



**BERGISCHE
UNIVERSITÄT
WUPPERTAL**

Fachbereich E

Elektrotechnik, Informationstechnik, Medientechnik

Lehrstuhl für Allgemeine Elektrotechnik und Theoretische Nachrichtentechnik

Bachelor-Thesis

**Untersuchungen zur inertialsensorbasierten Positions- und
Bewegungsbestimmung mobiler Objekte**

Felix Schürmann
1110259
Elektrotechnik

Wuppertal, den 10.11.2015

Betreuer: Kathrin Kalischewski M.Sc. Erstgutachter: Prof. Dr.-Ing. Anton Kummert
Zweitgutachter: Prof. Dr. rer. nat. Ullrich Pfeiffer



BACHELOR-STUDIENGANG ELECTRICAL ENGINEERING

THEMA FÜR DIE BACHELOR-THESIS

Kandidat: **Felix Schürmann**

Matrikelnummer: **1110259**

Betreuer: **Prof. Dr.-Ing. A. Kummert**

Thema: Untersuchungen zur inertialsensorbasierten Positions- und Bewegungsbestimmung mobiler Objekte

Erläuterungen: Eine Möglichkeit zur Überwachung der Bewegung und/oder der Position von mobilen Objekten bietet der Einsatz von Inertialsensoren. Da diese Systeme möglichst kompakt gehalten werden sollen, um eine Beeinflussung der Objekte zu vermeiden, bietet sich ein Raspberry Pi – ein Computer in der Größe einer Kreditkarte – als Grundlage zur Kopplung mit einem Inertialsensor an.

Der Kandidat soll Möglichkeiten zur Bewegungs- und Positionsbestimmung eines beweglichen Objektes, ausgerüstet mit dem Raspberry Pi, untersuchen. Dazu soll zunächst der Stand der Technik dokumentiert und die Qualität der Daten der Bewegungssensoren untersucht werden. Es ist eine Applikation zu entwickeln, die in der Lage ist, die Sensordaten zur späteren Auswertung zu speichern. Aufbauend auf diesen Daten soll ein Verfahren entwickelt werden, welches eine Bewegungs- und Positionsbestimmung erlaubt. Die Ergebnisse sind in geeigneter Form zu dokumentieren und zu diskutieren.

Wuppertal, 10. August 2015

(Unterschrift)

Erstgutachter: **Prof. Dr.-Ing. A. Kummert**

Zweitgutachter: **Prof. Dr. rer. nat. U. Pfeiffer**

Prüfungsamt:

Kennziffer: **15 THB EE 227Ku**

Ausgabedatum: **10.08.2015**

Abgabedatum: **10.11.2015**

.....

(Unterschrift)

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die Arbeit selbstständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Wuppertal, den 10.11.2015

(Unterschrift)

Einverständniserklärung

Ich bin damit einverstanden, dass meine Abschlussarbeit wissenschaftlich interessierten Personen oder Institutionen zur Verfügung gestellt werden kann. Korrektur- oder Bewertungshinweise in meiner Arbeit dürfen nicht zitiert werden.

Wuppertal, den 10.11.2015

(Unterschrift)

Kurzfassung

In dieser Arbeit wurde eine inertiale Messeinheit mit einem Raspberry Pi gekoppelt. Es wurden Untersuchungen zur Qualität der gelieferten Daten durchgeführt. Weiterhin wurden Methoden vorgestellt um diese Daten zur Positions- und Bewegungsbestimmung zu nutzen. Algorithmen zur Neigungs- und Streckenberechnung wurden entwickelt sowie eine Schritterkennung. Hierfür wurden die Grundlagen und Funktionsweisen von Beschleunigungssensor, Gyroskop und Magnetometer erörtert um anhand ihrer Charakteristiken eine Optimierung der Positions- und Bewegungsbestimmung zu erzielen.

Abstract

In this paper an Inertial Measurement Unit(IMU) was coupled with a Raspberry Pi. The quality of the provided data was analysed. Furthermore methods for positioning and movement calculation were presented. Algorithms for position estimation and tilt sensing have been developed as well as a pedestrian reconing system. Therefor the basics and functionality of an accelerometer, gyroscope and magnetometer were discussed in order to optimize the calculation for positioning and movement.

Danksagung

An dieser Stelle möchte ich mich bei denjenigen bedanken, die mir diese Arbeit ermöglicht haben.

Ich danke Herrn Prof. Dr.-Ing. Anton Kummert für die Ermöglichung dieser Arbeit seitens des Lehrstuhls.

Ich bedanke mich bei Frau Kathrin Kalischewski, die meine Arbeit und mich betreut hat und mir das Thema der Arbeit vorgeschlagen hat. Ihre fortwährend engagierte Unterstützung brachte mich mehrmals dazu über meine Grenzen hinaus zu denken. Auch die investierte Zeit für Korrekturlesungen halfen mir, die noch verbleibenden Ungereimtheiten zu erkennen sowie zahlreiche Kommata und Rechtschreibfehler zu eliminieren.

Besonderer Dank gilt meinen Eltern, die mich während der Zeit meiner Arbeit in allen Belangen bestmöglich unterstützt haben.

Inhaltsverzeichnis

1 Einleitung	1
1.1 Motivation	1
2 Stand der Technik	2
2.1 Inertiale Navigation	2
2.2 Elektromagnetische Wellen	2
2.3 Bildverarbeitungsmethoden	3
2.4 Anwendungsgebiete von inertialbasierten Sensoren	4
3 Sensoren der inertialen Messeinheit	5
3.1 Beschleunigungssensor	5
3.1.1 Mechanischer Beschleunigungssensor	5
3.1.2 Solid State	5
3.1.3 Fehlerquellen im Beschleunigungssensor	8
3.2 Gyroskop	9
3.2.1 Fehlerquellen im Gyroskop	11
3.3 Magnetometer	11
3.3.1 Fehlerquellen im Magnetometer	12
4 Koordinatensysteme	14
5 Hardware	16
6 Implementierung	17
6.1 stationärer Zustand	17
6.2 Neigung	23
6.3 Drehung	27
6.4 Positionsbestimmung	31
7 Zusammenfassung	47
A Literaturverzeichnis	50
B Webquellen	52

1 Einleitung

1.1 Motivation

Die Entwicklung der MEMS(Micro-Electro-Mechanical-Systems) Technologie ermöglicht die kostengünstige Herstellung und kompakte Bauweise von inertialbasierten Sensoren wie Gyroskopen und Beschleunigungssensoren sowie Magnetometern. Um bessere Kontrolle über Flugzeuge, Raketen und Waffensysteme zu erlangen wurde dieses Forschungsgebiet, wie viele andere, durch militärischen Bedarf getrieben und findet sich heute in vielen Anwendungsgebieten wieder [1].

Eine IMU (Inertial Measurement Unit) bezeichnet in der Regel ein Gerät, das jeweils drei Beschleunigungssensoren, Gyroskope und Magnetometer beinhaltet, welche orthogonal zueinander angeordnet sind. Die Vorteile der MEMS Technologie liegen in ihrer Funktionalität, geringem Stromverbrauch sowie geringem Gewicht und Abmaßen.

Die inertialbasierte Navigation beruht auf Kenntnis von Beschleunigung, Geschwindigkeiten sowie der Startposition und Messungen von Flugstellungen und folgt Newtons Gesetzen der klassischen Mechanik. Es ist die einzige Art der Navigation, die sich nicht über äußere Referenzpunkte realisiert [2]. Inertialbasierte Navigation kann mit andersartigen Referenzpunkten gekoppelt werden um Ergebnisse der Positionsbestimmung zu verbessern. In Gebieten, die von äußeren Referenzsignalen wie GPS abgekoppelt sind, wie etwa in Innenräumen, bieten die inertialbasierten Sensoren die Möglichkeit zur Positionsbestimmung und damit zur Koordination von mobilen Objekten. Eine Positionsbestimmung in Innenräumen kann beispielsweise die Navigation innerhalb eines Museums oder Einkaufszentrums ermöglichen sowie eine Unterstützung für in der Mobilität eingeschränkte Personen bieten. Ziel dieser Arbeit soll es sein die Möglichkeiten und Herausforderungen dieser Positionsbestimmung im Hinblick auf die Qualität der Daten zu untersuchen.

2 Stand der Technik

Dieser Abschnitt zeigt in Kurzform die verschiedenen Möglichkeiten zur Positionsbestimmung. Sie beruhen darauf physikalische Größen oder deren Änderung im Zusammenhang mit einer geographischen Position zu messen [6].

2.1 Inertiale Navigation

Inertiale Navigation beruht auf Messung der Bewegung und deren Richtung, um die Position ausgehend vom Startpunkt zu bestimmen. Inertiale Navigation kann mittels Koppelnavigation (*engl. dead reckoning*) die Position, Orientierung und Geschwindigkeit berechnen. Hierfür werden Beschleunigungssensoren, Gyroskope und Magnetometer verwendet die für eine korrekte Funktion eine hohe Güte aufweisen müssen, da die Position durch den Beschleunigungssensor über ein Doppelintegral berechnet wird [6].

2.2 Elektromagnetische Wellen

Eine Positionsbestimmung kann mittels sichtbarem, sowie unsichtbarem Licht erreicht werden. Triangulation und Trilateration sind Möglichkeiten um die Position relativ zu Referenzpunkten wie Satelliten, WiFi-Zugangspunkten oder GSM Stationen herzustellen. Bei der Triangulation wird durch Winkelmessungen innerhalb von Dreiecken eine Abstandsbestimmung durchgeführt. Die Trilateration beruht auf einer Messung der Abstände zu drei Punkten um eine Positionsbestimmung durchzuführen.

Satellitenbasiert

Die meistverwendete Technik zur Positionsbestimmung ist heutzutage das in den USA entwickelte GPS (Global Positioning System). Russland und Europa stellen alternativ auch satellitenbasierte Navigationssysteme bereit(GLONASS und GALILEO) [6]. Diese Technik benötigt zur korrekten Funktion eine Sichtverbindung der Satelliten zum Empfänger. Daher ist sie ungeeignet für Anwendung in Innenräumen.

Radiowellen

Kommunikationsnetze wie WiFi, Bluetooth oder Zigbee können verwendet werden um die Position über den RSSI (Received Signal Strength Indicator) zu bestimmen. Weil Radiowellen bei der Ausbreitung dem Abstandsgesetz, das die Strahlungsintensität bei Ausbreitung von magnetischen Wellen im Raum beschreibt, unterliegen, kann der Abstand zum Sender durch die Beziehung zwischen der gesendeten und empfangenen Signalstärke abgeschätzt werden. Absorptionen, Reflektionen sowie nicht-stationäre Objekte wie Türen, Personen oder Einrichtungsgegenstände verursachen eine schwer zu kalkulierende Signalstärkendynamik. Daten der Signalstärken sind sehr verrauscht und daher Thema aktueller Forschung mit dem Ziel falsche Daten zu filtern [7].

RFID (Radio Frequency Identification) bietet eine weitere Möglichkeit, die allerdings eine preparierte Umgebung erfordert [6]. Ein RFID System besteht aus einem Transponder und einem gekennzeichneten Code. RFID Transponder können in der Größe eines Reiskorns hergestellt werden und bieten die Möglichkeit innerhalb weniger Meter den Erkennungscode zu übertragen.

Derzeit wird das RF Capture System am MIT entwickelt, das die Erkennung von Körperbewegungen und Gesten und die Indoor Positionsbestimmung revolutionieren könnte. Anwendungen virtueller Realität sowie Altenbetreuung und Überwachung ließen sich mit Hilfe von WiFi Signalen realisieren. Hierbei werden die Signalreflektionen gemessen, die ein Körper erzeugt, um personenspezifische Abbilder zu erstellen. Eine enorme Anzahl an Anwendungsmöglichkeiten könnte durch diese Technologie erreicht werden, die u.a. auch einige Aufgabenbereiche einer inertialen Messeinheit obsolet machen [24].

2.3 Bildverarbeitungsmethoden

Optische Bildinformationen von Kameras können genutzt werden, um umgebungsabhängige Muster zu erkennen. So können Markierungen, wie z.B. Barcodes, mit Positionsdatenbanken verknüpft sein. Weiterhin können besondere Merkmale im Bildstream erkannt werden. Diese Methode fordert allerdings große Rechenleistung [6].

2.4 Anwendungsgebiete von inertialbasierten Sensoren

Inertialbasierte Messsensoren werden heute unter anderem in folgenden Anwendungsbereichen eingesetzt:

- Indoor Navigation
- Schrittzählung, Klassifizierung von Bewegungsarten
- Steuerung von Flugzeugen, Marschflugkörpern, Drohnen
- Fahrzeugtechnik: Airbags, elektronische Dämpfungsteuerung, Stabilitätssysteme
- Ausrichtung von Maschinen und Antennen
- Unterhaltungselektronik: Augmented Reality, Bildstabilisierung, 3D Computermäuse
- Transportüberwachung von Frachtgut

3 Sensoren der inertialen Messeinheit

In einer inertialen Messeinheit werden meist Beschleunigungssensoren, Gyroskope und Magnetometer mit jeweils drei Freiheitsgraden verwendet. In diesem Kapitel werden die Funktionsweisen der Sensoren und ihre Fehlerquellen betrachtet.

3.1 Beschleunigungssensor

Man unterscheidet zwei Klassen von Beschleunigungssensoren: mechanisch oder Solid State Geräte [4]. Jede der drei Achsen des Beschleunigungssensors liefert die gemessene Beschleunigung. Die Erdgravitation ist bei jeder Messung enthalten. Daher ist die gemessene Beschleunigung nicht zwingend der Beschleunigung in einem globalen Koordinatensystem entsprechend und muss anhand der Neigung des Gerätes berechnet werden (siehe Kapitel: 4). Um die durch eine Bewegung verursachte Beschleunigung zu bestimmen muss der Gravitationsoffset heraus gerechnet werden. Verschiedene Sensoren nutzen unterschiedliche Methoden zur Datengewinnung. Diese umfassen u.A. Spannungsmessungen mit Piezo-Kristallen oder Messungen von Kapazitätsänderungen.

3.1.1 Mechanischer Beschleunigungssensor

Der mechanische Beschleunigungssensor wie in Abbildung 3.1 besteht aus einer von Federn eingespannten Masse. Bei Bewegung der Masse kann durch Messung der Auslenkung die Beschleunigung mittels Newtons 2. Gesetz berechnet werden [4]:

$$F = m \cdot a \quad (3.1)$$

3.1.2 Solid State

Solid State Beschleunigungssensoren werden in verschiedene Untergruppen unterteilt. Hierzu zählen silikon-, quarz basierte Geräte und Oberflächenwellen Sensoren [4]. Der Oberflächenwellen Beschleunigungssensor(siehe Abbildung: 3.2) besteht aus einem Kragarm, der mit einer bestimmten Frequenz angeregt wird. Die Masse am

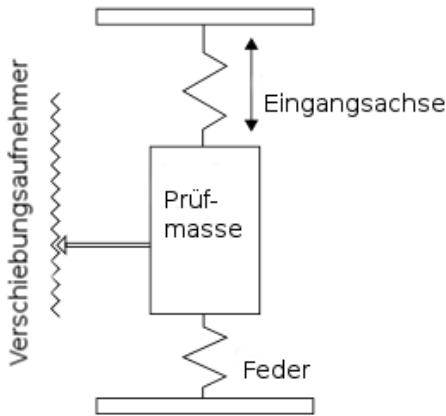


Abbildung 3.1: Mechanischer Beschleunigungssensor [4]

Ende des Kragarms ist frei beweglich. Beim Auftreten einer Beschleunigung wird der Kragarm gebogen, was zu einer Frequenzänderung führt, die in Abhängigkeit zu dieser steht. MEMS Sensoren können mit Solid State und mechanischen Sensoren realisiert werden. Anwendungsbezogen muss ein Sensortyp gewählt werden. Es unterscheiden sich über Kapazitätsänderung betriebene Sensoren, wie in der Mikroskopaufnahme in Abbildung 3.3 zu sehen, zu piezoelektrischen und piezoresistiven Sensoren in ihrer Überlegenheit im niederfrequenten Bereich, dem Rauschverhalten, hoher Empfindlichkeit, niedrigem Drift und geringerer Anfälligkeit gegenüber Temperaturschwankungen.

Piezoelektrische Sensoren sind sehr stabil, benötigen aber eine Kräftezufuhr einer bestimmten Grenzfrequenz, da bei Unterschreitung dieser kein akzeptables Ausgangssignal vom Sensor erzeugt wird. In diesem Bereich kommt es zu starken Dämpfungen und Phasenverschiebungen, die die Anwendungsbereiche limitiert [8].

