben sich bei den Messungen ergeben:

	Längengrad [°]	Breitengrad [°]	Ellipsoidhöhe [m]	2D-Fehler [m]	Höhenfehler [m]
Referenzpunkt	11,55310556	52,12347306	115,3		
1	11,55310605	52,12347257	115,64	0,06	0,34
2	11,55310604	52,1234726	115,65	0,05	0,35
3	11,55310603	52,12347254	115,64	0,06	0,34
4	11,55310603	52,12347252	115,64	0,06	0,34
5	11,55310599	52,12347256	115,64	0,06	0,34

Tabelle 4 Positionsmesswerte mit Korrekturdaten von SAPOS mit dazugehörigen 2D- und Höhenfehlern zum Referenzpunkt

Die Messwerte zeigen eine erheblich gesteigerte Positions- und Wiederholgenauigkeit im Vergleich zu den vorherigen Messungen. So ergeben sich sehr konstante Fehler von 6 cm in der Ebene und maximal 35 cm in der Höhe. Um die Wiederholgenauigkeit zu belegen und eine zufällig optimale Messbedingung als Grund für das Ergebnis auszuschließen wurden die Messungen an einem anderen Tag mit einem anderen Empfänger und einer anderen Antenne wiederholt, welche annähernd die gleichen Ergebnisse erbrachten.

4.2.5 Berechnung des Drohnenmittelpunktes

Für die Drohnenmessungen ist bei der Positionsbestimmung der Drohnenmittelpunkt entscheidend, da sich dort die Messantenne befindet. Die Positionsdaten beziehen sich auf den Standort der GNSS-Antenne des Moving-Base-Empfängers, welche sich jedoch aufgrund der Antennenanordnung des GNSS-Kompasses nicht am Drohnenmittelpunkt befindet. Deshalb müssen die Koordinaten des Drohnenmittelpunktes ausgehend von der GNSS-Antenne berechnet werden.

Die UBX-Nachricht UBX-NAV-RELPOSNED enthält im NED-frame drei Werte, die den Basislinienvektor des GNSS-Kompasses in Nord-, Ost- und Abwärtskomponente als positive Längenmaße beschreiben. Befindet sich die Rover-Antenne südlich, westlich oder höher bezogen auf die Moving-Base-Antenne, wird dies durch negative Werte ausgedrückt. Wenn die GNSS-Antennen auf der Drohne so angeordnet werden, dass die Basislinie des GNSS-Kompasses durch den Drohnenmittelpunkt führt, können diese Längenangaben dazu verwendet werden,

um den Drohnenmittelpunkt zu bestimmen. Dazu müssen zunächst alle Werte halbiert werden, um auf den Mittelpunkt der Basislinie, also den Drohnenmittelpunkt zu kommen. Erfolgt die Positionsangabe durch den Moving-Base-Empfänger in ellipsoidischen Koordinaten (Breite, Länge, Höhe), kann die Positionsberechnung des Drohnenmittelpunktes analog zur Positionsfehlerberechnung in Abschnitt 4.2.2 erfolgen. Die Nordkomponente des NED-frame wird in Gleichung (6) als dx eingesetzt und die Ostkomponente in Gleichung (5) als dy . Zur Berücksichtigung des variablen Abstands der Längengrade wird der aktuelle Breitengrad der Drohnenposition in den Kosinus der Gleichung (5) eingesetzt. Abhängig von der Position der Rover-Antenne relativ zur Moving-Base-Antenne erhält man positive oder negative Ergebnisse, welche zum Breiten- und Längengrad der Moving-Base-Antennenposition addiert bzw. davon subtrahiert werden können. Die Höhenkomponente kann nach der Halbierung direkt mit der aktuellen Höhe verrechnet werden.

5 Fazit und Ausblick

Das Ziel dieser Arbeit war die Erprobung eines Multiband-GNSS-Empfängers der Firma u-blox für den Einsatz auf Hochfrequenzmessdrohnen, welche vor allem bei der Störungssuche und bei Vermessungen von Antennenanlagen durch den Prüf- und Messdienst der Bundesnetzagentur einen erleichterten Arbeitsablauf erbringen sollen. Das Hauptaugenmerk lag dabei auf einer möglichst genauen und stabilen Richtungsbestimmung unter dynamischen Bedingungen, welche mithilfe eines GNSS-Kompasses, bestehend aus zwei Antennen und Empfängern, realisiert wurde. Zudem sollte untersucht werden, welche Genauigkeiten bei der absoluten Positionierung mit den beschafften Empfängern mit unterschiedlichen Korrekturmethode erreicht werden können.

