

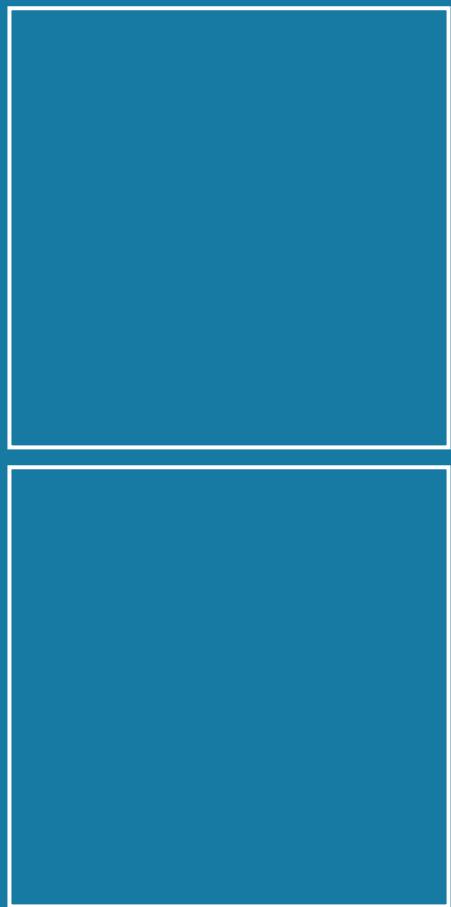


Dissertation

Benedikt Seeger

Charakterisierung der dynamischen Eigenschaften von Sensoren mit digitalem Interface

ISSN 2941-1297
ISBN 978-3-944659-55-8
DOI 10.7795/110.20251017



Physikalisch-Technische Bundesanstalt

Dissertationen
PTB-Diss-23
Braunschweig, Oktober 2025

Benedikt Seeger

Charakterisierung der dynamischen
Eigenschaften von Sensoren mit
digitalem Interface

ISSN 2941-1297
ISBN 978-3-944659-55-8

DOI 10.7795/110.20251017

Empfohlene Zitierweise/recommended citation

Seeger, B., 2025. *Charakterisierung der dynamischen Eigenschaften von Sensoren mit digitalem Interface*. Dissertation, Technische Universität Carolo-Wilhelmina zu Braunschweig. Braunschweig: Physikalisch-Technische Bundesanstalt. PTB-Bericht Diss-23. ISBN 978-3-944659-55-8. Verfügbar unter: <https://doi.org/10.7795/110.20251017>

Impressum:

Physikalisch-Technische Bundesanstalt
ISNI: 0000 0001 2186 1887

Presse und Öffentlichkeitsarbeit

Bundesallee 100
38116 Braunschweig

Telefon: (05 31) 592-93 21
Telefax: (05 31) 592-92 92
www.ptb.de

Charakterisierung der dynamischen Eigenschaften von Sensoren mit digitalem Interface

**Von der Fakultät für Maschinenbau
der Technischen Universität Carolo-Wilhelmina zu Braunschweig**

**zur Erlangung der Würde
eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation**

von: Benedikt Seeger
geboren in: Hildesheim
eingereicht am: 07.11.2024
mündliche Prüfung am: 18.07.2025

Vorsitz: Prof. Dr.-Ing. Michael Müller
Gutachter: Prof. Dr.-Ing. Rainer Tutsch
Prof. Dr. Meinhard Schilling
Dr. Thomas Bruns