Beispielsweise dürfen Messdaten für Robotersteuerungen und medizinische Anwendungen keine Offset Verschiebungen enthalten, da dies zu groben Fehlern in der Positionsbestimmung bedingt durch die numerische Integration führt [8].

Piezoresistive Sensoren eignen sich durch ihre geringen Abmaße und niedrigen Preis vor allem zur Implementierung in einfachen Erkennungsmechanismen. Sie bieten annehmbare Ergebnisse im niederfrequenten Bereich und bei stationären Zuständen, sind aber im betreibbaren Temperaturbereich begrenzt [8]. Leckströme begrenzen den nutzbaren Temperaturbereich für piezoresistive Sensoren auf bis zu 150°[14].

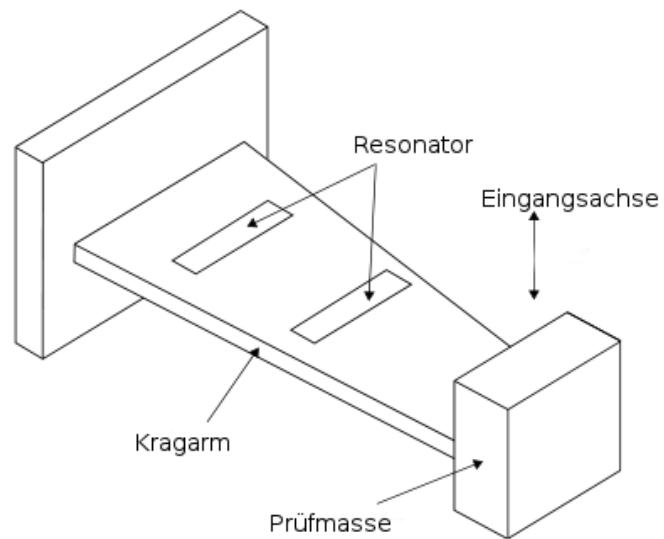


Abbildung 3.2: Oberflächenwellen Sensor [4]

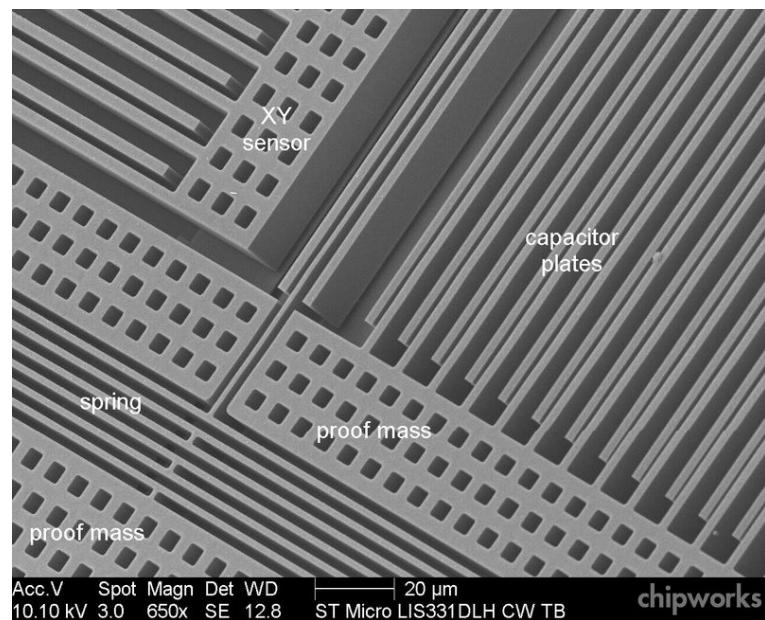


Abbildung 3.3: Kapazitiver MEMS Beschleunigungssensor [16]

3.1.3 Fehlerquellen im Beschleunigungssensor

In diesem Abschnitt werden die Fehler und deren Ursachen im Beschleunigungssensor diskutiert.

Konstante Messabweichung

Die konstante Messabweichung (*engl. Bias*) ist eine Fehlerquelle des Beschleunigungssensors mit sehr hoher Bedeutung. Der Bias ist der Offset des Ausgangssignals von seinem Wert in der Realität. Ein konstanter Bias des Fehlers ϵ verursacht einen mit der Zeit quadratisch steigenden Fehler in der Positionbestimmung, bedingt durch das Doppelintegral. Der Fehler bestimmt sich zu:

$$s(t) = \epsilon * \frac{t^2}{2} \quad (3.2)$$

Mit t als Zeit der Integration [3].

Der Bias kann mit Hilfe einer Langzeitmessung bestimmt werden indem das Mittel der Abweichung im ruhenden Gerät aufgenommen wird. Nicht korrigierte Bias Fehler sind die typische Fehlerquelle, die die Leistung des Gerätes limitieren [4].

weitere Fehlerarten

Zu den weiteren Fehlerquellen im Beschleunigungssensor zählt das thermo-mechanische weiße Rauschen. Das Ausgangssignal des Sensors ist mit diesem überlagert und verursacht bei Integration einen Random Walk, dessen Standardabweichung proportional zu \sqrt{t} anwächst. Der Random Walk ist ein mathematisches Modell bei dem die einzelnen Schritte zufällig erfolgen [4].

Der Bias des Sensors unterliegt zudem dem Funkelrauschen, das eine Änderung des Bias über Zeit verursacht. Es stellt sich als Rauschen mit einer inversen Spektralen Leistungsdichte bzw. einer $1/f$ Charakteristik dar. Die Effekte des Funkelrauschens können überlicherweise bei niedrigen Frequenzen in Bauelementen beobachtet werden. Im hochfrequenten Spektrum wird es durch das weiße Rauschen überlagert [4]. Ursächlich für das Funkelrauschen sind Transistoren bei denen es durch verschiedene Effekte zu Widerstandsschwankungen kommt.

Temperaturänderung der Umgebung sowie Eigenerwärmung verursachen zudem eine Änderung des Bias. Die Beziehung zwischen Temperatur und Änderung des Bias ist häufig nicht linear. Viele IMUs haben daher einen Temperatursensor eingebaut, der zur Korrektur eingesetzt wird.

Zudem verursachen Ungenauigkeiten bei der Herstellung der Sensoren Bias Fehler, die nur während einer angelegten Beschleunigung in Erscheinung treten [4].

3.2 Gyroskop

Ein konventionelles Gyroskop (siehe Abbildung: 3.4) dient zur Messung der Ausrichtung und besteht aus einem drehenden Rad in einer kardanischen Aufhängung, welches eine Rotation in drei Achsen erlaubt. Die Drehimpulserhaltung des Rades sorgt dafür, dass das Rad einer Rotation des Gesamtgerätes widersteht. Somit bleibt die globale Ausrichtung des Rades konstant und die Winkel zwischen den kardanischen Aufhängungen unterliegen einer Änderung. Ein konventionelles Gyroskop misst die Orientierung direkt, wohingegen ein MEMS Gyroskop die Drehrate misst und daher auch als Drehratensensor bezeichnet wird [3].

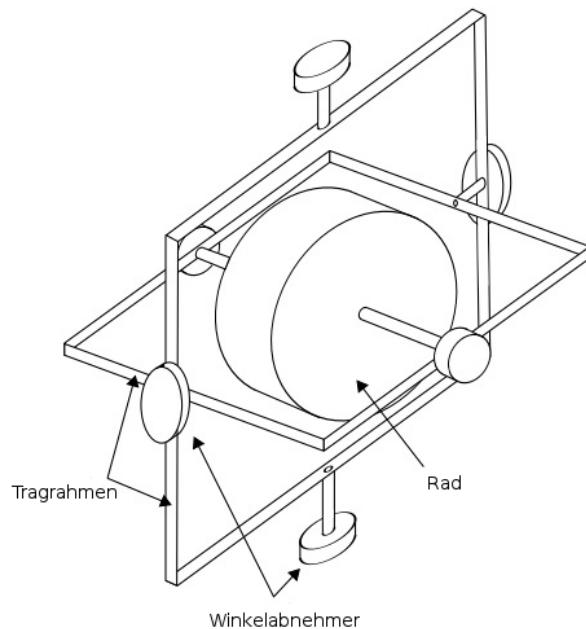


Abbildung 3.4: Konventionelles Gyroskop [4]

Der Faserkreisel ist ein weiterer Gyroskop Typ und beruht auf der Interferenz von Lichtstrahlen, die durch eine große Spule, bestehend aus fiberoptischem Kabel, geleitet werden. Um die Winkelgeschwindigkeit zu messen werden Lichtstrahlen in entgegengesetzte Richtung abgegeben. Wenn der Sensor rotiert, muss der in Drehrichtung laufende Lichtstrahl einen längeren Weg zurücklegen als der Strahl in Gegenrichtung. Beim Verlassen des Lichtwellenleiters interferieren die Strahlen aufgrund ihrer Phasenverschiebung. Dies führt zu einer Änderung der Lichtintensität, die auf die Drehrate zurückzuführen ist. Diese Eigenschaft ist als Sagnac Effekt bekannt [4].

Die Vorteile der optischen Gyroskope bestehen in ihrer Genauigkeit, die von der Länge des Lichtwellenleiters abhängt und in ihrer kurzen Anlaufzeit.

Konventionelle, Lichtwellenleiter und Ring Laser Gyroskope haben eine große Zahl an Anwendungsgebieten ermöglicht. Diese Bauteypen sind aber meist zu groß und kostenintensiv für viele heutige Anforderungen. Trotzdem werden für Anwendungen, die hohe Präzision erfordern, noch große Stückzahlen dieser klassischen Gyroskope hergestellt [4].

$$\vec{F}_c = -2m(\vec{\omega} \times \vec{v}) = m \cdot \vec{a}_c \quad (3.3)$$

Das Funktionsprinzip der MEMS Gyroskope liegt der Coriolisbeschleunigung \vec{a}_c zugrunde (Formel: 3.3). Darin ist $\vec{\omega}$ die Winkelgeschwindigkeit des Bezugssystems, \vec{v} die Geschwindigkeit des Körpers relativ zu diesem Bezugssystem und m die Masse des Körpers. Die Corioliskraft ist eine Trägheitskraft, die sich als Ergebnis von Newtons Grundgesetzen der Bewegung angewendet auf rotierende Rahmen ergibt. Die Prüfmasse des Gyroskops vibriert in einer Achse. Bei Anlegen einer Winkelbeschleunigung wird nach dem Corioliseffekt die Prüfmasse zusätzlich in einer anderen Achse vibrieren. Diese zweite Vibration ist proportional zur angelegten Winkelgeschwindigkeit [9].

Die meisten MEMS Gyroskope nutzen eine Konfiguration mit Stimmgabeln, die wie in Abbildung 3.5 die beiden Prüfmassen verbindet [15].

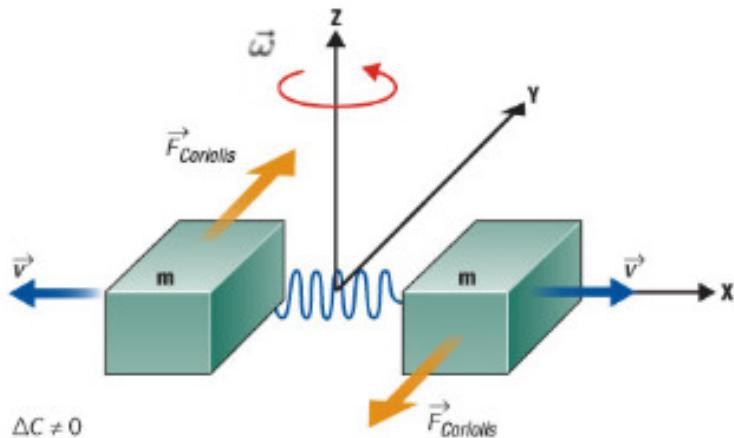


Abbildung 3.5: Funktionsweise des Gyroskops [15]

Wie in Abbildung 3.5 zu sehen, oszillieren zwei Massen konstant in verschiedene Richtungen. Bei Anlegen einer Winkelgeschwindigkeit wirkt daher die Corioliskraft auf beide Massen in unterschiedliche Richtungen. Dies führt zu einer Kapazitätsänderung, die der Winkelgeschwindigkeit proportional ist [15]. Bei einer linearen Beschleunigung des Sensors bewegen sich beide Massen in die gleiche Richtung und

es entsteht keine Kapazitätsänderung. Daher sind MEMS Gyroskope nicht anfällig gegenüber linearer Beschleunigung, Neigung oder Vibrationen [15].

3.2.1 Fehlerquellen im Gyroskop

Die Arten der Fehler im Gyroskop ähneln denen des Beschleunigungssensors und werden in Kapitel 3.1.3 genauer besprochen. Ein wichtiger Unterschied ist, dass Fehler im Beschleunigungssensor doppelt integriert werden, wohingegen die Drehraten nur einmal integriert werden um die Position zu bestimmen [4]. Daher wächst der Winkelfehler linear bei konstantem Bias ϵ , zu sehen in Formel 3.4.

$$\theta(t) = \epsilon * t \quad (3.4)$$

Weitere Fehlerquelle im Gyroskop ist das weiße Rauschen, dessen Standardabweichung mit \sqrt{t} anwächst. Außerdem ergeben sich Temperatureffekte, Kalibrierungsfehler und ein unstabiler Bias wie beim Beschleunigungssensor [4].

3.3 Magnetometer

Magnetometer sind Instrumente zur Messung von magnetischen Feldstärken und deren Richtungen. Hierfür können Effekte der Lorentz Kraft, Hall Effekte (Formel: 3.5) oder Änderungen von Spannungen und Resonanzfrequenzen genutzt werden.

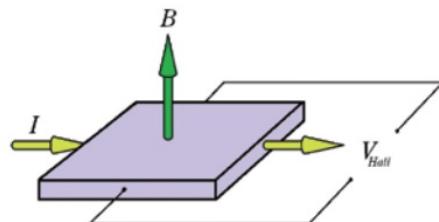


Abbildung 3.6: Hall Effekt [10]

$$V_{Hall} = \frac{I \cdot B}{n \cdot e \cdot d} \quad (3.5)$$

Hierbei ist I die Stromstärke, B die magnetische Feldstärke, n die Dichte der freien Ladungsträger, d die Dicke der Metallplatte und e die Elektronenladung [17].

Die magnetischen Feldlinien der Erde zeigen zu den magnetischen Polen der Erde, welches genutzt werden kann, um die Orientierung im Raum zu bestimmen. Der

Hall-Effekt ist die meistverwendete Messmethode in Magnetometern. Er beruht auf dem Prinzip, dass eine Spannung V_{Hall} über einem metallischen Element gemessen werden kann wenn dieses orthogonal zu einem magnetischen Feld steht. Durch die vom Magnetfeld verursachte Lorentzkraft kommt es zu einer Auslenkung der Elektronen senkrecht zur Bewegungsrichtung. Dies führt zu einer Ladungstrennung vergleichbar mit einem Kondensator. Diese Ladungstrennung kann als Spannung abgegriffen werden und wird als Hall-Spannung bezeichnet. Die Nachteile der Hall-Effekt Methode sind die kleinen Ausgangssignale und relative niedrige Empfindlichkeit, sowie die Stabilität bei Temperaturänderungen. Dafür sind diese Sensoren in kleiner Größe, mit geringem Stromverbrauch und niedrigen Kosten herstellbar [10].

3.3.1 Fehlerquellen im Magnetometer

Signifikante Fehlerquellen bei MEMS Magnetometern sind Empfindlichkeit, nahegelegene Materialien aus Eisen, Neigung des Sensors, magnetische Deklination und Temperatur [11].

Magnetische Empfindlichkeit des Sensors

In horizontaler Ebene bewegen sich die Feldstärken üblicherweise im Bereich von 200-300 mG. Um eine 0.1° Auflösung zu erreichen, benötigt man eine Empfindlichkeit höher als 0.35 mG. Durch Bauteilgenauigkeiten variiert die Empfindlichkeit des Magnetometers zwischen den verschiedenen Achsen. Jede Achse benötigt also ihren eigenen Normierungsfaktor [11].

Materialien aus Eisen

Nahegelegene eisenhaltige Materialien stören durch ihr lokales Magnetfeld die Messung des globalen Magnetfeldes. Eisenhaltige Materialien werden in zwei Gruppen unterteilt: hartes und weiches Eisen. Als hartes Eisen werden Materialien mit hohen Koerzitivfeldstärken bezeichnet. Die Koerzitivfeldstärke ist die benötigte magnetische Feldstärke, um eine bis zur Sättigungsflussdichte aufgeladene ferromagnetische Substanz zu entmagnetisieren. Das Störfeld des harten Eisens bleibt konstant in fixer Position zum Magnetmeter für alle Richtungen. Es führt daher zu einer konstanten Addition auf die Feldstärken des Erdmagnetfeldes und kann später bestimmt und herausgerechnet werden. Weiche Eisen verdichten die Magnetlinien zu sich selbst, was zu ausrichtungsabhängigen Fehlern führt [11].