Für die ersten Testversuche des GNSS-Kompasses wurde zunächst die Evaluationssoftware u-center verwendet, welche das Konfigurieren der Empfänger und das Auslesen der Daten über das proprietäre UBX-Protokoll erleichtert. Zuerst wurden die für den Messaufbau des GNSS-Kompass benötigten Konfigurationsparameter ermittelt und ein stabiler Messablauf sichergestellt. Da die standardmäßige Messrate von 1 Hz für die Drohnenmessungen aufgrund der hohen Geschwindigkeiten nicht ausreicht, wurde diese auf 5 Hz erhöht. Dabei ergaben sich trotz einer hohen Symbolrate Probleme bei der Datenübertragung über die seriellen Schnittstellen, da die Sendepuffer der Empfänger aufgrund der größeren Datenmenge überlastet wurden. Abhilfe hat eine veränderte Konfiguration geschaffen, bei der alle standardmäßig aktivierten Navigationsnachrichten, darunter die nicht benötigten NMEA-Nachrichten, deaktiviert wurden. Da u-center zur genauen Auswertung der Messdaten ungeeignet ist und zudem eine Datenschnittstelle zur Anbindung der Empfänger an einen Drohnenrechner geschaffen werden musste, wurde eine Lösung bzgl. einer Programmierschnittstelle gesucht. Dazu diente das Python-Modul „pyubx2“, mit welchem durch einen integrierten Parser die proprietären UBX-Navigations- und Konfigurationsnachrichten sowohl gelesen als auch generiert werden können. Mithilfe des Moduls wurde ein Python-Skript geschrieben, womit das Auslesen der Richtungsdaten und die Konfiguration der Empfänger möglich ist. Dieses dient zum einen als Beispiel für die spätere Integration der Empfänger auf der Drohne. Zum anderen wurden mit dem Programm die in dieser Arbeit ausgewerteten Messwerte aufgenommen.

Für die Erprobung des Kompass-Messaufbaus wurde versucht, die dynamischen Verhältnisse bei Drohnenflügen nachzubilden. Dazu wurden Messungen bei erhöhten Geschwindigkeiten

bei Geradeausfahrt und bei hohen Winkelgeschwindigkeiten durchgeführt. Dabei konnte festgestellt werden, dass die Empfänger auch unter schnellen dynamischen Bedingungen präzise und stabile Richtungswerte liefern. So lag die maximale Messwertabweichung nach Berücksichtigung von Messfehlern unter $0,5^\circ$. Zudem wurden auch das Verhalten und die Präzision bei schlechten Empfangsbedingungen für einen möglichen Einsatz der Anwendung auf Messfahrzeugen des PMD untersucht. Bei statischen Messungen mit hoher Signalabschattung konnte eine deutlich größere Messwertschwankung beobachtet werden als bei einer Messung mit optimalen Empfangsbedingungen, bei welcher die maximale Messwertschwankung $0,4^\circ$ betrug. Jedoch zeichnen sich die Schwankungen durch einen zufälligen Charakter aus, sodass diese durch eine Mittelwertbildung nach mehrminütiger Messzeit kompensiert werden können. Bei einer Testfahrt durch dicht besiedelte und bewaldete Gebiete konnten darüber hinaus ebenfalls stabile und korrekte Messwerte sichergestellt werden, wobei es lediglich bei Tunnel- oder Brückendurchfahrten zu Signalabbrüchen und fehlerhaften Richtungswerten kam. Auf Grundlage der erreichten Ergebnisse kann als Fazit festgehalten werden, dass sich der u-blox ZED-F9P-Empfänger für den Einsatz auf einer Messdrohne zur Richtungsbestimmung eignet.

Im zweiten Teil der Arbeit wurde untersucht, welche absoluten Positionsgenauigkeiten mit den Empfängern und mithilfe unterschiedlicher Korrekturmethode erreicht werden können. Die einfachste Variante mit dem geringsten Hardware- und Kostenaufwand stellt die gebührenfreie Korrektur durch geostationäre EGNOS-Satelliten dar, welche standardmäßig von den Empfängern ausgewertet werden. Dabei ist besonders in der Höhenangabe mit größeren Abweichungen von ca. 5-10 Metern zu rechnen und in der Ebene mit ca. 1-3 Metern.

Durch die Einmessung einer eigenen Referenzstation ergaben sich auch nach 24 Stunden Messzeit absolute 3D-Fehler von 1-2 Metern. Der Grund für die Abweichung könnten systematische Fehler sein, welche sich durch eine Mittelwertbildung nicht ausgleichen lassen. Die Positionsgenauigkeit der Referenzstation lässt sich jedoch durch die präzise relative Positionierung der RTK-Technik an der Drohne konstant halten. Die geringste Abweichung von 6 cm in der Ebene und 34 cm in der Höhe wurde mit dem SAPOS-Korrekturdienst erzielt. Im Gegensatz zum PointPerfect-Dienst von u-blox konnte mit SAPOS stets ein stabiler RTK-fixed-Status und damit ein Streukreisradius (CEP) der gemessenen Positionen von wenigen Zentimetern erreicht werden.