2025

Inhaltsverzeichnis

1 Motivation	7
2 Einleitung	9
2.1 Messgröße Beschleunigung	9
2.1.1 Schwingbeschleunigung	10
2.2 Beschleunigungsübertragungsfunktion	12
2.2.1 Analoge Kalibrierung der Übertragungsfunktion nach ISO 16063 – 11	12
2.3 Laser-Doppler-Vibrometer	15
2.4 Sinus-Approximation	17
2.4.1 Drei-Parameter-Sinus-Approximation	17
2.4.2 Multisinus-Approximation	18
2.4.3 Vier-Parameter-Sinus-Approximation	19
2.5 MEMS-Sensoren	21
2.5.1 MEMS-Beschleunigungssensor	23
2.5.2 Modell des Sensorelements	25
2.6 Abtastphasenrauschen und Absolutzeitstempelung	28
3 Digitale Datenerfassungseinheit	29
3.1 Hardware	29
3.1.1 Embedded-Software	33
3.1.2 Konversion und Übertragung der Messwerte in SI-Einheiten	35
3.1.3 C++-Schnittstelle für Sensoren	40
3.2 Software für den Empfang der Protobuf-Nachrichten auf einem Personal Computer (PC)	41
3.2.1 HDF-Dateiformat	43
3.3 GNSS-Zeitbasis	46
3.3.1 Charakterisierung der GNSS-Zeitbasis	48
3.3.2 Analoge Phasenreferenz	53
3.3.3 Kalibrierung der analogen Phasenreferenz	54
4 Nicht-äquidistantes Abtasten und seine Auswirkungen	59
4.1 Bandbreite der Drei-Parameter-Sinus-Approximation	59
4.2 Sequentielle Sinus-Approximation	62
4.2.1 Vier-Parameter-Sinus-Kalibrierung durch sequentielle Drei-Parameter-Sinus-Kalibrierung mit Phasenauswertung	62
4.2.2 Beispiel mit realen Messdaten	64
4.2.3 Limitierung des Verfahrens	64

4.3	Statistische Momente der Approximationsergebnisse in polaren und kartesischen Koordinaten	66
4.3.1	Konversion in Polarkoordinaten	66
4.3.2	Mittelwert und Varianz	66
4.3.3	Simulationen zur Mittelwertbildung und Streuung	68
4.4	Jitter und Phasenrauschen	73
4.4.1	Grafische Darstellung der Auswirkungen von Phasenrauschen	74
4.5	Bestimmung des Phasenrauschens aus Abtastzeitdifferenzen	80
4.5.1	Berechnung der Phasenrauschleistungsdichte aus den Abtastzeitpunkten . .	81
4.5.2	Monte-Carlo-Simulation der Amplitudenabweichungen durch Phasenrauschen	86
4.6	Vergleich von Multisinus-Approximation und DFT	87
5	Kalibrierung digitaler Sensoren	91
5.1	Kalibrierung digitaler Sensoren	92
5.2	Unsicherheitsbetrachtung nach GUM Typ A	94
5.2.1	Anregungs- und Approximationsfrequenz	100
6	Verifizierung des Kalibrierungskonzeptes	101
6.1	Sensoraufbau, Montage und Konfiguration	101
6.1.1	Sensormontage	103
6.2	Schwingungsanregung und Testablauf	104
6.3	Hochfrequenz-Beschleunigungs-Normalmesseinrichtung der PTB	107
6.4	Hochfrequenzmesseinrichtung des CEM	109
6.4.1	Systemaufbau	109
6.4.2	Kalibrierung	110
6.4.3	Einfluss des Montagewinkels	112
6.5	Vergleich der Messungen von PTB und CEM	117
7	Ausblick	121
7.1	Drei-Achsen-Multisensor-Kalibriersystem mit Absolutzeitbasis	121
7.2	Linearität	125
7.3	Die DAU als Smart-Sensor	127
8	Zusammenfassung	131
9	Eigene Veröffentlichungen	141
	Literaturverzeichnis	144
	Tabellenverzeichnis	153
	Abbildungsverzeichnis	154
	Quellcodeverzeichnis	157
	Danksagung	158

10 Anhang	160
10.1 CMCs für Beschleunigung der beteiligten nationalen Metrologieinstitute	160
10.2 Berechnung der Unsicherheitsellipsen mit Pearson-Korrelation in Python	162
10.3 Grafiken	163
10.3.1 Gleichverteilt zufälliges Abtasten bei Sinus-Approximation	163
10.3.2 Verteilung der S3PSA-Ergebnisse bei Frequenzabweichungen	164
10.3.3 Darstellung der Auswirkungen von Phasenrauschen mit linearer Y-Achse . .	164

1 Motivation

Durch die kontinuierliche Miniaturisierung und Digitalisierung in allen technischen Bereichen erfahren mikro-elektromechanische Sensoren (**MEMS**) mit ausschließlich digitalen Schnittstellen in den letzten Jahren eine stark steigende Verbreitung. So befindet sich beispielsweise in fast jedem Smartphone eine **MEMS**-Inertial-Messeinheit (**IMU**) zur Lageerkennung, Bildstabilisierung und als Schrittzähler. Im Jahr 2020 wurden weltweit etwa 17 Milliarden **MEMS**-Sensoren und Aktuatoren mit einem Gesamtumsatz von 15,9 Mrd. USD verkauft [2].