Neigung des Sensors

Die magnetische Ausrichtung wird als Winkel zwischen der lokalen horizontalen Ausrichtung im Uhrzeigersinn zum magnetischen Norden gemessen. Für eine korrekte

Bestimmung des Ausrichtungswinkels muss das Gerät horizontal positioniert sein. Neigen des Sensors verringert die Genauigkeit. Der Roll-Winkel Φ und Nick-Winkel Θ gegeben von Beschleunigungssensor und Gyroskop können genutzt werden, um die magnetischen Komponenten bei Neigung auszugleichen [11]. Roll- und Nickwinkel werden in Kapitel 4 beschrieben.

Deklinationswinkel

Der Deklinationswinkel hängt von der Position des Sensors auf der Erde ab und kann aus Karten und Tabellen bestimmt werden. Der Fehler leitet sich aus dem Unterschied zwischen dem magnetischen Nordpol und dem geografischen Norden ab [11].

Temperatur

Empfindlichkeit und Offset des Magnetometers ändern sich im Zusammenhang mit einer Änderung der Temperatur. Diese verursacht zwei Fehlerarten. Einen Drift des Offsets, der bis $0,3^\circ$ erreichen kann sowie einen Empfindlichkeitsparameter, der sich als Änderung des Verstärkungsfaktors für das Augangssignal bemerkbar macht. Dieser Ausrichtungsfehler hat keinen linearen Zusammenhang zur Temperaturänderung [11].

4 Koordinatensysteme

Ein Koordinatensystem ist zur Bezeichnung von Punkten und Objekten im geometrischen Raum unabdingbar. Die Roll-, Nick- und Gier Winkel sind Bezeichnungen für Lagewinkel im Eulerschen Raum und wurden zunächst für die Orientierung von Luftfahrzeugen verwendet. Mittlerweile finden sie auch Verwendung für die Orientierung bei Wasser-, Land- und Raumfahrzeugen. Für eine Navigation ist zwischen einem körperfesten und erdfesten Koordinatensystem zu unterscheiden.

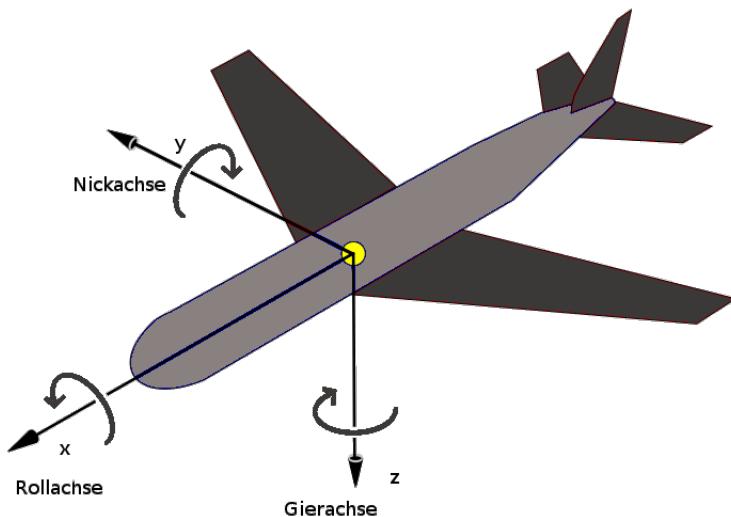


Abbildung 4.1: Roll-Nick-Gier Achse [20]

Die Roll-, Nick- und Gier Achse sind in Abbildung 4.1 dargestellt.

- Gieren (*engl. yaw*): Drehung um die z-Achse des Referenzsystems. Hier bezeichnet mit Ψ
- Nicken (*engl. pitch*): Drehung um die y-Achse. Hier bezeichnet mit Θ .
- Rollen (*engl. roll*): Drehung um die x-Achse. Hier bezeichnet mit Φ .

Koordinatentransformation

Um einen Punkt oder Vektor vom erdfesten zum körperfesten Koordinatensystem und andersherum zu transformieren wird eine Rotationsmatrix verwendet, um eine Drehung im euklidischen Raum zu beschreiben. Sie besteht aus trigonometrischen Ausdrücken des Drehwinkels, sodass bei Multiplikation mit einem Vektor eine Drehung um den Winkel bewirkt wird [21]. Für ein korrektes Umrechnen des Koordinatensystems muss eine Konvention eingehalten werden. Die Konvention begründet die Reihenfolge der zu drehenden Achsen. Es gibt zwei Möglichkeiten eine Drehung durchzuführen. Entweder werden die Achsen jeweils um die erdfesten Roll-, Nick- und Gier Winkel gedreht oder es werden Euler Winkel genutzt, um wie in Abschnitt 4 bereits gedrehte Achsen weiter zu drehen.

Gier-Nick-Roll Konvention

Zunächst wird mit dem im erdfesten System gemessenen Gierwinkel die z-Achse gedreht. Es entsteht die Knotenachse N(y'). Danach wird mit dem gegen die Erdoberfläche (x-y-Ebene) gemessenen Nickwinkel um die Knotenachse N(y') gedreht. Hierbei entsteht die fahrzeugfeste x-Achse. Zuletzt wird mit dem Rollwinkel die fahrzeugfeste x-Achse gedreht [21]. Hier entsteht die fahrzeugfeste y- und z-Achse. Die drei Drehmatrizen für die Einzeldrehungen lauten:

$$M_G = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{pmatrix}$$

$$M_N = \begin{pmatrix} \cos\Theta & 0 & -\sin\Theta \\ 0 & 1 & 0 \\ \sin\Theta & 0 & \cos\Theta \end{pmatrix}$$

$$M_R = \begin{pmatrix} \cos\Psi & \sin\Psi & 0 \\ -\sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$M_{GNR} = \begin{pmatrix} \cos\Theta\cos\Psi & \cos\Theta\cos\Psi & -\sin\Theta \\ \sin\Phi\sin\Theta\cos\Psi - \cos\Phi\sin\Psi & \sin\Phi\sin\Theta\sin\Psi + \cos\Phi\cos\Psi & \sin\Phi\cos\Theta \\ \cos\Phi\sin\Theta\cos\Psi + \sin\Phi\sin\Psi & \cos\Phi\sin\Theta\sin\Psi - \sin\Phi\cos\Psi & \cos\Phi\cos\Theta \end{pmatrix}$$

Mit der Transponierten Drehmatrix $M_{GNR}^\top = M_{RNG}$ können die im körperfesten System gemessenen Größen in ein erdfestes Koordinatensystem umgerechnet werden [21].

5 Hardware

Die verwendete inertiale Messeinheit 10-DOF IMU Breakout von Adafruit (siehe Abbildung: 5.1) enthält folgende Sensoren : L3GD20H 3-Achsen Gyroskop, LSM303 3-Achsen Kompass, LSM303 3-Achsen Beschleunigungssensor, BMP180 Druck und Temperatursensor [22].



Abbildung 5.1: Adafruit 10-DOF IMU Breakout [22]

Die inertiale Messeinheit wird über eine I2C Schnittstelle angesteuert. I2C ist ein serieller Datenbus mit Master-Slave System. Der Datentransfer wird immer vom Master initiiert. Der Master wird in diesem Fall durch den Raspberry Pi gegeben. Dieser gibt die Taktgeschwindigkeit vor und kann im Ultra-Fast-mode bis zu 5,0 Mbit/s übertragen. I2C benötigt zwei Signalleitungen. Eine Takteleitung (SCL) und eine Datenleitung (SDA). Diese werden mit den GPIO-Ports des Raspberry-Pi verbunden.

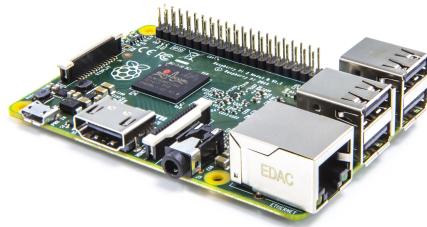


Abbildung 5.2: Raspberry Pi 2 Model B [23]

Der Raspberry Pi 2 ist ein Computer in Debitkartengröße. Er verfügt über einen 900Mhz quad core ARM Prozessor, 40 GPIO Ports, einen HDMI Anschluss, 4 USB Ports und einige weitere Schnittstellen. In diesem Fall wurde Raspbian, ein Unix-basiertes Betriebssystem, das auf einer tauschbaren SD-Karte liegt, verwendet.

6 Implementierung

Um eine Erprobung der inertialen Messeinheit durchzuführen, wurde dieses auf ein Steckbrett gelötet und mit den I2C Ports des Raspberry Pi verbunden. Grundlagen für das Verkabeln der Hardware, ein Programm zum Auslesen des I2C Bus, sowie erste Berechnungen wurden durch ein Web-Tutorial [18] gegeben und für die verwendeten Sensoren angepasst.

6.1 stationärer Zustand

Zunächst wurde der Sensor flach auf dem Tisch platziert, um die Auswirkungen der Gravitationskraft und den Offset zu untersuchen.

Der Beschleunigungssensor gibt einen Wert abhängig der gewählten Maßstabsskala aus. Für die größte Auflösung wurde eine Skala von $\pm 16g$ gewählt [13].

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix} \quad (6.1)$$

Wobei g die Erdbeschleunigung $9.81m/s^2$ ist.

Im ruhenden Zustand ergeben sich die Beschleunigungen auf die Achsen anhand der Formel 6.1. Das dazugehörige Koordinatensystem ist aus 4.1 zu entnehmen. Abbildung 6.1 zeigt die Beschleunigung der z-Achse pendelnd um den Wert 16238, der als Referenzwert zur Erdbeschleunigung genommen werden kann. Für Tabelle 6.1 wurde wegen der $\pm 16g$ Skala 16000 als Referenzwert zu $9.81m/s^2$ gewählt. Die Statistik ergibt sich aus zwei aufgenommenen Messreihen. Die Reichweite des Rauschens beträgt in der z-Achse bis $0.9 m/s^2$. In der Messreihe in Tabelle 6.1 über $1m/s^2$. Bei einer Positionsbestimmung mit Doppelintegration verursacht dies einen beachtlichen Fehler.

Die Standardabweichung für einen Vektor A bestehend aus N Beobachtungen ist definiert als:

$$S = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |A_i - \mu|^2} \quad (6.2)$$

μ ist der Durchschnitt von A:

$$\mu = \frac{1}{N} \sum_{i=1}^N A_i \quad (6.3)$$

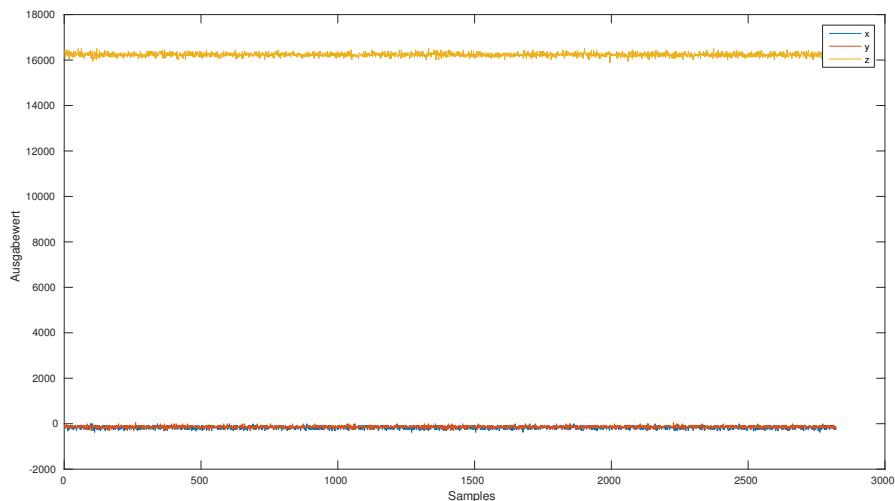


Abbildung 6.1: Rohwerte des ruhenden Beschleunigungssensors

Achse	Durchschnitt	Maximalwert	Minimalwert	Standardabweichung
X[m/s ²]	-0,10839	0	-0,23543	0,03780
Y[m/s ¹]	-0,08567	0,03923	-0,19619	0,03442
Z[m/s ²]	9,95585	10,12385	9,731414	0,05202
Gesamt[m/s ²]	9,76094	10,16308	9,299874	

Tabelle 6.1: Beschleunigungssensor Statistik

Beim ruhenden Gyroskop treten Offsets auf. Der Sensor wurde mit einer Empfindlichkeit von 2000 dps(degrees per second) betrieben. Dabei ist diese Empfindlichkeit die maximale Winkeländerung, die der Sensor messen kann. 360 dps bedeuten 60 RPM (revolutions per minute) und damit 1 Umdrehung pro Sekunde. Ein Umrechnungsfaktor bringt die Werte auf °/s wie in Abbildung 6.2.

Wie in Tabelle 6.1 zu sehen, unterscheidet sich der Offset der y-Achse von den anderen Achsen. Die Standardabweichung durch das Rauschen befindet sich im Bereich von 0,1°/s.

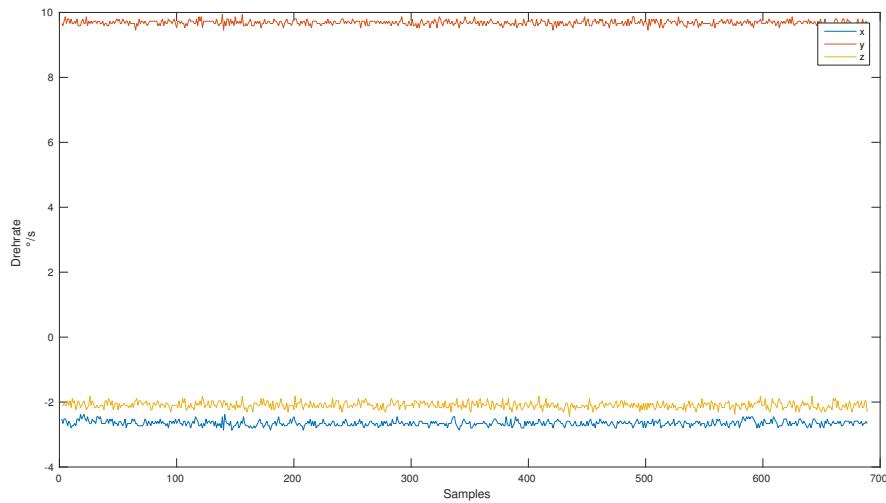


Abbildung 6.2: Rohwerte des ruhenden Gyroskops

Achse	Durchschnitt	Maximalwert	Minimalwert	Standardabweichung
X[°/s]	-2,6509	-2,3802	-2,8701	0,0849
Y[°/s]	9,4501	9,9400	9,4500	0,0770
Z[°/s]	-2,0997	-1,8200	-2,4500	0,1028
Gesamt[°/s]	14,2007	14,14308	14,7701	

Tabelle 6.2: Gyroskop Statistik

Um die Neigungen des Sensors zu bestimmen wurden die Drehraten mit der Durchlaufzeit des Programms integriert:

```
gyroXangle+=rate_gyr_x*DT;
```

In Abbildung 6.3 zu sehen, folgt aus der Integration ohne Offset Korrektur ein Drift der Winkel in Ruhelage. Dieser bedeutet für die y-Achse nach 600 Samples bereits einen Winkel von 120 °. Dies entspricht bei einer Loop Zeit von 20 ms einer Zeit von 12 Sekunden bzw. einem Drift von 10°/s.

Um den Offset des Gyroskops und des Beschleunigungssensors zu bestimmen wurde eine Funktion geschrieben, die eine bestimmte Anzahl an Sensorwerten liest und daraus den Mittelwert bildet. Wie in Abbildung 6.5 wird dieser konstant von der gemessenen Beschleunigung abgezogen. Um die Rohwerte in Beschleunigungen in m/s^2 umzurechnen benötigt es einen Umrechnungsfaktor. Dieser wird anhand des Offsets der z-Achse als Referenz zur Erdbeschleunigung gebildet.