Die für die Erprobung des GNSS-Kompasses verwendeten Evaluationsplatinen mit den zugehörigen Gehäusen und den robusten Patchantennen lassen sich problemlos für einen dauerhaften Einsatz auf den PMD-Messfahrzeugen integrieren. Für den Anbau an eine Drohne ist die Hardware jedoch aufgrund von Größe und Gewicht eher ungeeignet, weshalb es in folgenden Entwicklungen erforderlich sein wird, kompaktere Platinen und ggf. auch Antennen zu beschaffen. Weiterhin ist eine weiterführende Softwareentwicklung nötig, um die Empfänger über einen Drohnenrechner auslesen und bedienen zu können. Abhängig von der erforderlichen Positionsgenauigkeit bei den Drohnenmessungen müssen die Konzepte für die Verwendung einer eigenen Referenzstation oder externer Korrekturdienste (außer SBAS) weiter konkretisiert und erprobt werden.

6 Abkürzungsverzeichnis

CDMA	<i>Code Division Multiple Access</i>
CEP	<i>Circular Error Probable</i>
DGNSS	<i>Differential Global Navigation Satellite System</i>
e^2	<i>Exzentrizität</i>
ECEF	<i>Earth Centered Earth Fixed</i>
EGNOS	<i>European Geostationary Overlay Service</i>
EMV	<i>elektromagnetische Verträglichkeit</i>
f	<i>Abflachungsfaktor</i>
GLONASS	<i>Global Navigation Satellite System</i>
GNSS	<i>Global Navigation Satellite System</i>
GPS	<i>Global Positioning System</i>
h	<i>Höhe</i>
HEPS	<i>Hochpräziser Echtzeit Positionierungs-Service</i>
HF	<i>Hochfrequenz</i>
LNA	<i>Low Noise Amplifier</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
MSM	<i>Multiple Signal Messages</i>
N	<i>Querkrümmungsradius</i>
NED	<i>North East Down</i>
NMEA	<i>National Marine Electronics Association</i>
NTRIP	<i>Networked Transport of RTCM via Internet Protocol</i>
OSR	<i>Observation Space Representation</i>
PMD	<i>Prüf- und Messdienst</i>
PPP	<i>Precise Point Positioning</i>
PRN	<i>Pseudo Random Noise</i>
RTCM	<i>Radio Technical Commission for Maritime Services</i>
RTK	<i>Real Time Kinematic</i>
SA	<i>Selective Availability</i>
SAPOS	<i>Satellitenpositionierungsdienst der deutschen Landesvermessung</i>
SAW	<i>Surface Acoustic Wave</i>
SBAS	<i>Satellite Based Augmentation System</i>

SSR	<i>State Space Representation</i>
WLAN	<i>Wireless Local Area Network</i>
λ	<i>Längengrad</i>
Φ	<i>Breitengrad</i>

7 Literaturverzeichnis

- [1] M. Michaelis, „Konzept und Erprobung einer zentimetergenauen Positionsbestimmung für HF-Messdrohnen,“ 2023.
- [2] H. Albrecht, Geometrie und GPS: Mathematische, physikalische und technische Grundlagen der Satellitenortung verständlich erklärt, Springer, 2022.
- [3] A. Heßelbarth, „Statische und kinematische GNSS-Auswertung mittels Precise Point Positioning (PPP),“ München, 2011.
- [4] J.-M. Zogg, GPS und GNSS: Grundlagen der Ortung und Navigation mit Satelliten, Chur, 2011.
- [5] F. Schrödter, GPS Satellitennavigation, Poing: Franzis-Verlag GmbH, 1994.
- [6] E. K. Jens Riecken, „Der Satellitenpositionierungsdienst der deutschen Landesvermessung – SAPOS,“ zfv, 2017.
- [7] J. V. Sickel, „John A. Dutton e-Education Institute,“ [Online]. Available: <https://www.e-education.psu.edu/geog862/node/1727>. [Zugriff am 14 März 2024].
- [8] B. K. K. F. M. U. d. H. Thomas Dautermann, „GNSS Double Differences Used as Beacon Landing System for Aircraft,“ Springer, 2021.
- [9] P. B. Jens Riecken, „Ein neuer Satellitenpositionierungsdienst für Deutschland,“ zfv, 2020.
- [10] D. T. Prinz, „GPS in den Geowissenschaften,“ [Online]. Available: https://ivvgeo.uni-muenster.de/vorlesung/GPS_Script/messung_transformation.html. [Zugriff am 23 Januar 2024].
- [11] „qualimobil,“ [Online]. Available: <https://qualimobil.de/index.php/koordinatensysteme/ecef-system-und-kartesische-gps-koordinaten#:~:text=ECEF%2DSystem%20bestimmt%2C%20einem%20erdzentrierten,um%20die%20Z%2DAchse%20rotiert..> [Zugriff am 23 Januar 2024].