Aufgrund des niedrigen Preises der Sensoren (ab 1 USD) und der Möglichkeit, diese in Systeme des *Internet of Things* (IoT) zu integrieren, werden digitale **MEMS**-Sensoren in Zukunft ubiquitäre Verbreitung finden. Für metrologische oder industrielle Anwendungen ist eine Kalibrierung beziehungsweise Charakterisierung der Sensoren nötig oder von Vorteil [31, 30].

Im Rahmen des europäischen **EMPIR**-Forschungsprojekts **Met4FoF** wurde daher das in dieser Arbeit vorgestellte primäre dynamische Kalibrierverfahren für Beschleunigungs- [83] und Winkelgeschwindigkeitssensoren [86] mit ausschließlich digitalen Schnittstellen entwickelt. Bisher existierten für diese Art von Sensoren nur statische primäre Kalibrierverfahren [92, 77, 36] oder Verfahren, die **MEMS**-Sensoren mit analogem Signalausgang kalibrieren [78, 24]. Die hier vorgestellten Kalibrier- und Charakterisierungsverfahren sind generell für alle dynamischen Messgrößen einsetzbar.

Zur Integration der digitalen Sensoren in bestehende analoge Kalibriersysteme und Messsysteme wurde eine auf einem Mikrocontroller (STMicroelectronics *STM32F7xx*) basierende Datenerfassungseinheit (**Data Acquisition Unit**, **DAU**) entwickelt. Die **DAU** erfasst die Messdaten der Sensoren und versendet diese nach **GNSS**-Absolutzeitstempelung (**GNSS**: *Global Navigation Satellite System*) und Konversion in **SI**-Einheiten über **Ethernet** an einen Empfangs-PC. Der **GNSS**-Empfänger der **DAU** wertet die Signale folgender **GNSS**-Systeme aus: **GPS**, **GLONASS**, **BeiDou** und **Galileo**. Zusätzlich zu den Messdaten werden Metadaten wie beispielsweise der Name der Messgröße, **SI**-Einheit, Quantisierungsauflösung und Messbereichsgrenzen übertragen. Die Empfangssoftware ist flexibel gestaltet, sodass sowohl das Speichern der Daten als auch das Weiterleiten in Sensornetze, wie beispielsweise dem *Multi-Agent System for Industrial Internet of Things* (*agent-Met4FoF*) [60], möglich ist.

Mit Hilfe der **DAU**-Absolutzeitbasis kann auch der Abtasttakt, den die Sensoren intern erzeugen, charakterisiert werden. Dafür wurden mathematische Methoden zur Bestimmung des Abtast-Phasenrauschens entwickelt. Der Einfluss dieses Phasenrauschens sowohl auf dynamische Messungen als auch auf die Kalibrierung der Übertragungsfunktion wurde mittels Simulationen untersucht. Die **DAU** ermöglicht auch den Einsatz der kalibrierten digitalen Sensoren in bestehenden Systemen, wie zum Beispiel dem *Testbed for Electromechanical Cylinders* am **ZeMA** [28]. Die Projektpartner konnten hier zeigen, dass eine Interpolation der nichtäquidistant abgetasteten Messwerte auf eine äquidistante Zeitbasis für die auf maschinellem Lernen basierende Zustandsüberwachung deutliche Verbesserungen in der Vorhersage bringt.

2 Einleitung

Die in dieser Arbeit vorgestellten dynamischen Kalibrierverfahren lassen sich für verschiedene Messgrößen, wie zum Beispiel Beschleunigung, Winkelgeschwindigkeit, magnetische Flussdichte oder Beleuchtungsstärke anwenden. Aktuell werden die Verfahren für die Messgrößen Winkelgeschwindigkeit [86] und Beschleunigung [88] angewendet. In dieser Arbeit wird nur die einachsige primäre dynamische Kalibrierung von Beschleunigungssensoren behandelt. Die vorgestellten Methoden zur Bestimmung des Abtastphasenrauschens von digitalen Sensoren aus den Absolutzeitstempeln sind dabei universell einsetzbar.