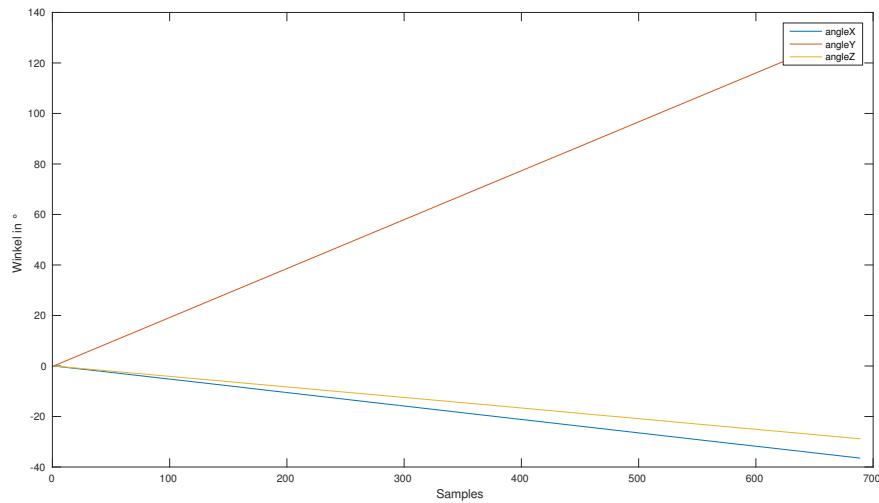


Abbildung 6.3: Drift des Gyroskops

```

do {
    readACC(accRaw);
    calibrateAcc(accRaw[0], accRaw[1], accRaw[2]);
    countx++;
} while(countx <=1024);
    offset_acc_x= vel_acc_x/1024;
    offset_acc_y= vel_acc_y/1024;
    offset_acc_z = vel_acc_z/1024;
    offsetTolerance=(offset_acc_z/9.81);

void calibrateAcc(float ox1,float ox2,float ox3){
    vel_acc_x= vel_acc_x + ox1;
    vel_acc_y= vel_acc_y + ox2;
    vel_acc_z= vel_acc_z + ox3;
}
    
```

Nachdem der Offset bestimmt wurde, kann dieser vor der Integration der Rohwerte herausgerechnet werden.

Danach zeigt sich das Signal des Gyroskops wie in Abbildung 6.5. Aufgetragen sind die drei Achsen des Gyroskops mit deren Drehraten. Zu erkennen ist, dass der Offset mit der Zeit Schwankungen unterliegt.

Die Drehraten des Gyroskops wurden integriert, um den Drift zwischen dem off-

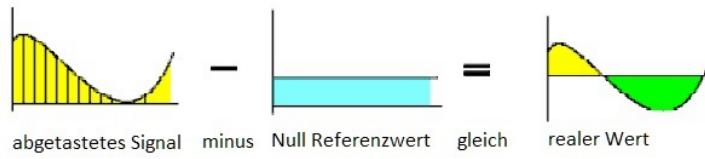


Abbildung 6.4: Eliminierung des Offsets [12]

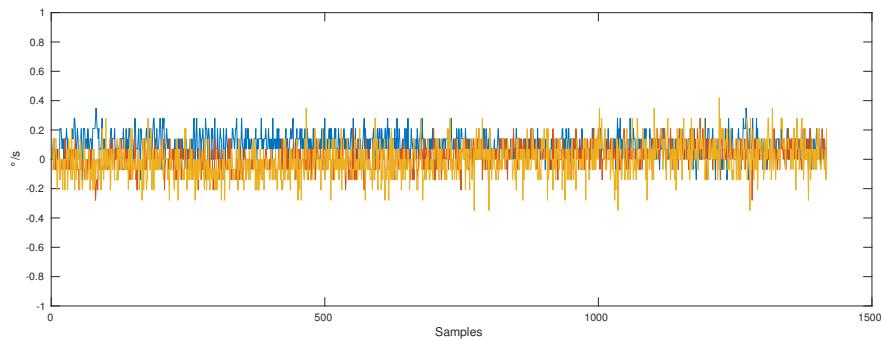


Abbildung 6.5: Gyroskop Werte nach Offset Eliminierung

setkorrigierten und unkorrigierten Signal zu vergleichen. Abbildung 6.6 zeigt eine deutliche Verbesserung des Drifts. Tabelle 6.1 zeigt die Winkelraten des Drifts.

	Drift	Drift mit Offsetkorrektur
X[°/s]	2,6509	0,028
Y[°/s]	9,686	0,18
Z[°/s]	2,0997	0,00833

Tabelle 6.3: Reduzierung des Drift durch Offsetkorrektur im Gyroskop

Für den Beschleunigungssensor wurde das Rauschen des Signals von zwei Messreihen in Tabelle 6.1 im stationären Zustand aufgenommen. Eine Fourier Transformation für das Signal zeigte, dass im Rahmen der Abtastfrequenz bis 50 Hz, die Amplituden gleichmäßig über dem Frequenzspektrum verteilt sind.

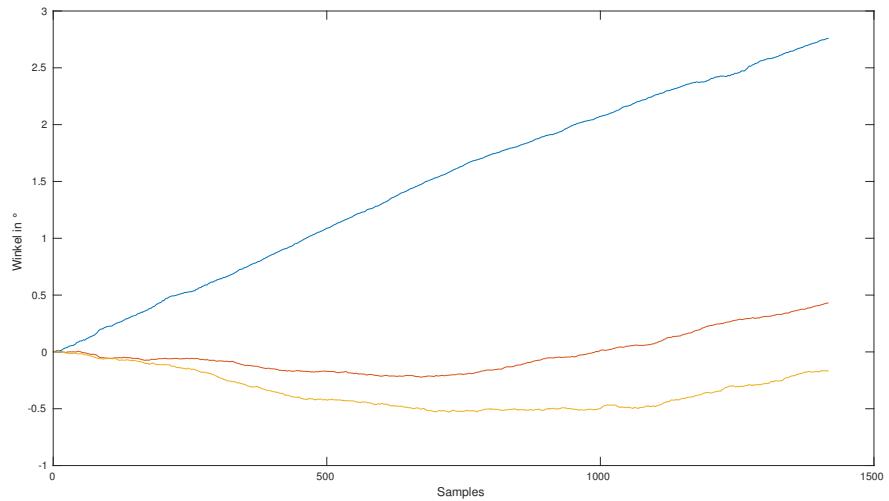


Abbildung 6.6: Drift des Gyroskops nach Eliminierung des Offsets

Achse	Durchschnitt	Maximalwert	Minimalwert	Standardabweichung	Bereich
X[m/s ²]	0,004622	0,1328	-0,1858	0,03331	0,3186
Y[m/s ²]	-0,021200	0,1415	-0,9379	0,0469	1,0790
Z[m/s ²]	-0,1547	0,0252	-0,4435	0,05533	0,4670

Tabelle 6.4: Statistik des Rauschens

6.2 Neigung

Um die Neigung der inertialen Messeinheit zu erhalten, wurde zunächst der Nick- und Roll Winkel mit Hilfe der im Beschleunigungssensor gemessenen Werte bestimmt. Zur akkuratesten Bestimmung des Gier Winkels wird ein Magnetometer benötigt.

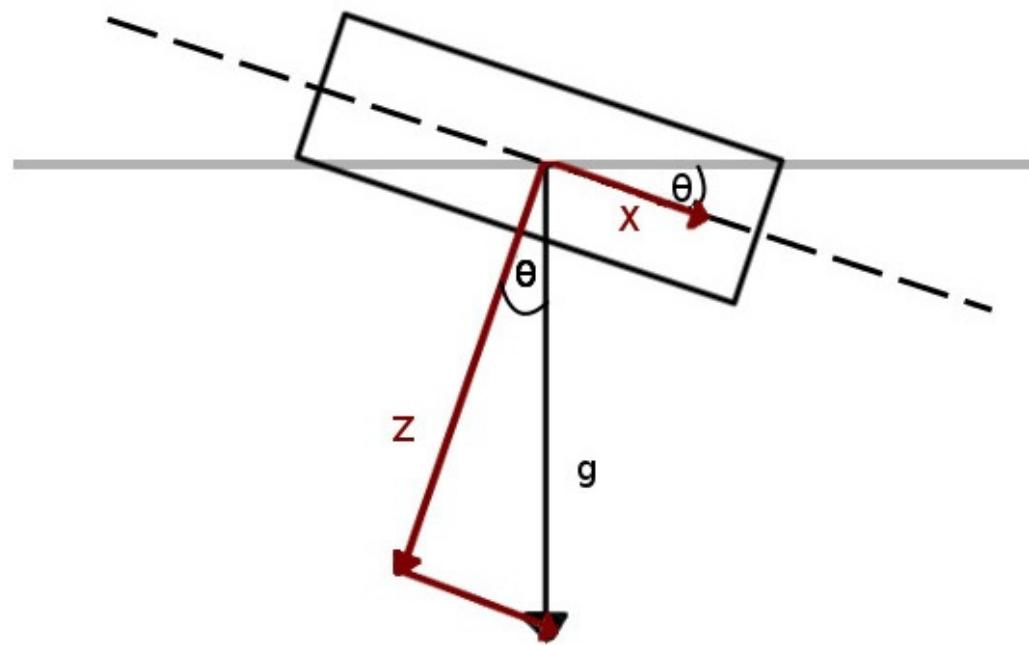


Abbildung 6.7: Inertiale Messeinheit bei Neigung

In Abbildung 6.7 ist ein Modell der Messeinheit bei Neigung zu sehen. Es veranschaulicht die Berechnung des Nick Winkels Θ . Die Erdbeschleunigung teilt sich in diesem Fall je nach Neigung auf die X und Z Achse auf. Der Winkel Θ kann mit der Tangens Funktion bestimmt werden. Viele Programmiersprachen haben zu diesem Zweck die Arcustangens2 Funktion implementiert. Ihr Vorteil besteht darin, für beide Parameter die Vorzeichen beachten zu können um so den Quadranten des Ergebnisses zu bestimmen. Das Ergebnis der Atan2 Funktion befindet sich zwischen PI und -PI und ist im Bogenmaß angegeben. Sie nimmt die Werte in folgender Form entgegen.

```
float atan2 ( float $y , float $x )
```

Um die Beschleunigungssensorwerte in Grad umzurechnen wurde zunächst PI auf das Ergebnis des Atan2 addiert um Werte zwischen 0 und 2 zu erhalten. Eine Multiplikation mit dem Umrechnungsfaktor $180/\text{PI}$ liefert den Θ Winkel AccYangle. Der Roll Winkel Φ berechnet sich analog als AccXangle.

```
AccXangle = (float)(atan2(accRaw[1], accRaw[2]) + M_PI)
*RAD_TO_DEG;
AccYangle = (float)(atan2(accRaw[2], accRaw[0]) + M_PI)
*RAD_TO_DEG;
```

Wie bereits erwähnt, unterliegt das Gyroskop einem Drift und muss daher durch den Beschleunigungssensor korrigiert werden. Der Beschleunigungssensor alleine kann keine akkurate Winkel liefern, da sein Signal sehr verrauscht ist und eine Bewegung der Messeinheit zu Störeinflüssen führt. Aus diesem Grund wurden beide Sensoren mit einem Komplementärfilter, wie in Abbildung 6.8 zu sehen, kombiniert.

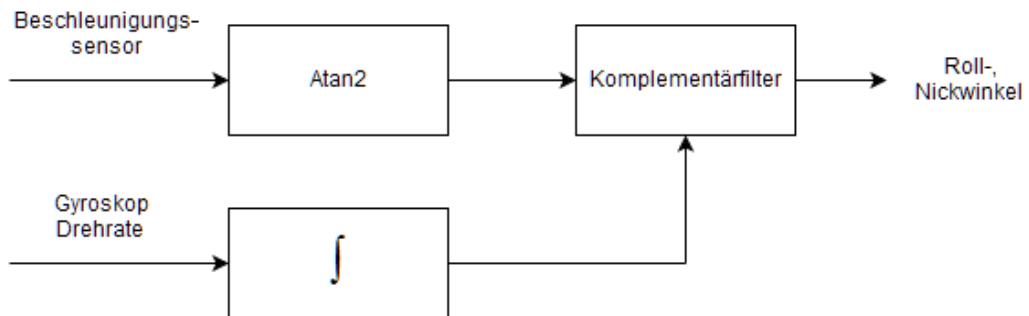


Abbildung 6.8: Bestimmung von Nick- und Roll Winkel

Für kurzzeitige Bewegungen liefert das Gyroskop genauere Ergebnisse und wird daher im Komplementärfilter höher gewichtet. Er stellt sich dar als eine Kombination aus Highpass und Lowpassfilter.

```
CFangleY=AA*(CFangleY+rate_gyr_y*DT)+(1-AA)*AccYangle;
// Highpass AA*(CFangleY+rate_gyr_y*DT)
// Lowpass (1-AA)*AccYangle
```

Die Filterkonstante AA definiert die Grenze zu der der Beschleunigungssensor oder das Gyroskop mehr Gewichtung erhält. Der Grenzwert kann mit der Länge des Loops dt berechnet werden.

$$\tau = \frac{AA \times dt}{1 - AA} = \frac{0.95 \times 0.02s}{0.05} = 0.38s \quad (6.4)$$

Zur Veranschaulichung wurden zwei Komplementärfilter parallel verwendet. In Abbildung 6.9 verwendet der CF1 eine Konstante von 0.95 und CF2 eine Konstante von 0.6. Zu erkennen ist, dass die Gewichtung bei CF2 stärker beim Beschleunigungssensor liegt und diesem auch bei kurzfristigen Bewegungen folgt.

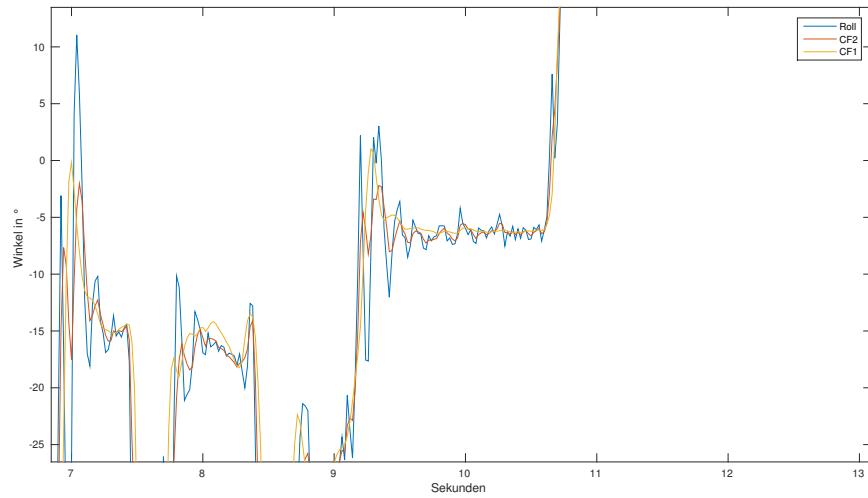


Abbildung 6.9: Vergleich der Filterkonstanten

Um eine weitere Verbesserung der Nick- und Roll Winkel zu erzielen kann ein Kalman Filter implementiert werden. Er bietet die Möglichkeit, einen zukünftigen Wert anhand der vorherigen Werte abzuschätzen indem diese mit ihrem Rauschen und anderen Ungenauigkeiten über einen Zeitraum beobachtet werden, da so Rückschlüsse über den Zustand des Systems geschlossen werden. Der Kalman Filter hat weithin genutzte Anwendungsgebiete im Bereich der Navigation und Kontrolle über Fahrzeuge und Flugzeuge. Die Vorteile des hier genutzten Komplementärfilters liegen im deutlich geringeren Rechenaufwand und der im Vergleich zum Kalman Filter einfacheren Theorie.

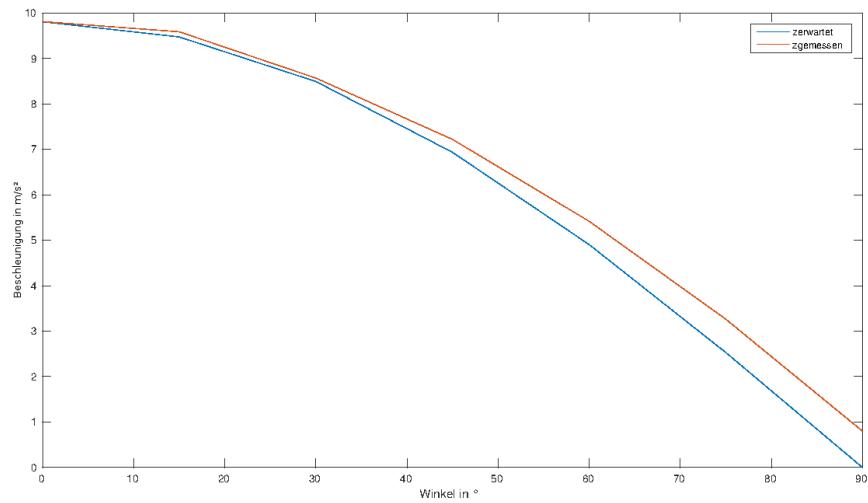


Abbildung 6.10: Genauigkeit der Winkel

Im stationären Zustand liegt die Erdgravitation voll auf der z-Achse des Sensors. Bei einer Kippung um den Nick Winkel teilt sich je nach Winkel die Komponente auf z- und x-Achse auf. In Abbildung 6.11 wurde die erwartete Beschleunigung in Abhängigkeit des Winkels aufgetragen und mit dem gemessenen Wert verglichen. Zu erkennen ist, dass die Ungenauigkeit mit Zunahme des Winkels ansteigt. Beträgt der Fehler bei 15° erst 1,6%, so sind es bei 75° bereits 7,4%.