- [12] „Advanced Navigation,“ 2022. [Online]. Available: <https://docs.advancednavigation.com/gnss-Compass/NedCoordSystem.htm>. [Zugriff am 27 März 2024].
- [13] u-blox AG, „ZED-F9P Moving base applications Integration manual,“ Switzerland, 2023.
- [14] „GeoBasis-DE Satellitenpositionierungsdienst der deutschen Landesvermessung,“ [Online]. Available: <https://zentrale-stelle-sapos.de/gebuehren-anmeldung/>. [Zugriff am 25 März 2024].
- [15] u-blox AG, „ZED-F9P High precision GNSS module Integration manual,“ Switzerland, 2023.
- [16] u-blox AG, „EVK-F9P-01 Evaluation kit User guide,“ Switzerland, 2023.
- [17] D. Mietke, „elektroniktutor,“ [Online]. Available: <https://www.elektroniktutor.de/elektrophysik/rauschen.html>. [Zugriff am 22 Februar 2024].
- [18] u-blox AG, „ANN-MB series Multi-band, high precision GNSS antennas Data Sheet,“ Switzerland, 2022.
- [19] SAPOS, „Landesamt für Vermessung und Geobasisinformation,“ Januar 2014. [Online]. Available: <https://lvermgeo.rlp.de/fileadmin/lvermgeo/pdf/produkte/sapos/Antennen.pdf>. [Zugriff am 21 Februar 2024].
- [20] E. Gakstatter, „GPS WORLD,“ 4 Februar 2015. [Online]. Available: <https://www.gpsworld.com/what-exactly-is-gps-nmea-data/>. [Zugriff am 22 Februar 2024].
- [21] T. Wagner, Genesys, 23 Juni 2023. [Online]. Available: <https://genesys-offenburg.de/support/application-aids/gnss-basics/the-rtcm-multiple-signal-messages-msm/>. [Zugriff am 24 Februar 2024].
- [22] u-blox AG, „u-blox F9 HPG 1.32 u-blox F9 high precision GNSS receiver Interface Description,“ Switzerland, 2022.
- [23] „github.com,“ [Online]. Available: <https://github.com/semuconsulting/pyubx2>. [Zugriff am 26 Februar 2024].

- [24] u-blox, „ZED-F9H-01B Module for heading applications Professional grade Data sheet,“ Switzerland, 2023.
- [25] „1. Georeferenzpunkt Sachsen-Anhalts in Magdeburg Diesdorf,“ [Online]. Available: <https://www.magdeburg.de/GeoreferenzpunktMagdeburg>. [Zugriff am 18 Januar 2024].
- [26] „SciencePickle,“ 2023. [Online]. Available: <https://sciencepickle.com/earth-systems/coordinate-system/>. [Zugriff am 26 März 2024].
- [27] T. u. I. v. M. K. Kompf.de - Projekte, „Kompf.de,“ [Online]. Available: <https://www.kompf.de/gps/distcalc.html>. [Zugriff am 24 Januar 2024].
- [28] „GISWIKI,“ [Online]. Available: <http://giswiki.org/wiki/WGS84>. [Zugriff am 23 Januar 2024].
- [29] D. J. Bolz, „Navigation mit Satelliten: Das Global Positioning System (GPS),“ [Online]. Available: <http://www.joachim-bolz.de/projekte/gps/gps.html>. [Zugriff am 23 Januar 2024].

8 Anhang

8.1 Empfängerkonfiguration GNSS-Kompass

8.1.1 Moving-Base-Empfänger 5 Hz Messrate

Config changes format version 1.0

created by u-center version 23.08 at 16:36:39 on Tuesday, 23 Jan 2024

[del]