2.1 Messgröße Beschleunigung

Die Beschleunigung \vec{a} ist die Änderung der Geschwindigkeit \vec{v} pro Zeit t , und die Geschwindigkeit wiederum ist die Änderung des Ortes \vec{x} pro Zeit. Die vektoriellen Größen sind hier mit einem Pfeil gekennzeichnet.

$$\vec{a} = \frac{\Delta \vec{v}}{\Delta t} = \frac{\Delta \vec{x}}{(\Delta t)^2} \quad (2.1)$$

Die Gleichung 2.1 liefert die mittlere Beschleunigung im Zeitraum Δt . Zur Berechnung der instantanen Beschleunigung $\vec{a}(t)$ müssen die Differenzenquotienten in Differentialquotienten überführt werden.

$$\vec{a}(t) = \frac{d\vec{v}(t)}{dt} = \dot{\vec{v}}(t) = \frac{d^2\vec{x}(t)}{dt^2} = \ddot{\vec{x}}(t) \quad (2.2)$$

Die SI-Einheit der Beschleunigung ist Meter pro Sekunde zum Quadrat. Außerhalb der metrologischen Community ist auch die Angabe von Beschleunigungen in Vielfachen eines Werts für die Erdbeschleunigung $g_n = 9,806\,65 \text{ m/s}^2$ mit dem Einheitensymbol g üblich [64]. Werden Sensoren statisch im Schwerefeld der Erde kalibriert, ist $g = g_{\text{lokal}}$. Dabei kann die Gravitationsbeschleunigung je nach Position auf der Erde um bis zu 50 mm/s^2 von g_n abweichen [43]. Die in den Geowissenschaften verbreitete Einheit Gal entspricht 1 mm/s^2 .

Konstante, gleichförmige Beschleunigungen (statisch/DC) sind für die Geophysik und für die Richtungsbestimmung im Schwerefeld von Interesse. Bei allen anderen Messaufgaben werden dynamische Beschleunigungen mit zum Teil breitbandigem Frequenzspektrum erfasst. Abbildung 2.1 zeigt einen Überblick über die typischen Frequenzen und benötigten Sensitivitäten in verschiedenen Anwendungsfeldern.

Der Frequenzbereich seismischer Messungen reicht von einigen Millihertz bis zu einigen 10 Hz. Trägheitsnavigationssysteme (IMUs) messen üblicherweise im Frequenzbereich von DC bis zu einigen 100 Hz. Klassische Beschleunigungssensor-Messketten, zum Beispiel aus piezoelektrischen Sensoren und Ladungsverstärkern, arbeiten im Frequenzbereich von einigen Hz bis in die Nähe ihrer Resonanzfrequenzen im kHz-Bereich.

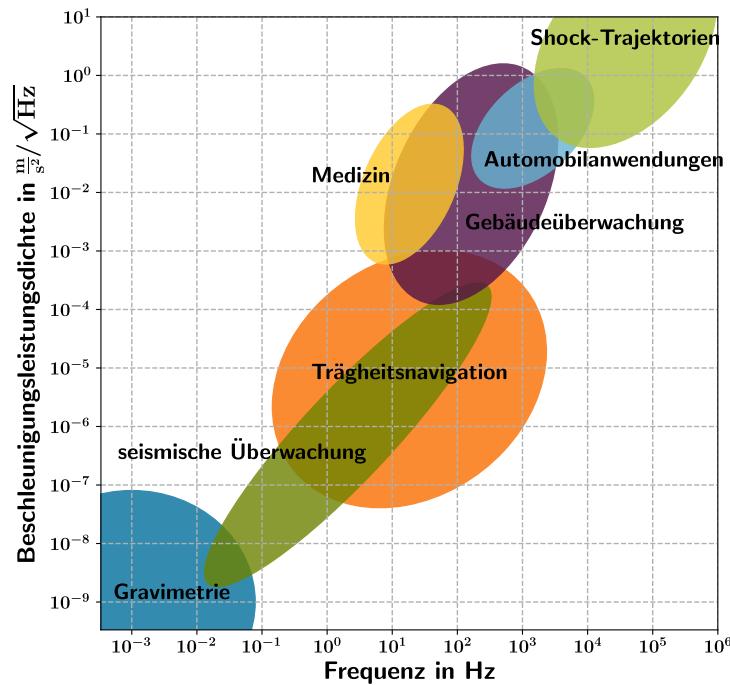


Abbildung 2.1: **Frequenz- und Amplitudenbereiche in der Beschleunigungsmesstechnik.** Verändert aus [57]