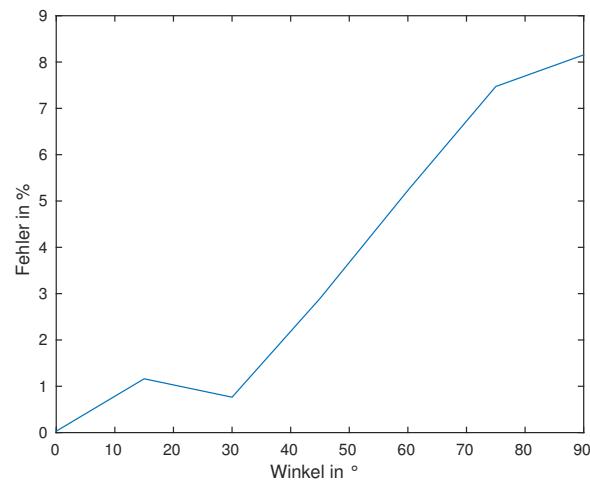


Abbildung 6.11: Winkelabhängiger Fehler

6.3 Drehung

Um das Verhalten des Gyroskops und des Magnetometers bei Drehung genauer zu untersuchen wurde die inertiale Messeinheit auf dem Teller eines Plattenspielers, wie in Abbildung 6.12 zu sehen, platziert.



Abbildung 6.12: Versuchsanordnung auf einem Plattenspieler

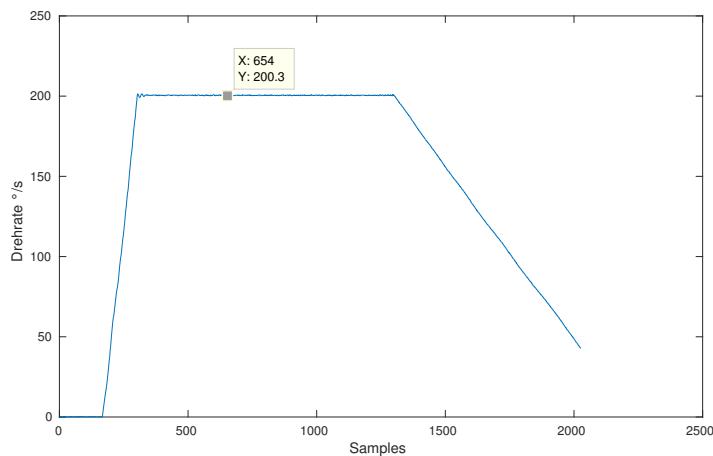


Abbildung 6.13: Gyroskop auf einem Plattenspieler

Das Verhalten des Gyroskops auf dem Plattenspieler zeigt sich in Abbildung 6.13. Es zeigt eine Drehrate von $200^{\circ}/s$. Der Plattenspieler dreht mit 33 Umdrehungen/-Minute. Dies sind $198^{\circ}/s$. Das Gyroskop ist also auf 2° genau bzw. hat einen Fehler

von 1% bei 33 U/min. Ein weiterer Test mit 45 U/min ergab eine Drehrate von 272 °/s. Erwartet wurden hier 270 °. Der Fehler bleibt gleich. Aufgrund der hohen Genauigkeit könnte ein Drehratensor auf einem Rad mit bekanntem Umfang einen weiteren Referenzwert bei der Positionsbestimmung liefern. Das Gyroskop könnte an einer beliebigen Stelle auf der Oberfläche eines Rades montiert werden um mit dem Umfang des Rades und der Drehrate die zurückgelegte Strecke zu berechnen.

Zur Orientierung in der z-Achse kann das Magnetometer verwendet werden. Um dieses zu nutzen, wurde dieses zunächst wie in [18] kalibriert. Hierfür wurden die Rohdaten der x und y-Achsen des Sensors nach mehreren Drehungen um die z-Achse in Abbildung 6.14 geplottet. Der Offset setzt sich zusammen aus den Störeinflüssen des harten und weichen Eisens (siehe: 3.3.1). Störungen durch hartes Eisen zeigt sich in einer Verschiebung des Kreises in Abbildung 6.14 vom Nullpunkt. Weiches Eisen verursacht ein Verkrümmen des Kreises zu einer Ellipse.

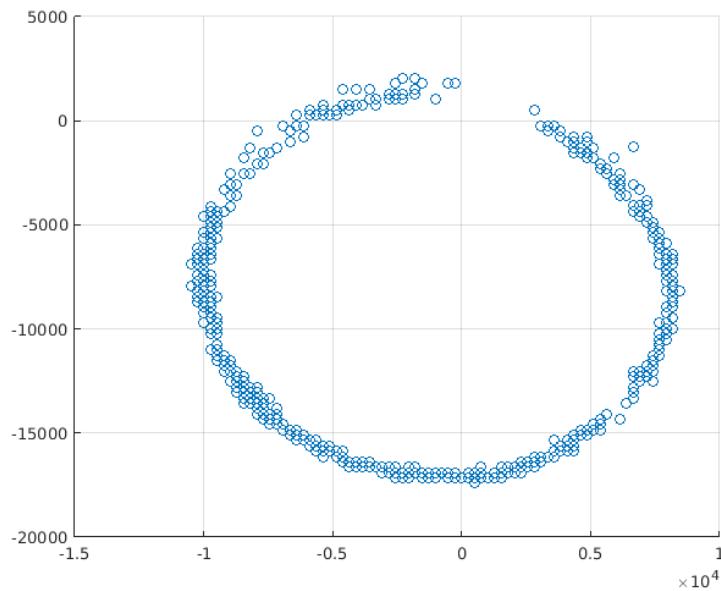


Abbildung 6.14: Offset des Magnetometers

Wie zu sehen ist, gibt es kaum Einfluss durch weiches Eisen, dafür aber eine Offset Verschiebung durch hartes Eisen. Für annehmbare Ergebnisse wurde dieser Offset korrigiert, indem die Maximal- und Minimalwerte der drei Magnetometer Achsen aufgenommen wurden um mit diesen die Abweichung vom Nullwert vom rohen Signal subtrahieren zu können. Nach der Kalibrierung zeigt sich das X-Y Diagramm der Rohwerte in Abbildung 6.15.

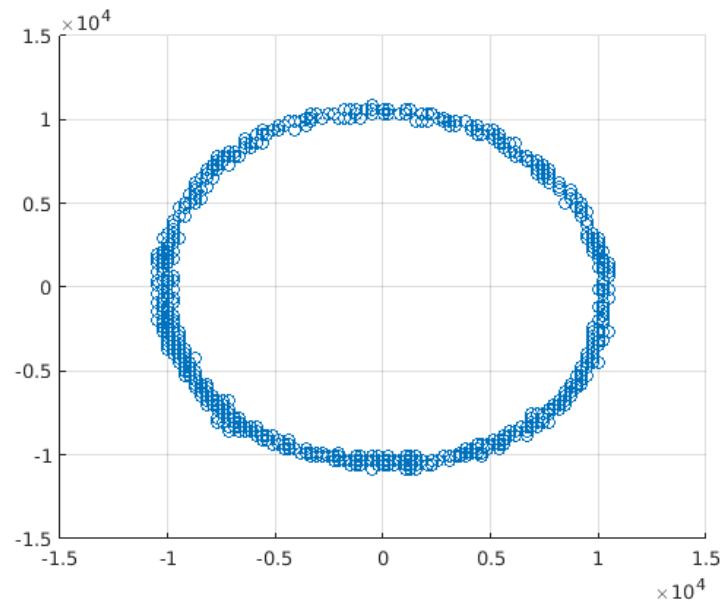


Abbildung 6.15: kalibriertes Magnetometer

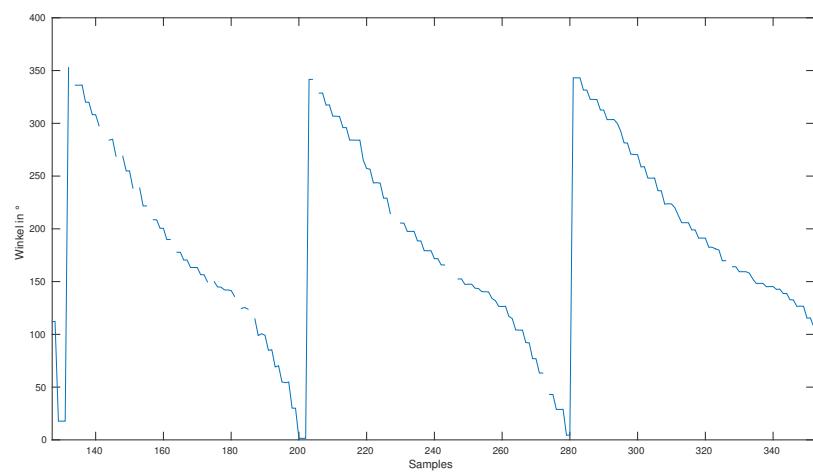


Abbildung 6.16: Magnetometer auf Plattenspieler

Um das Verhalten des Magnetometers bei Drehung zu analysieren, wurde der Sensor wie in Abbildung 6.16 einer Drehung auf dem Plattenspieler unterzogen. Erwartet wurde ein Sägezahnsignal mit einer gleichmäßigen zeitlichen Abnahme des Gier-Winkels. Wie zu erkennen, kommt es zu einem nicht gleichmäßigem Verhalten zwischen 100° und 250° . Verantwortlich hierfür können Bauteile des Plattenspielers sein, die eine Störung durch hartes oder weiches Eisen verursachen. Um das Signal des Magnetometers zu glätten, kann dieses ebenfalls über einen Komplementärfilter mit dem Gyroskop fusioniert werden, bei dem das Magnetometer Tiefpass- und das Gyroskop Hochpass gefiltert wird. Für eine korrekte Winkelberechnung müssen die Daten des Magnetometer einer Neigungskompensation unterzogen werden, die, ähnlich wie die Rotationsmatrix, über geometrische Beziehungen realisiert wird.

Vollständiges System zur Orientierung

Für eine vollständige und korrekte Orientierung wird das System in Abbildung 6.17 gezeigt.

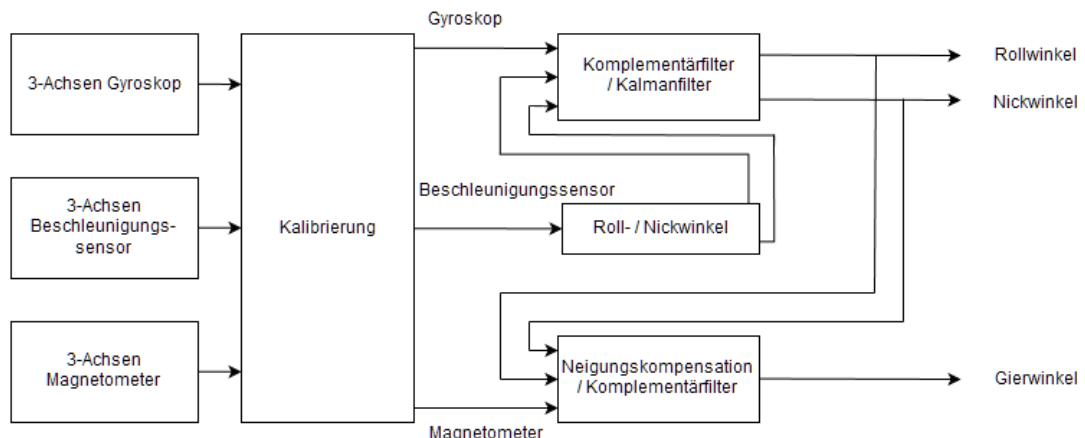


Abbildung 6.17: Vollständiges System zur Bestimmung der Orientierung

6.4 Positionsbestimmung

Um aus der Beschleunigung des Sensors die zurückgelegte Strecke zu berechnen muss das Signal zweifach integriert werden. Abbildung 6.18 zeigt die Funktionsweise des hier implementierten Algorithmus zur Positionsbestimmung.

$$\vec{v} = \int(\vec{a})dt \quad \vec{s} = \int(\vec{v})dt \quad (6.5)$$

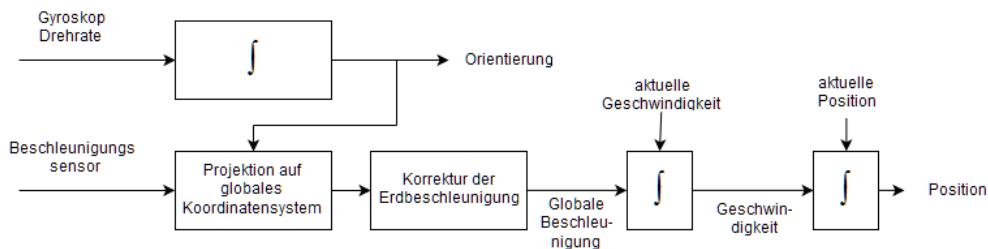


Abbildung 6.18: Algorithmus zur Positionsbestimmung

Tiefpass Filter

Zunächst wird das Signal mit einem FIR Tiefpass Filter von hochfrequentem Anteil befreit, der programmtechnisch wie folgt umgesetzt wurde:

```

// Low pass filter
/*
 * Typ: gleitender Mittelwert(FIR Simple moving average)
 * Berechnet den Durchschnittswert der Beschleunigung
 * von einer konstanten Anzahl an Werten
 */
do{
    readACC(accRaw);
    act_acc_x=(float)(act_acc_x+accRaw[0])-offset_acc_x;
    act_acc_y=(float)(act_acc_y+accRaw[1])-offset_acc_y;
    act_acc_z=(float)(act_acc_z+accRaw[2])-offset_acc_z;
    i++;
}while(i<8);
    act_acc_x=act_acc_x/8;
    act_acc_y=act_acc_y/8;
    act_acc_z=act_acc_z/8;
    i=0;

```

Abbildung 6.19 zeigt das ungefilterte Signal des Beschleunigungssensors neben dem des Tiefpass Filters, der innerhalb der Loop Zeit von 20 ms 25 Werte mittelt. Dies entspricht einer Abtastfrequenz von 1250 Hz. Zu erkennen ist wie bei langsam Bewegungen die Spitzen geglättet werden. Es zeigte sich jedoch, dass für die nachfolgenden komplexeren Berechnungen mit höchstens 800 Hertz abgetastet werden konnte, da es sonst nicht mehr möglich war die Loop Zeit von 20 ms einzuhalten.

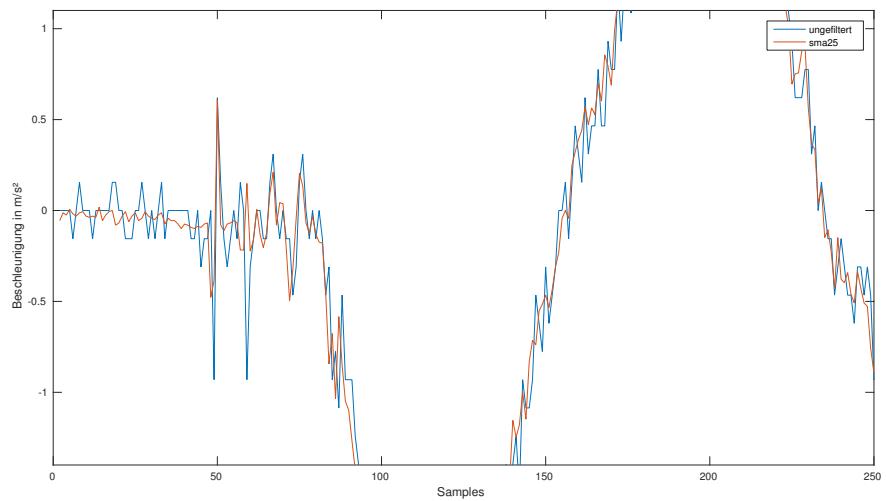


Abbildung 6.19: FIR Tiefpass mit einer Summierung von 25 Werten

Integration

Eine Integration kann als die Summierung der Flächen unter einer Kurve erklärt werden. Die Flächen werden als infinitesimal klein angesehen. Für eine reale Anwendung wird die Breite der Fläche Δx über die Durchlaufzeit des Programms bestimmt. Abbildung 6.20 zeigt die Abtastung eines Signals.

$$\int_a^b f(x) dx = \lim_{n \rightarrow \infty} \sum_{i=1}^n f(x_i) \Delta x \quad (6.6)$$

Formel 6.6 berechnet die Fläche unter der Kurve, in diesem Fall wird so zunächst die Geschwindigkeit und anschließend die Position (Formel: 6.5) berechnet. Δx beträgt in diesem Fall 20 ms. Um eine stimmige Integration zu gewährleisten muss die Sampling Zeit immer gleich sein.