[set]

```
RAM CFG-RATE-MEAS    0xc8      # write value 200 0xc8      to item id 0x30210001 in layer 0
Flash CFG-RATE-MEAS  0xc8      # write value 200 0xc8      to item id 0x30210001 in layer 2
RAM CFG-UART1-BAUDRATE 0x70800    # write value 460800 0x70800    to item id 0x40530001 in layer 0
Flash CFG-UART1-BAUDRATE 0x70800    # write value 460800 0x70800    to item id 0x40530001 in layer 2
RAM CFG-UART1INPROT-UBX 1          # write value 0          to item id 0x10750001 in layer 0
Flash CFG-UART1INPROT-UBX 1          # write value 0          to item id 0x10750001 in layer 2
RAM CFG-UART1INPROT-NMEA 0          # write value 0          to item id 0x10750002 in layer 0
Flash CFG-UART1INPROT-NMEA 0          # write value 0          to item id 0x10750002 in layer 2
RAM CFG-UART1INPROT-RTCM3X 0          # write value 0          to item id 0x10750004 in layer 0
Flash CFG-UART1INPROT-RTCM3X 0          # write value 0          to item id 0x10750004 in layer 2
RAM CFG-UART1OUTPROT-UBX 1          # write value 0          to item id 0x10760001 in layer 0
Flash CFG-UART1OUTPROT-UBX 1          # write value 0          to item id 0x10760001 in layer 2
RAM CFG-UART1OUTPROT-NMEA 0          # write value 0          to item id 0x10760002 in layer 0
Flash CFG-UART1OUTPROT-NMEA 0          # write value 0          to item id 0x10760002 in layer 2
RAM CFG-UART1OUTPROT-RTCM3X 0          # write value 1          to item id 0x10760004 in layer 0
Flash CFG-UART1OUTPROT-RTCM3X 0          # write value 1          to item id 0x10760004 in layer 2
RAM CFG-UART2-BAUDRATE 0x70800    # write value 460800 0x70800    to item id 0x40530001 in layer 0
Flash CFG-UART2-BAUDRATE 0x70800    # write value 460800 0x70800    to item id 0x40530001 in layer 2
RAM CFG-UART2INPROT-UBX 0          # write value 0          to item id 0x10750001 in layer 0
Flash CFG-UART2INPROT-UBX 0          # write value 0          to item id 0x10750001 in layer 2
RAM CFG-UART2INPROT-NMEA 0          # write value 0          to item id 0x10750002 in layer 0
Flash CFG-UART2INPROT-NMEA 0          # write value 0          to item id 0x10750002 in layer 2
RAM CFG-UART2INPROT-RTCM3X 0          # write value 0          to item id 0x10750004 in layer 0
Flash CFG-UART2INPROT-RTCM3X 0          # write value 0          to item id 0x10750004 in layer 2
RAM CFG-UART2OUTPROT-UBX 0          # write value 0          to item id 0x10760001 in layer 0
Flash CFG-UART2OUTPROT-UBX 0          # write value 0          to item id 0x10760001 in layer 2
RAM CFG-UART2OUTPROT-NMEA 0          # write value 0          to item id 0x10760002 in layer 0
Flash CFG-UART2OUTPROT-NMEA 0          # write value 0          to item id 0x10760002 in layer 2
RAM CFG-UART2OUTPROT-RTCM3X 1          # write value 1          to item id 0x10760004 in layer 0
Flash CFG-UART2OUTPROT-RTCM3X 1          # write value 1          to item id 0x10760004 in layer 2
RAM CFG-USBOUTPROT-UBX 0          # write value 0          to item id 0x10780001 in layer 0
Flash CFG-USBOUTPROT-UBX 0          # write value 0          to item id 0x10780001 in layer 2
RAM CFG-USBOUTPROT-NMEA 0          # write value 0          to item id 0x10780002 in layer 0
Flash CFG-USBOUTPROT-NMEA 0          # write value 0          to item id 0x10780002 in layer 2
RAM CFG-USBOUTPROT-RTCM3X 0          # write value 0          to item id 0x10780004 in layer 0
```

```

Flash CFG-USBOUTPROT-RTCM3X 0          # write value 0          to item id 0x10780004 in layer 2
RAM CFG-MSGOUT-RTCM_3X_TYPE1074_UART2 0x1      # write value 1 0x1      to item id 0x20910360 i
Flash CFG-MSGOUT-RTCM_3X_TYPE1074_UART2 0x1      # write value 1 0x1      to item id 0x20910360 i
RAM CFG-MSGOUT-RTCM_3X_TYPE1084_UART2 0x1      # write value 1 0x1      to item id 0x20910365 i
Flash CFG-MSGOUT-RTCM_3X_TYPE1084_UART2 0x1      # write value 1 0x1      to item id 0x20910365 i
RAM CFG-MSGOUT-RTCM_3X_TYPE1094_UART2 0x1      # write value 1 0x1      to item id 0x2091036a i
Flash CFG-MSGOUT-RTCM_3X_TYPE1094_UART2 0x1      # write value 1 0x1      to item id 0x2091036a i
RAM CFG-MSGOUT-RTCM_3X_TYPE1124_UART2 0x1      # write value 1 0x1      to item id 0x2091036f i
Flash CFG-MSGOUT-RTCM_3X_TYPE1124_UART2 0x1      # write value 1 0x1      to item id 0x2091036f i
RAM CFG-MSGOUT-RTCM_3X_TYPE1230_UART2 0x1      # write value 1 0x1      to item id 0x20910305 i
Flash CFG-MSGOUT-RTCM_3X_TYPE1230_UART2 0x1      # write value 1 0x1      to item id 0x20910305 i
RAM CFG-MSGOUT-RTCM_3X_TYPE4072_0_UART2 0x1      # write value 1 0x1      to item id 0x20910300 i
Flash CFG-MSGOUT-RTCM_3X_TYPE4072_0_UART2 0x1      # write value 1 0x1      to item id 0x20910300 i
RAM CFG-MSGOUT-UBX_NAV_PVT_UART1 0x1          # write value 1 0x1      to item id 0x20910007 in lay
Flash CFG-MSGOUT-UBX_NAV_PVT_UART1 0x1          # write value 1 0x1      to item id 0x20910007 in lay