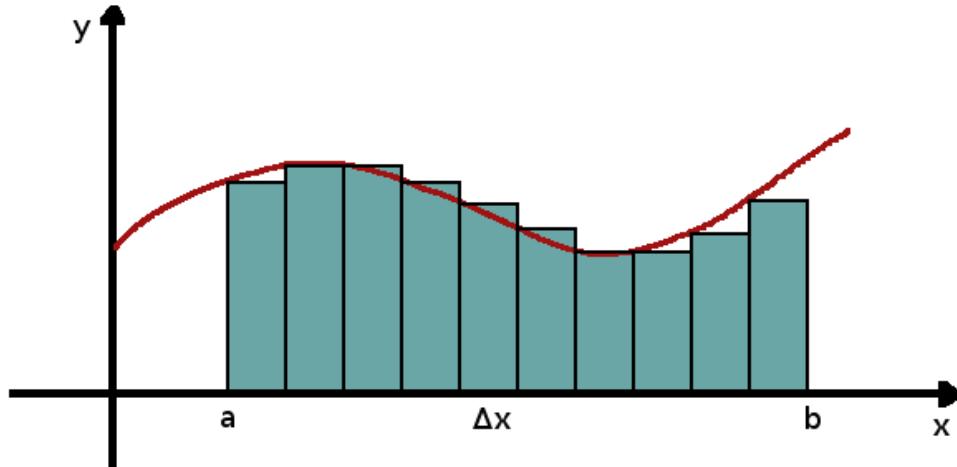


Abbildung 6.20: Abtastung eines Signals

Interpolation

Bei einer nicht infinitesimalen Abtastungszeit zeigt sich ein Fehler wie in Abbildung 6.20. Dieser Fehler häuft sich an und steigt durch das Doppelintegral für eine Positionsbestimmung somit quadratisch an.

Zur Reduzierung des Abtastungsfehlers kann das Signal interpoliert werden. Hierfür wird es wie in Abbildung 6.21 betrachtet. Fläche 1 zeigt die berechnete Fläche des Samples n. Fläche 2 wird zur Interpolation des Signals mit Hilfe der dreieckigen Fläche benötigt. Diese erschließt sich aus dem Wertunterschied zum vorhergegangen Sample n-1.

Die Fläche unter der Kurve berechnet sich damit in Formel: 6.7

$$\text{Fläche}_n = (\text{Sample}_n + \frac{|\text{Sample}_n - \text{Sample}_{n-1}|}{2}) \cdot \Delta x \quad (6.7)$$

Programmtechnisch wurde die Interpolation wie nachfolgend durchgeführt. Hierbei ist acc_buff_ der Wert der Beschleunigung im vorherigen Programmdurchlauf. OutputAcc ist der Wert des aktuellen Durchlaufs. Analog verhält sich die Berechnung der Position aus der Geschwindigkeit.

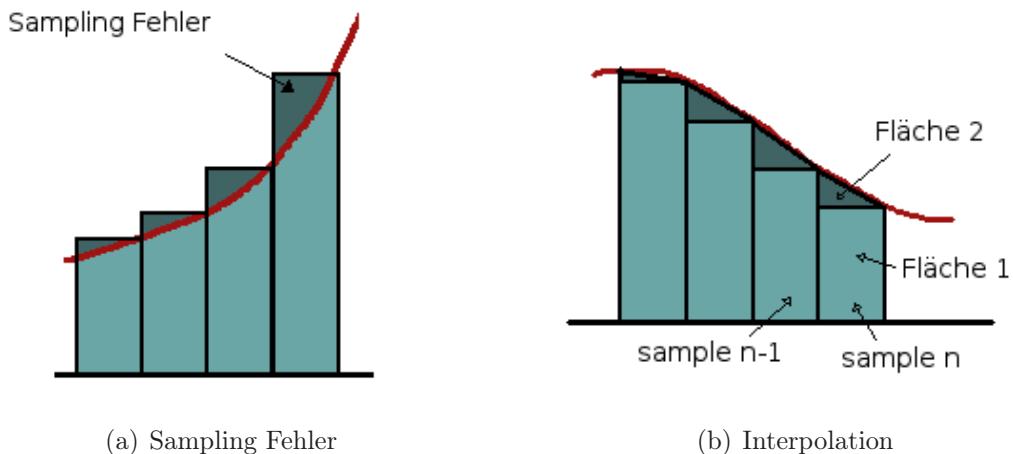


Abbildung 6.21: Sampling Fehler und Interpolation

```

//Geschwindigkeit mit Approximation 1. Ordnung
speed_acc_x+=(acc_buff_x+((OutputAcc_x-acc_buff_x)/2))*DT;
speed_acc_y+=(acc_buff_y+((OutputAcc_y-acc_buff_y)/2))*DT;
speed_acc_z+=(acc_buff_z+((OutputAcc_z-acc_buff_z)/2))*DT;
//Position mit Approximation 1. Ordnung
positionx+=(speed_buff_x+((speed_acc_x-speed_buff_x)/2))*DT;
positiony+=(speed_buff_y+((speed_acc_y-speed_buff_y)/2))*DT;
positionz+=(speed_buff_z+((speed_acc_z-speed_buff_z)/2))*DT;

```

Ansprechschwelle

Wenn keine Bewegung stattfindet, wird das Rauschen des Sensors als Beschleunigung interpretiert und verursacht damit ein Ansteigen der Geschwindigkeit. Um diesen Effekt zu minimieren wurde eine Ansprechschwelle eingesetzt (siehe Abbildung: 6.22). Werte unterhalb der Ansprechgrenze werden pauschal als Rauschen interpretiert und direkt zu 0 gesetzt. Die Grenze kann sich nach der in Abschnitt: 6.1 gemessenen Spanne des Rauschens richten bzw. wurde experimentell ermittelt und für verschiedene Aufgaben angepasst und kann sowohl für das Gyroskop und den Beschleunigungssensor implementiert werden.

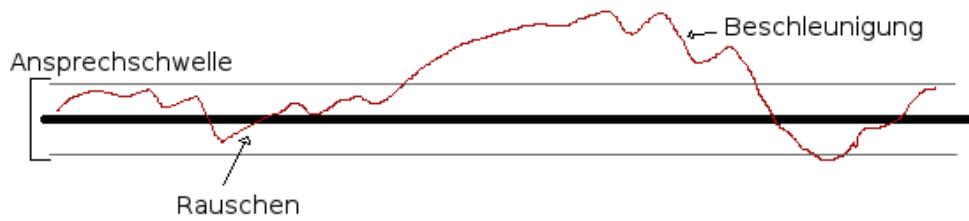


Abbildung 6.22: Ansprechschwelle

```
//Rauschunterdrueckung
if((OutputAcc_z <=0.25)&&(OutputAcc_z >= -0.25)){
    OutputAcc_z=0.0;
}
if((OutputAcc_x <=0.3)&&(OutputAcc_x >= -0.3)){
    OutputAcc_x=0.0;
}
if((OutputAcc_y <=0.25)&&(OutputAcc_y >= -0.25)){
    OutputAcc_y=0.0;
}
```

Bewegungsende

Bei Abschluss einer Bewegung ist die reale Beschleunigung gleich Null. Durch auftretende Fehler ist die Berechnung des Beschleunigungssensors nach Abschluss nicht Null. Es wurde daher eine Funktion implementiert, die die Anzahl von gemessenen Beschleunigungen unterhalb der Ansprechschwelle zählt um abzuschätzen wann eine Bewegung beendet ist. Hierfür kann je nach Anwendung die Anzahl der Werte (ME) angepasst werden, die es braucht, bis die Geschwindigkeit zu null gesetzt wird. Der nachfolgende Code zeigt die Erkennung eines Bewegungsendes für die x-Achse. Analog gilt die Erkennung für die anderen Achsen.

```
if(OutputAcc_x==0.0){
    j++;
}
else{
    j=0;
}
if(j>=ME){
    speed_acc_x=0;
}
```

NXT Roboter

Um eine Positionsbestimmung für ein Fahrzeug zu simulieren, wurde der Sensor auf einem Lego-NXT Roboter wie in Abbildung 6.23 montiert.

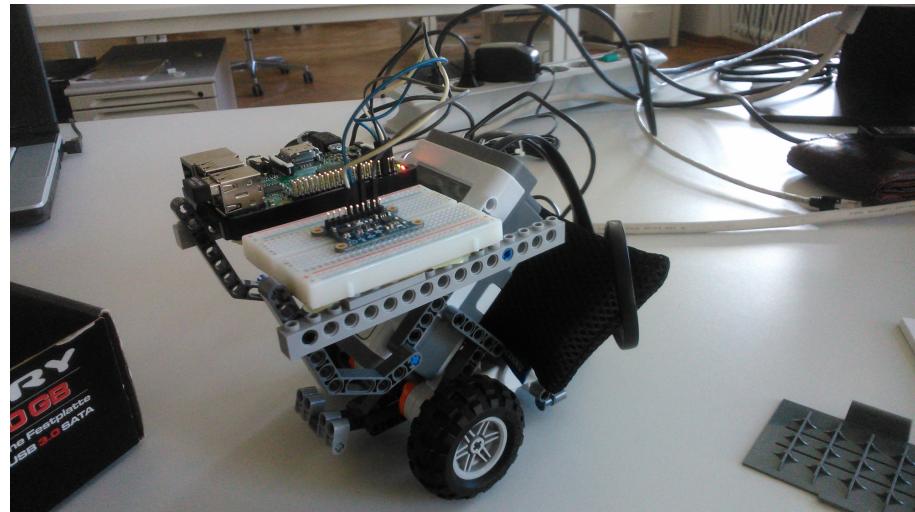


Abbildung 6.23: IMU und Raspberry Pi auf Lego NXT

Die erste durchgeführte Messreihe wurde mit den in Abschnitt: 6.4 vorgestellten Methoden durchgeführt. Aufgezeichnet wurde die errechnete zurückgelegte Strecke, um diese mit der wirklich gefahrenen Strecke zu vergleichen.

Tabelle 6.5: Messung der Strecke bei Fahrt von 1,6 m
zurückgelegte Strecke[m]: 1,60
gemessene Strecke[m]: 5,91 2,171 1,8854 1,2064

Tabelle 6.6: Messung der Strecke bei Fahrt von 2,80 m
zurückgelegte Strecke[m]: 2,80
gemessene Strecke[m]: 40,26 12,17 14,584 20,64

Für eine Strecke von 1,6 m in Tabelle 6.5 liefert der Sensor in drei von vier Fällen ein Ergebnis unter 50 cm Abweichung. Fährt der Roboter wie in Tabelle 6.6 eine Strecke von 2,8m so sind die errechneten Ergebnisse bereits vollkommen unbrauchbar. Aufgrund der Vibrationen des Roboters sowie eventueller Abweichungen bei Montage des Sensors von 0 ° in der horizontalen Achse wurde versucht, eine Rotationsmatrix

wie in Kapitel 4 zur Kompensation dieser Störeinflüsse zu implementieren. Abbildung 6.24 zeigt die gemessenen Beschleunigungen bei der Roboterfahrt. Gefahren wurde in -x Richtung. Auffällig sind die starken Beschleunigungen die auf die vermeintlich nicht beanspruchten Achsen wirken.

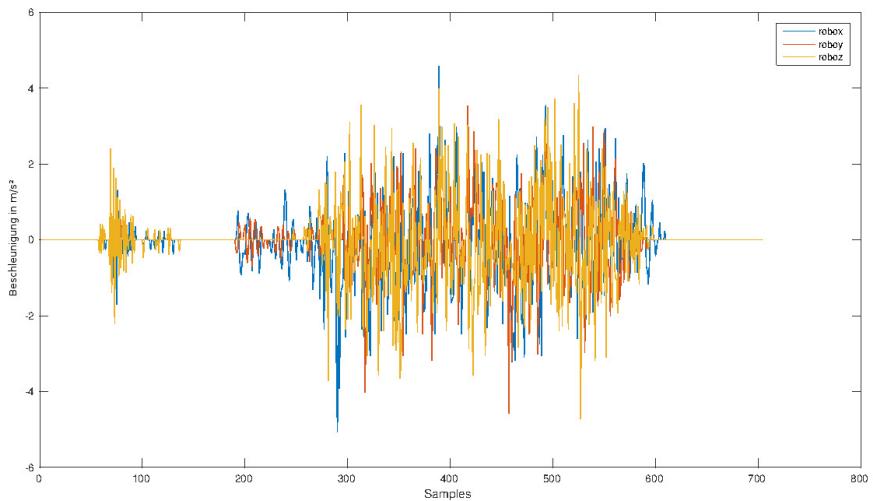


Abbildung 6.24: Beschleunigungen der Achsen bei Roboterfahrt

Der nachfolgende Code zeigt dabei die Methode mittels Drehung der Roll, Nick und Gierachse in das erdfeste System. Hierzu wird zunächst auf Grundlage der Rotationsmatrix um die x-Achse und danach um die y- und z-Achse gedreht.

```
void rotieren(float RangleX, float RangleY, float RangleZ,
float outputX, float outputY, float outputZ){
    //rotieren um X
    RangleX=RangleX*M_PI/180;
    float temp = outputY;
    outputY= outputY *cos(RangleX)-outputZ*sin(RangleX);
    outputZ= temp*sin(RangleX)+ outputZ*cos(RangleX);
    //rotieren um y
    RangleY=RangleY*M_PI/180;
    temp=outputZ;
    outputZ=outputZ*cos(RangleY)-outputX*sin(RangleY);
    outputX= temp*sin(RangleY)+outputX*cos(RangleY);
    //rotieren um Z
    RangleZ=RangleZ*M_PI/180;
    temp =outputX;
```

```

        outputX= outputX*cos(RangleZ)-outputY*sin(RangleZ);
        outputY= temp*sin(RangleZ)+ outputY*cos(RangleZ);

        OutputAcc_x=outputX;
        OutputAcc_y=outputY;
        OutputAcc_z=outputZ;

    }

```

Beim Aufruf der Funktion sind die übergebenen Parameter die Winkel des Komplementärfilters und die gemessenen Beschleunigungen im körperfesten System. Wenn die Beschleunigungen im erdfesten System sind kann die Erdgravitation herausgerechnet werden.

```

rotieren(CFangleX,CFangleY,0,OutputAcc_x,
          OutputAcc_y,OutputAcc_z);
//Herausrechnen der Erdbeschleunigung:
OutputAcc_z=OutputAcc_z-9.81;

```

Mit der Streckenberechnung über die Rotationsmatrix wurden weitere Fahrttests durchgeführt wie in den Tabellen 6.4 und 6.4 zu sehen.

zurückgelegte Strecke[m]:	2,8
gemessene Strecke[m]:	3,45 5,527 7,09 4,67 4,8

Tabelle 6.7: Messung der Strecke bei Fahrt von 2,8 m mit Rotationsmatrix

zurückgelegte Strecke[m]:	7
gemessene Strecke[m]:	55,42 19,141 39,86

Tabelle 6.8: Messung der Strecke bei Fahrt von 7 m mit Rotationsmatrix

Die Implementierung der Rotationsmatrix bietet ein genaueres Ergebnis bei einer Fahrt von 2,8 m. Trotzdem sind die Störeinflüsse für doppelte Integration zu groß als dass hier eine geeignete Berechnung der Strecke erzielt würde. Ein Grund für die Fehlkalkulation stellt hier der Komplementärfilter, der die Winkel für die Rotationsmatrix bereit stellt. Anfahren und Abbremsen sorgt durch die Beschleunigung für falsche Winkel, woraufhin die Komponenten der Beschleunigung nicht korrekt in das globale System umgerechnet werden. Besonders auffällig ist dieses Verhalten bei sehr kurzer Strecke und Schieflage des Sensors. Abbildung 6.25 zeigt dieses Verhalten bei einer Strecke von ca. 50 cm. Der erste Pfeil von links beschreibt den Punkt

nachdem das Gerät aufgenommen wurde und sich die Beschleunigung in Schieflage zu 0 rechnet. Danach erfolgt die Beschleunigung bis an den zweiten Punkt. Ab hier wird deutlich mehr negative Beschleunigung erzeugt als das Abstoppen des Sensors in Wirklichkeit mit sich bringt. Die Folge ist eine falsche Kalkulation der Strecke, die am dritten Punkt abgeschlossen ist.