```

8.1.2 Rover-Empfänger 5 Hz Messrate

Config changes format version 1.0

created by u-center version 23.08 at 16:33:27 on Tuesday, 23 Jan 2024

[del]

[set]

```

RAM CFG-RATE-MEAS      0xc8          # write value 200 0xc8    to item id 0x30210001 in layer 0
Flash CFG-RATE-MEAS    0xc8          # write value 200 0xc8    to item id 0x30210001 in layer 2
RAM CFG-UART1-BAUDRATE 0x70800        # write value 460800 0x70800 to item id 0x40520001 in layer 0
Flash CFG-UART1-BAUDRATE 0x70800        # write value 460800 0x70800 to item id 0x40520001 in layer 2
RAM CFG-UART1INPROT-UBX 1          # write value 0          to item id 0x10750001 in layer 0
Flash CFG-UART1INPROT-UBX 1          # write value 0          to item id 0x10750001 in layer 2
RAM CFG-UART1INPROT-NMEA 0          # write value 0          to item id 0x10750002 in layer 0
Flash CFG-UART1INPROT-NMEA 0          # write value 0          to item id 0x10750002 in layer 2
RAM CFG-UART1INPROT-RTCM3X 0          # write value 1          to item id 0x10750004 in layer 0
Flash CFG-UART1INPROT-RTCM3X 0          # write value 1          to item id 0x10750004 in layer 2
RAM CFG-UART1OUTPROT-UBX 1          # write value 0          to item id 0x10760001 in layer 0
Flash CFG-UART1OUTPROT-UBX 1          # write value 0          to item id 0x10760001 in layer 2
RAM CFG-UART1OUTPROT-NMEA 0          # write value 0          to item id 0x10760002 in layer 0
Flash CFG-UART1OUTPROT-NMEA 0          # write value 0          to item id 0x10760002 in layer 2
RAM CFG-UART1OUTPROT-RTCM3X 0          # write value 0          to item id 0x10760004 in layer 0
Flash CFG-UART1OUTPROT-RTCM3X 0          # write value 0          to item id 0x10760004 in layer 2
RAM CFG-UART2-BAUDRATE 0x70800        # write value 460800 0x70800 to item id 0x40530001 in layer 0
Flash CFG-UART2-BAUDRATE 0x70800        # write value 460800 0x70800 to item id 0x40530001 in layer 2
RAM CFG-UART2INPROT-UBX 0          # write value 0          to item id 0x10750001 in layer 0
Flash CFG-UART2INPROT-UBX 0          # write value 0          to item id 0x10750001 in layer 2
RAM CFG-UART2INPROT-NMEA 0          # write value 0          to item id 0x10750002 in layer 0
Flash CFG-UART2INPROT-NMEA 0          # write value 0          to item id 0x10750002 in layer 2
RAM CFG-UART2INPROT-RTCM3X 1          # write value 1          to item id 0x10750004 in layer 0
Flash CFG-UART2INPROT-RTCM3X 1          # write value 1          to item id 0x10750004 in layer 2

```

RAM CFG-UART2OUTPROT-UBX 0	# write value 0	to item id 0x10760001 in layer 0
Flash CFG-UART2OUTPROT-UBX 0	# write value 0	to item id 0x10760001 in layer 2
RAM CFG-UART2OUTPROT-NMEA 0	# write value 0	to item id 0x10760002 in layer 0
Flash CFG-UART2OUTPROT-NMEA 0	# write value 0	to item id 0x10760002 in layer 2
RAM CFG-UART2OUTPROT-RTCM3X 0	# write value 0	to item id 0x10760004 in layer 0
Flash CFG-UART2OUTPROT-RTCM3X 0	# write value 0	to item id 0x10760004 in layer 2
RAM CFG-USBOUTPROT-UBX 0	# write value 1	to item id 0x10780001 in layer 0
Flash CFG-USBOUTPROT-UBX 0	# write value 1	to item id 0x10780001 in layer 2
RAM CFG-USBOUTPROT-NMEA 0	# write value 0	to item id 0x10780002 in layer 0
Flash CFG-USBOUTPROT-NMEA 0	# write value 0	to item id 0x10780002 in layer 2
RAM CFG-USBOUTPROT-RTCM3X 0	# write value 0	to item id 0x10780004 in layer 0
Flash CFG-USBOUTPROT-RTCM3X 0	# write value 0	to item id 0x10780004 in layer 2
RAM CFG-MSGOUT-UBX_NAV_RELPOSNED_UART1 0x1	# write value 1 0x1	to item id 0x2091008e
Flash CFG-MSGOUT-UBX_NAV_RELPOSNED_UART1 0x1	# write value 1 0x1	to item id 0x2091008e
RAM CFG-NAVSPG-CONSTR_DGNSSTO 0x2	# write value 2 0x2	to item id 0x201100c4 in layer
Flash CFG-NAVSPG-CONSTR_DGNSSTO 0x2	# write value 2 0x2	to item id 0x201100c4 in layer