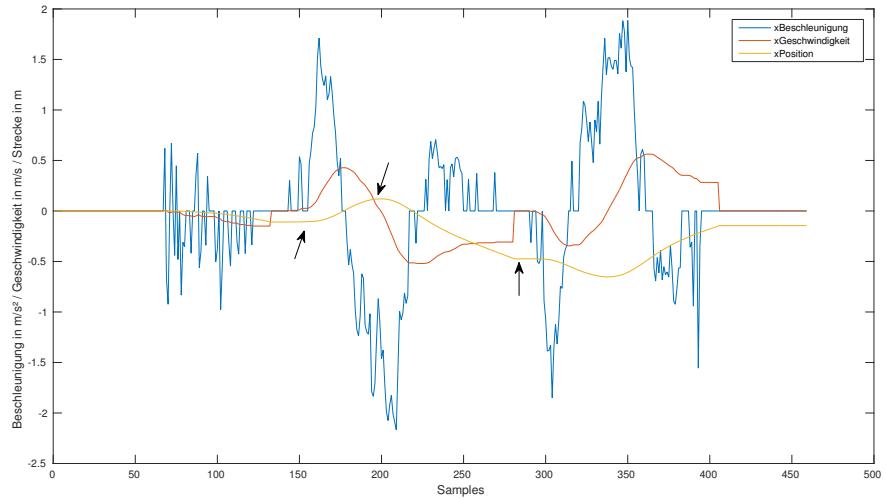


Abbildung 6.25: Kurze Strecke berechnet mit Rotationsmatrix

Um dieses Verhalten zu verhindern wurden verschiedene Koeffizienten für den Komplementärfilter verwendet. Ein Beispiel ist in Abbildung 6.26 zu sehen. Hier liegt die Gewichtung des Filters stark beim Beschleunigungssensor.

Die inertiale Messeinheit wurde im 45 °Winkel platziert und anschließend beschleunigt um die Änderung der Winkel abzuschätzen. Keine der getesteten Filterkonstanten und keine dynamische Anpassung konnte ein zufriedenstellendes Ergebnis liefern. Bei zu hoher Gewichtung des Gyroskops ist oft eine Diskrepanz zwischen den errechneten Winkeln der beiden Sensoren zu finden. So kommt es bei ausschließlicher Nutzung des Gyroskops zu keiner Winkeländerung bei Beschleunigung. Der schnell auftretende Drift sorgt allerdings dafür, dass eine korrekte Umrechnung ins erdfeste System nicht zufriedenstellend funktioniert.

Um das Verhalten des Sensors in großem Maßstab zu testen, wurden einige Zugfahrten zwischen S-Bahnhöfen aufgezeichnet. Der Algorithmus wurde hierfür nicht angepasst. Gemessen wurden hier die Geschwindigkeiten und Beschleunigungen, wie in Abbildung 6.27 zu sehen. Für die Fahrten zwischen zwei Bahnhöfen wurden nach der Beschleunigungsphase Werte zwischen 80 und 100 km/h erkannt, die somit ein relativ genaues Ergebnis stellen. Fehler treten vor allem bei Abbremsung auf, wie

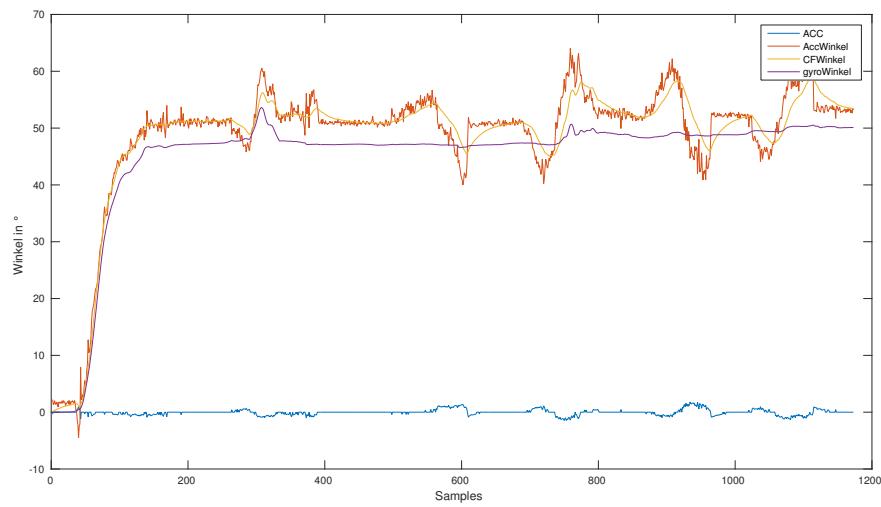


Abbildung 6.26: Winkeländerung bei Beschleunigung

auch in Abbildung 6.27 zu sehen. Hier fällt die Geschwindigkeit auf -40 m/s. Erwartet nach Bremsung wären in diesem Fall -22 m/s.

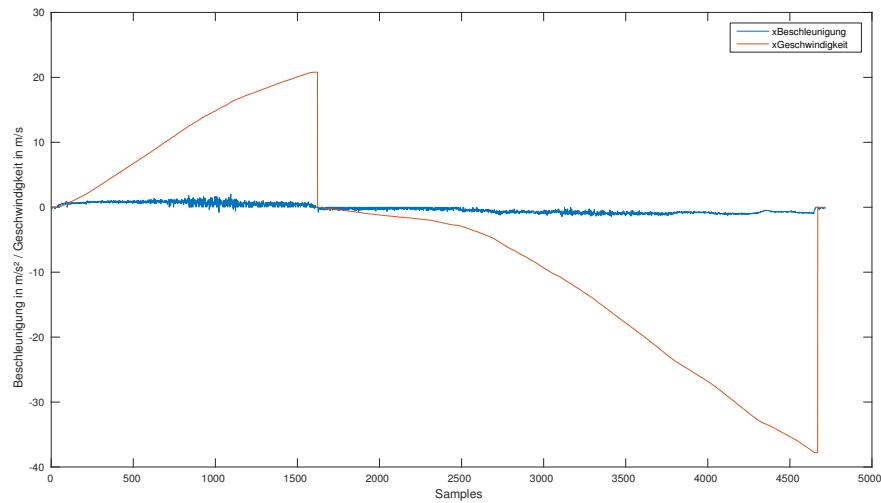


Abbildung 6.27: Zugfahrt

Schiebetest

Um die Genauigkeit der Positionsbestimmung im Bereich bis 1 Meter festzustellen, wurde der Sensor auf einem Tisch geschoben. Verglichen wurde hier der in Abschnitt 6.4 beschriebene Tiefpassfilter mit endlicher Impulsantwort mit einem Tiefpass Filter mit unendlicher Impulsantwort, der die Mittelwerte exponentiell glättet. Der Vorteil des Filters mit exponentiell gleitendem Mittelwert (Formel: 6.8) liegt in seiner kurzen Verzögerungszeit und dem geringen Rechenaufwand. Für diese Anwendung zeigte er das beste Ergebnis bei einer Filterkonstanten von $\alpha = 0.98$, wie in Abbildung 6.28 zu sehen, bei der der Sensor genau 1 m bzw. 0,5 m geschoben wurde.

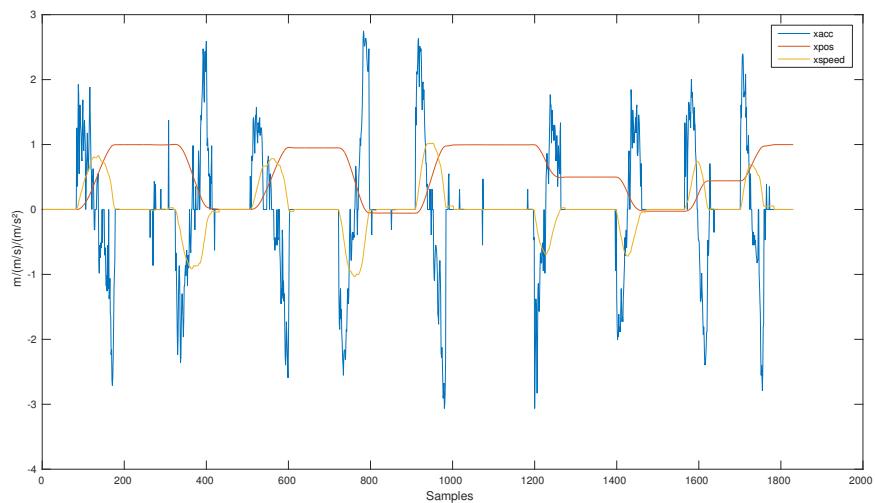


Abbildung 6.28: Schiebung des Sensors mit IIR Exponential Moving Average Filter(1 Pol Tiefpass)

$$y_n = y_{n-1} \cdot (1 - \alpha) + \alpha \cdot x_n \quad (6.8)$$

Um zu analysieren, wie der Filter die Ausgabefrequenz beeinflusst, wurde die Übertragungsfunktion (Formel: 6.10) im z-Bereich betrachtet und mit der Filterkonstanten von $\alpha = 0,98$ in ein Bode Diagram (Abbildung: 6.29) überführt. Wie zu erkennen, genügt bei einer Abtastzeit von 20ms bereits eine Amplitudenreduzierung der Beschleunigungssensordaten um -0,35 dB ab ca. 15 Hz (100 rad/s) um diese für eine derartige Anwendung bestmöglich zu glätten.

$$\beta = 1 - \alpha \quad (6.9)$$

$$H[z] = \frac{\alpha}{1 - \beta \cdot z^{-1}} \quad (6.10)$$

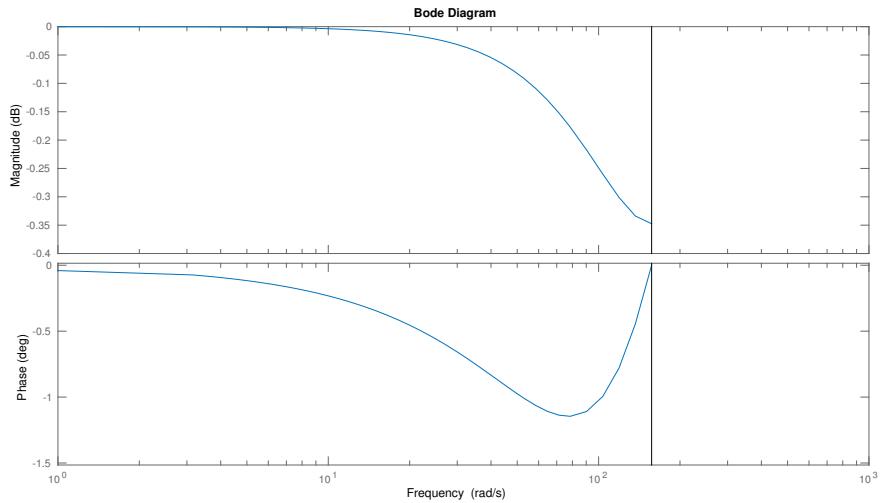


Abbildung 6.29: Bode Diagramm des EMA Tiefpass bei $\alpha = 0,98$

In Tabelle 6.4 wurde der Simple-Moving-Average mit verschiedenen Anzahlen zu mittelnder Werte dem Exponential-Moving-Average Filter mit verschiedenen Konstanten gegenübergestellt und die Abweichung vom erwarteten Wert gemessen. Der Sensor wurde jeweils 7 bis 8 mal um genau einen Meter bewegt, wobei jeweils 5 bis 6 Messreihen aufgenommen wurden. Die Tabelle zeigt den durchschnittlichen Fehler je Bewegung. Der Exponential-Moving-Average arbeitet mit 50 Hz, da einmal pro Loop abgetastet wird. Der Simple-Moving-Average versucht mit einer gewissen Anzahl je Loop abzutasten um den Durchschnittswert zu bilden. Bei einer Abtastfrequenz des Sensors von 200 Hz wie in Tabelle 6.4 könnten also bis zu 4 verschiedene Werte bei einem Loop mit 50 Hz entgegengenommen werden. Um die Anzahl der Werte für den Simple-Moving-Average zu vergrößern, wurde der Sensor mit seiner maximalen Abtastfrequenz von 5,367 KHz betrieben. Im Idealfall könnten so maximal 107 verschiedene Werte zur Berechnung des Filters pro Loop verwendet werden. Es zeigte sich jedoch, dass 16 Abtastungen im Loop, welches einer Abtastung von ca. 800 Hz entspricht, bereits zuviel CPU Ressourcen benötigen. Der Programmdurchlauf in 20 ms ist hier nichtmehr gewährleistet. Für den SMA Filter wurden daher Konstanten von 2,4 und 8 getestet. Geringe Filterung für den SMA mit 2 Summierungen pro Loop zeigte sich am effektivsten.

Ideale Voraussetzungen für eine Anwendung zur Positionsbestimmung unter einem Meter zeigte der Exponential-Moving-Average mit $\alpha = 0,98$ bei einem durchschnittlichen Fehler von 7,1%.

Filter:	SMA 2	SMA 4	SMA 8	
Fehler[%]:	15,68	22,26	16,67	
Standardabweichung[cm]:	13,1	27,75	21,95	
Filter:	EMA $\alpha=0,2$	EMA $\alpha=0,9$	EMA $\alpha=0,98$	ohne
Fehler[%]:	55,14	19,25	7,1	13,76
Standardabweichung[cm]:	21,95	14,4	5,53	12,98

Tabelle 6.9: Vergleich der Filter für Schiebetest

Schritterkennung

Die inertiale Messeinheit bietet die Möglichkeit einen Schritterkennungsalgorithmus (siehe Abbildung: 6.31) zu entwickeln um die Position anhand von zurückgelegten Schritten abzuschätzen.

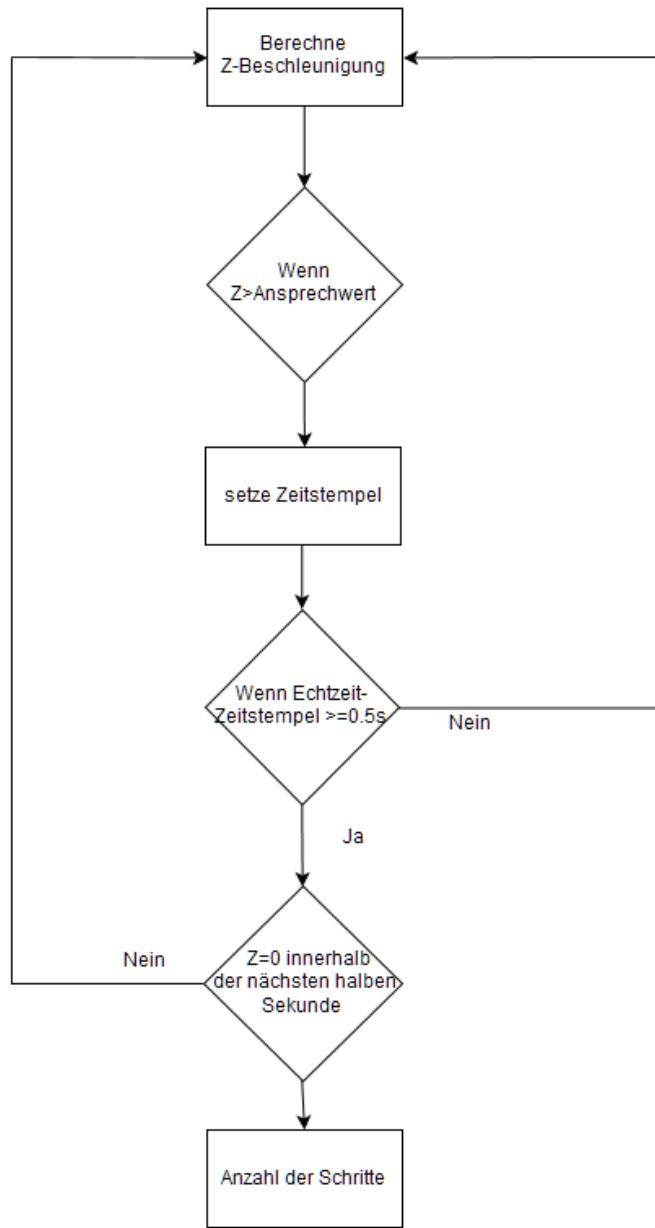


Abbildung 6.30: Algorithmus zur Schritterkennung

Zunächst wurde die Beschleunigung der z-Achse bei einer Schrittbewegung betrachtet. Hierfür wurde der Sensor in der Hand gehalten. Eine bessere Erkennung der Schritte ist bei einer Montage des Sensors auf einem Schuh möglich. Ein erster Algorithmus versuchte die Peaks der z-Beschleunigung zu erkennen. Dieser erkannte jedoch zu viele Schritte. Daraufhin wurde die Prämisse gebildet, dass zwei erkannte Peaks mindestens 0,5 Sekunden Abstand zueinander haben müssen. Auch dieser Algorithmus berechnete noch zuviele Schritte. Wie in Abbildung 6.31 zu sehen, fällt die Beschleunigung nach einem Peak auf 0. Dieses ist die zweite Bedingung um einen Schritt zu detektieren. Dieser Algorithmus verwendet die in 6.4 beschriebene Abwandlung der Rotationsmatrix zur Berechnung der z-Beschleunigung, um ein korrektes Verhalten bei Schieflage zu gewährleisten.

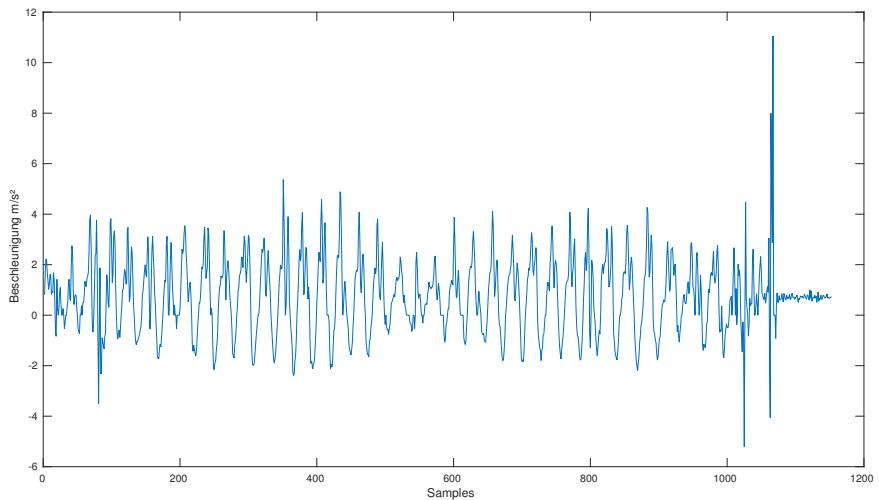


Abbildung 6.31: Beschleunigung der z-Achse bei einer Schrittbewegung

```

if(OutputAcc_z >= 2.1){
    //Bei Erkennen von z>=2.1 wird Zeit festgehalten
    steptimedifference=runtime;
    if(runtime-steptimedifference >= 0.5){
        sprung=1;
    }
}
if(sprung==1){
    //Wenn Beschleunigung Null innerhalb der
    //naechsten halben Sekunde
    //wird ein Schritt erkannt.
}
    
```

```

float secsteptimedifference=runtime;
if(OutputAcc_z<=0.5&&OutputAcc_z>=-0.5){
    if (runtime - secsteptimedifference <=0.5){
        realstep++;
        sprung=0;
    }
}
stepdistance=realstep*0.7;
}

```

Die Variable *stepdistance* liefert eine Abschätzung der zurückgelegten Strecke. Sie nimmt eine Schrittweite von 70 cm an. Eine Verbesserung des Algorithmus kann erzielt werden, indem diese Distanz adaptiv über beispielsweise eine mit GPS gemessene Distanz ermittelt wird. Auch kann anhand der Frequenz der Peaks und der Orientierung des Gerätes eine Klassifizierung der Bewegung vorgenommen werden, die beeinhalten kann ob eine Person geht, läuft, liegt oder Treppen hinauf oder herunter geht. Tabelle 6.4 zeigt den Vergleich von verschiedenen Ansprechschwellen, die für den Schritterkennungsalgorithmus getestet wurden. Mehrere Messreihen mit jeweils 10 und 20 Schritten wurden durchgeführt. Das genaueste Ergebnis liefert für diese Anwendung ein Schwellwert von $2,1 m/s^2$.

Ansprechschwelle [m/s^2]	1,7	1,8	2,1	2,3
Fehler [%]:	23	20	11,6	31,4

Tabelle 6.10: Vergleich von Ansprechschwellen für eine Schritterkennung

Weiterhin wurden die zwei Wegstrecken von 7 und 14 m gelaufen, um die Berechnung über das Doppelintegral mit der Abschätzung anhand der zurückgelegten Schritte zu vergleichen. Jeweils 5 Messungen ergeben die Durchschnittswerte in Tabelle 6.4. Die Berechnung über das Doppelintegral bleibt ungenau. Vor allem für kürzere Strecken aus den in Abschnitt 6.4 beschriebenen Problemen. Ab einem gewissen Punkt setzt dann die starke Ungenauigkeit durch das Doppelintegral ein. Der Schritterkennungsalgorithmus bietet für Strecken in Innenräumen ein annehmbares Ergebnis, das bei etwaiger Kopplung mit Referenzpunkten eine Indoor Positionsbestimmung realisieren kann.

Strecke in m	Erkannte Schritte	Berechnung Schritte[m]	Berechnung Doppelintegral[m]
7	7,2	5,04	2,448
14	14,5	10,5	10,184

Tabelle 6.11: Vergleich Streckenberechnung Schritterkennungsalgorithmus und Doppelintegral

7 Zusammenfassung

In dieser Arbeit wurde eine inertiale Messeinheit mit einem Raspberry Pi gekoppelt. Grundlage hierfür bot die I2C Schnittstelle. Die Möglichkeiten eine Positionsbestimmung zu realisieren wurden erörtert. Hierfür können u.A. Koppelnavigation und auf elektromagnetischen Wellen basierende Methoden wie etwa Triangulation eingesetzt werden.

Die Funktionsweisen der in der inertialen Messeinheit integrierten Sensoren wurden zusammen mit den auftretenden Fehlerquellen dargestellt. Hier wurde der Aufbau von unterschiedlichen Sensoren dargestellt um anwendungsspezifisch einen Sensorotyp bestimmen zu können.

Es wurde ein Programm entwickelt, dass die Daten von Beschleunigungssensor, Gyroskop und Magnetometer liest. Es minimiert die für die Sensoren charakteristischen Fehler durch verschiedene Kalibrierungsfunktionen. Im stationären Zustand zeigte sich eine deutliche Reduzierung des Offsets bei allen Sensoren. Dieser unterliegt beim Beschleunigungssensor und beim Gyroskop zeitlichen Schwankungen, wie aus den Messungen ersichtlich wurde.

Das Rauschverhalten des Beschleunigungssensors und des Gyroskops wurde untersucht. Unterschiedliche Achsen unterlagen dabei verschiedenen Stärken. Messungen zeigten Maximalwerte des Rauschbereichs für die z-Achse bis 1 m/s^2 beim Beschleunigungssensor.

Das geschriebene Programm ist in der Lage, die Orientierung des Gerätes über Roll-, Nick- und Gierwinkel herzustellen. Ein Komplementärfilter diente zur Kombinierung von Gyroskop und Beschleunigungssensor, um eine Reduzierung der sensorspezifischen Fehler für die Berechnung von Roll- und Nickwinkel zu erreichen. Hier wurde die Genauigkeit der errechneten Winkel diskutiert. Es zeigte sich eine mit zunehmendem Winkel zunehmende Ungenauigkeit bei Berechnung der Orientierungswinkel.

Magnetometer und Gyroskop wurden auf einem drehenden Schallplattenspieler untersucht. Hierfür wurden zunächst die Störungen durch Einflüsse von hartem Eisen auf das Magnetometer reduziert. Es zeigte sich eine Abweichung des berechneten Gier Winkels in einem bestimmten Winkelbereich. In diesem Zuge zeigte die Drehrate des Gyroskops eine hohe Genauigkeit. Sie kann verwendet werden um eine Strecke anhand eines Radumfangs zu messen, wenn ein Gyroskop auf diesem montiert wird. Ein vollständiges System zur Orientierungsbeschreibung wurde aufgezeichnet.

Mit Hilfe einer Rotationsmatrix wurden die Daten des Beschleunigungssensors in ein erdfestes Koordinatensystem überführt. In diesem Zuge wurde eine Abwandlung der Rotationsmatrix verwendet. Diverse Experimente zur Bestimmung von zurückgelegten Strecken wurden durchgeführt. Hier zeigte sich die Notwendigkeit der Rotationsmatrix aufgrund der Fehlkalkulationen beim Gehen, verursacht durch Schieflage des Sensors. Durch Implementierung der Rotationsmatrix wurde die Positionsrechnung besonders für kurze Strecken verfälscht, da es bei Anlegen der Beschleunigung zur Fehlkalkulation der Winkel und damit zu Fehlern in der Berechnung der Rotationsmatrix kam. Es wurde versucht, diese durch dynamische Anpassung der Filterkonstanten zu minimieren. Hier zeigte sich die Grenze des Komplementärfilters für derartige Anwendungen. Weiterhin wurde gezeigt, wo die Grenzen der Positionsbestimmung durch einen Beschleunigungssensor liegen und welche Methoden zur weiteren Verbesserung genutzt werden könnten. Hier wurde eine Fahrt mit einem Fahrzeug simuliert. Starke Vibrationen verursachten auf allen Achsen zu große Störreinflüsse für eine hinreichende Berechnung der Strecke durch Akkumulation der Fehler bei zweifacher Integration der gemessenen Beschleunigung.

Filter wurden getestet um die Daten zur Positionsbestimmung zu verbessern. Ein Tiefpass Filter der arithmetische Mittelwerte bildet wurde vorgestellt. Es wurde getestet, bis zu welcher Frequenz dieser auf einem Raspberry Pi betrieben werden kann. Ab 800 Hz Abtastrate ergab sich eine zu große Auslastung des Raspberry Pi um die vorgegebene Durchlaufzeit des Programms einzuhalten.

Weitere Methoden zur Optimierung des Signals, z.B. eine Interpolation der Daten, eine Erkennung von Bewegungsabschlüssen und eine Ansprechschwelle wurden implementiert. Bei der Interpolation wurde durch Bildung von Dreiecken die Integration optimiert. Für eine Positionsbestimmung unter einem Meter wurde der Sensor auf einem Tisch geschoben und die Ergebnisse nach Filterung mit einem Tiefpass, der arithmetische Mittelwerte bildet, mit einem Tiefpass 1. Ordnung mit exponentieller Charakteristik verglichen. Das beste Ergebnis lieferte hier der Tiefpass 1. Ordnung. Ein Bode-Diagramm veranschaulichte die gering gewählte Filterung. Zudem wurde ein Algorithmus zur Schritterkennung entwickelt. Es zeigte sich, obwohl der Sensor nicht an einen Schuh montiert wurde, eine relativ hohe Genauigkeit beim Erkennen der Schritte. Über die Schrittanzahl wurde die zurückgelegte Strecke berechnet. Hier zeigte sich eine deutlich höhere Genauigkeit als durch die Streckenberechnung mittels Doppelintegral. Abschließend wurde aufgezeigt, wie die Schritterkennung optimiert werden und wie eine Klassifizierung von Bewegungszuständen erreicht werden kann.

Ausblick

Zur Orientierungsbestimmung lieferte die inertiale Messeinheit bereits mit einfachen Methoden annehmbare Ergebnisse, vor allem für stationäre Zustände und für Anwendungen, wie etwa die Orientierungsbestimmung einer Drohne, sind diese denkbar. Das Gyroskop kann verwendet werden um die Drehrate eines Rades zu bestimmen z.B. zur Positionsabschätzung. Der Raspberry Pi bot für einfache Aufgaben ausreichend Rechenkapazität. Für komplexere Berechnungen kann hier eine optimiertere Elektronik herangezogen werden. Die Positionsbestimmung durch den Beschleunigungssensor stößt schnell an ihre Grenzen mit den hier verwendeten Methoden. Für eine weitere Verbesserung kann hier eine folgende Arbeit aus der Implementierung eines Kalman Filters sowie die Nutzung mehrerer Sensoren bestehen. Für eine Indoor Positionierung bleibt die Nutzung von andersartigen Referenzsignalen unabdingbar. Eine Nutzung des RSSI von Funkstationen in Kombination mit einem Beschleunigungssensor kann hier ein nächster Schritt sein, der auch mit einem Raspberry Pi realisierbar wäre. Weitere Möglichkeiten wie Nahfeld Kommunikation oder fluoreszierende Lampen sind denkbar. Außerdem bietet sich eine Untersuchung der sich in der Entwicklung befindenden RF Capture Technologie an.

A Literaturverzeichnis

- [1] Pablo Esteban Quiroga Garcia, Wenjie Li: *On Indoor Positioning for Mobile Devices*. Masters Thesis, Göteborg. 2011.
- [2] Tampere University of Technology *Basic Principles of Inertial Navigation*. Seminar on inertial navigation systems.
- [3] Ubejd Shala, Angel Rodriguez: *Indoor Positioning using Sensor-fusion in Android Devices*. Masters Thesis, Kristianstad University. 2011.
- [4] Oliver J. Woodman: *An introduction to inertial navigation*. Technical Report, University of Cambridge. 2007.
- [5] Matej Andrejasic: *MEMS Accelerometers*. Seminar, University of Ljubljana, 2008
- [6] Michal Holcik: *Indoor Navigation for Android*. Masters Thesis, Masaryk University, 2012.
- [7] Chang N, Rashidzadeh R, Ahmadi M. *Robust indoor positioning using differential Wi-Fi access points* IEEE Transactions on Consumer Electronics 56(3):1860-7, 2010.
- [8] Cenk Acar, Andrei M Shkel: *Experimental Evaluation and comparitive analysis of commercial variable-capacitance MEMS accelerometers* University of California, Irvine, 2003.
- [9] Mehmet Akif Erismis: *MEMS Accelerometers and Gyroscopes for inertial measurement units* Masters Thesis, Middle East Technical University, 2004
- [10] Yongyao Cai, Yang Zhao, Xianfeng Ding, James Fennelly *Magnetometer basics for mobile phone applications* memsic, 2012.
- [11] Shahram Moafipoor, Dorota A. Grejner-Brzezinska, Charles Toth *Adaptive Calibration of a Magnetometer Compass for a Personal Navigation System* University of New South Wales, 2007
- [12] Kurt Seifert, Oscar Camacho: *Implementing Positioning Algorithms Using Accelerometers* Freescale Semiconductor Application Note, 2007

-
- [13] Accelerometer, Magnetometer LSM303DLHC Datasheet
 - [14] E Jesper Eklund, Andrei M Shkel, *Single-mask fabrication of high-G piezoresistive accelerometers with extended temperature range* Journal of Micromechanics and Microengineering, 2007.

B Webquellen

- [15] Jay Esfandyari, Roberto De Nuccio, Gang Xu, STMicroelectronics *Introduction to MEMS gyroscopes*, <http://electroiq.com/blog/2010/11/introduction-to-mems-gyroscopes/>, 2010
- [16] LIS331DLH XY Sensor, <http://www.memsjournal.com/2010/12/motion-sensing-in-the-iphone-4-mems-accelerometer.html>
- [17] Hall Effekt, <http://hyperphysics.phy-astr.gsu.edu/hbase/magnetic/hall.html>
- [18] Ozzmaker *Guide to Interfacing a Gyro and Accelerometer with a Raspberry Pi* <http://ozzmaker.com/2014/12/11/berryimu/>
- [19] Wikipedia, Roll-Nick-Gier Winkel <http://de.wikipedia.org/wiki/Roll-Nick-Gier-Winkel>.
- [20] Wikipedia, Flight Dynamics http://en.wikipedia.org/wiki/Flight_dynamics.
- [21] Wikipedia, Drehmatrix <http://de.wikipedia.org/wiki/Drehmatrix>
- [22] Produktbeschreibung 10-DOF IMU Breakout <http://www.adafruit.com/product/1604>
- [23] Produktbeschreibung Raspberry Pi 2 <http://www.raspberrypi.org/blog/raspberry-pi-2-on-sale/>
- [24] RF Capture http://www.csail.mit.edu/RF_capture