8.2 Python-Code GNSS-Kompass

```
import serial
from pyubx2 import UBXReader
from pyubx2 import UBXMessage

layers = 5 # Speicherebene: 1=RAM, 2=BBR, 4=Flash 5=RAM & FLASH
transaction = 0

def print_ubx_message(cfg, serial_out, typ):
    # Erstellen einer UBX-CFG-VALSET Nachricht
    msg = UBXMessage.config_set(layers, transaction, cfg)
    print('%s Konfiguration: %s'%(typ,msg))
    serial_out.write(msg.serialize())

def rover_config(rover_port_usb: str):
    # serielle Verbindung mit USB-Port für die Empfängerkonfiguration
    serial_out = serial.Serial(rover_port_usb, 38400, timeout=3)

    # Konfigurationsitems mit Wert
    cfg_data_rover = [
        ("CFG_RATE_MEAS", 200),
        ("CFG_UART1_BAUDRATE", 460800),
        ("CFG_UART1INPROT_UBX", 1),
        ("CFG_UART1INPROT_NMEA", 0),
        ("CFG_UART1INPROT_RTCM3X", 0),
        ("CFG_UART1OUTPROT_UBX", 1),
        ("CFG_UART1OUTPROT_NMEA", 0),
        ("CFG_UART1OUTPROT_RTCM3X", 0),
        ("CFG_UART2_BAUDRATE", 460800),
        ("CFG_UART2INPROT_UBX", 0),
        ("CFG_UART2INPROT_NMEA", 0),
        ("CFG_UART2INPROT_RTCM3X", 1),
```

```

        ("CFG_UART2OUTPROT_UBX", 0),
        ("CFG_UART2OUTPROT_NMEA", 0),
        ("CFG_UART2OUTPROT_RTCM3X", 0),
        ("CFG_USBOUTPROT_NMEA", 0),
        ("CFG_USBOUTPROT_RTCM3X", 0),
        ("CFG_USBOUTPROT_UBX", 0),
        ("CFG_MSGOUT_UBX_NAV_RELPOSNEED_UART1", 1),
        ("CFG_NAVSPG_CONSTR_DGNSSTO", 2)
    ]
    print_ubx_message(cfg_data_rover, serial_out, "Rover")

def moving_base_config(mb_port: str):
    serial_out = serial.Serial(mb_port, 38400, timeout=3)

    cfg_data_mb = [
        ("CFG_RATE_MEAS", 200),
        ("CFG_UART1_BAUDRATE", 460800),
        ("CFG_UART1INPROT_UBX", 1),
        ("CFG_UART1INPROT_NMEA", 0),
        ("CFG_UART1INPROT_RTCM3X", 0),
        ("CFG_UART1OUTPROT_UBX", 1),
        ("CFG_UART1OUTPROT_NMEA", 0),
        ("CFG_UART1OUTPROT_RTCM3X", 0),
        ("CFG_UART2_BAUDRATE", 460800),
        ("CFG_UART2INPROT_UBX", 0),
        ("CFG_UART2INPROT_NMEA", 0),
        ("CFG_UART2INPROT_RTCM3X", 0),
        ("CFG_UART2OUTPROT_UBX", 0),
        ("CFG_UART2OUTPROT_NMEA", 0),
        ("CFG_UART2OUTPROT_RTCM3X", 1),
        ("CFG_USBOUTPROT_NMEA", 0),
        ("CFG_USBOUTPROT_RTCM3X", 0),
        ("CFG_USBOUTPROT_UBX", 0),
    ]

    # erforderliche RTCM-Messages [MB Application Note]
    for rtcn_type in (
        "1074",
        "1084",
        "1094",
        "1124",
        "1230",
        "4072 0",
    ):
        corr_cfg_uart2 = f"CFG_MSGOUT_RTCM_3X_TYPE{rtcn_type}_UART2"
        cfg_data_mb.append([corr_cfg_uart2, 1])

    print_ubx_message(cfg_data_mb, serial_out, "Moving-Base")

# Übersetzen der empfangenen UBX-Nachricht und Daten in txt schreiben
# relPosHeading = Richtungdaten in Grad
def parse_write(file_name: str):
    stream = serial.Serial(rover_port_uart1, 460800, timeout=3)
    try:
        x = 0
        no_solution = 0
        float_solution = 0
        fix_solution = 0
        while True:

```

```

ubr = UBXReader(stream)
# UBXReader.read() liebt empfangene UBX-, NMEA- oder RTCM3-
Nachrichten
# raw_data = Binäre Rohdaten als Bytes
# parsed_data = übersetzte Rohdaten als UBX-, NMEA- oder RTCM3-
Nachricht
(raw_data, parsed_data) = ubr.read()
print(parsed_data)

# Einteilung in RTK-Modus: 0=keine Lösung, 1=RTK-float, 2=RTK-
fixed
try:
    f = open(file_name + '.txt', 'a')
    if parsed_data.carrSoln == 0:
        no_solution += 1
        f.write(
            str(x) + '\t' + str(parsed_data.relPosHeading) +
'\t' + str(parsed_data.carrSoln) + '\t' + str(
            no_solution) + '\n')
    elif parsed_data.carrSoln == 1:
        float_solution += 1
        f.write(
            str(x) + '\t' + str(parsed_data.relPosHeading) +
'\t' + str(parsed_data.carrSoln) + '\t' + str(
            float_solution) + '\n')
    elif parsed_data.carrSoln == 2:
        fix_solution += 1
        f.write(
            str(x) + '\t' + str(parsed_data.relPosHeading) +
'\t' + str(parsed_data.carrSoln) + '\t' + str(
            fix_solution) + '\n')
    x += 0.25
    f.close()
except AttributeError:
    pass
except KeyboardInterrupt:
    pass

# Eingabe USB-Port Moving-Base-Empfänger für die Konfiguration
def mb_com_port_input_usb() -> str:
    while True:
        mb_port = input('[+] Moving-Base Port eingeben (COMX-USB): ')
        try:
            serial.Serial(mb_port, 38400, timeout=3)
        except serial.SerialTimeoutException:
            print('[!] Timeout-Fehler')
            pass
        except serial.SerialException:
            print('[!] Port nicht gefunden')
            pass
        else:
            print('[!] Moving-Base USB COM-Port gefunden')
            return mb_port

# Eingabe USB-Port Rover-Empfänger für die Konfiguration
def rover_com_port_input_usb() -> str:
    while True:
        rover_port_usb = input('[+] Rover Port eingeben (COMX-USB): ')
        try:

```

```

        serial.Serial(rover_port_usb, 38400, timeout=3)
    except serial.SerialTimeoutException:
        print('[!] Timeout-Fehler')
        pass
    except serial.SerialException:
        print('[!] Port nicht gefunden')
        pass
    else:
        print('[!] Rover USB COM-Port gefunden')
        return rover_port_usb

# Eingabe UART1-Port Moving-Base-Empfänger für den Datenempfang
def rover_com_port_input_uart1() -> str:
    while True:
        rover_port_uart1 = input('[+] Rover Port eingeben (COMX-UART1): ')
        try:
            serial.Serial(rover_port_uart1, 460800, timeout=3)
        except serial.SerialTimeoutException:
            print('[!] Timeout-Fehler')
            pass
        except serial.SerialException:
            print('[!] Port nicht gefunden')
            pass
        else:
            print('[!] Rover UART1 COM-Port gefunden')
            return rover_port_uart1

# Textdatei erstellen
def create_file() -> str:
    while True:
        file_name = input('[+] Dateiname eingeben: ')
        try:
            f = open(file_name + '.txt', 'x')
        except FileExistsError:
            print('[!] Datei existiert bereits')
            pass
        else:
            print('[!] Datei wurde erstellt:')
            print(f.name)
            return file_name

while __name__ == '__main__':
    # Empfänger konfigurieren oder nur Daten empfangen
    answer1 = None
    while answer1 not in ('y', 'n'):
        answer1 = input('Empfänger konfigurieren? y/n: ')

    if answer1 == 'y':
        mb_port = mb_com_port_input_usb()
        print('[!] Moving-Base Empfänger USB-Port: ' + mb_port)
        rover_port_usb = rover_com_port_input_usb()
        print('[!] Rover Empfänger USB-Port: ' + rover_port_usb)
        print('[!] Empfänger konfigurieren...')
        rover_config(rover_port_usb)
        moving_base_config(mb_port)

    elif answer1 == 'n':
        rover_port_uart1 = rover_com_port_input_uart1()

```

```
print('[!] Rover Empfänger USB-Port: ' + rover_port_uart1)
file_name = create_file()
parse_write(file_name)
```