Optische Messsysteme wie Laser-Doppler-Vibrometer (**LDV**) können Geschwindigkeitssignale mit Frequenzen von weit über 100 MHz messen, wie sie zum Beispiel bei **Surface-Acoustic-Wave**-Filtern vorkommen [52]. Der Amplitudenbereich reicht von wenigen $\mu\text{m}/\text{s}^2$ bei seismischen Messungen bis hin zu vielen $100 \text{ km}/\text{s}^2$ für Stoßanregungen.

Die in dieser Arbeit untersuchten **MEMS**-Sensoren haben in der Regel eine konfigurierbare Bandbreite von **DC** bis zu einigen Kilohertz. Die Abtastrate wird entsprechend unter Einhaltung des **Nyquist-Shannon-Abtasttheorems** gewählt. Für sehr niedrige Bandbreiten können die Sensoren auch diskontinuierlich abtasten.

2.1.1 Schwingbeschleunigung

Wie im vorherigen Abschnitt erwähnt, werden mit Beschleunigungssensoren hauptsächlich dynamische Signale erfasst. Bei der Kalibrierung dieser Sensoren wird daher sinnvollerweise die komplexwertige Übertragungsfunktion $S(f)$ bestimmt. Die Übertragungsfunktion der Sensoren ist, wie für lineare zeitinvariante Systeme (**LTI**) üblich, im Frequenzraum definiert. Daher folgt nun eine Betrachtung der Bewegungsgleichungen im Frequenzraum. Eine Eigenschaft der **Fourier-Transformation** (**FT**) ($\xrightarrow{\mathcal{F}}$) ist es, dass der Differentialoperator $\frac{d}{dt}$ im Zeitbereich zu einer Multiplikation mit $i\omega$ im Frequenzbereich transformiert wird. Dabei ist i die imaginäre Einheit und $\omega = 2\pi f$ die Kreisfrequenz.

$$\frac{d}{dt} \xrightarrow{\mathcal{F}} i\omega \quad (2.3)$$

Eine Funktion $s(t)$ im Zeitbereich wird wie folgt zu $\mathfrak{S}(\omega)$ Fourier-transformiert.

$$s(t) \xrightarrow{\mathcal{F}} \frac{1}{2\pi} \int_{-\infty}^{+\infty} \mathfrak{S}(\omega) e^{i\omega t} d\omega \quad (2.4)$$

Differenzieren nach der Zeit und Umstellen führt zu folgenden Funktionen.

$$\frac{d}{dt} s(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \mathfrak{S}(\omega) i\omega e^{i\omega t} d\omega \quad (2.5)$$

$$\frac{d}{dt} s(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \{i\omega \mathfrak{S}(\omega)\} e^{i\omega t} d\omega \quad (2.6)$$

Die inverse Transformation $\xleftarrow{\mathcal{F}^{-1}}$ liefert dann die Ableitung im Zeitbereich.

$$\frac{d}{dt} s(t) \xleftarrow{\mathcal{F}^{-1}} i\omega \mathfrak{S}(\omega) \quad (2.7)$$

Für den Integraloperator ergibt sich analog dazu eine Division durch $i\omega$ im Frequenzraum.

$$\int dt \xrightarrow{\mathcal{F}} \frac{1}{i\omega} \quad (2.8)$$

Für sinusförmige Signale, wie sie bei der Kalibrierung von Beschleunigungssensoren nach ISO 16063 [3] verwendet werden, können daher folgende einfache funktionale Zusammenhänge für die Bewegungsgleichungen angesetzt werden. Hierbei werden Amplituden und Phasen der Sinusschwingungen durch \hat{x} und (φ) gekennzeichnet. Es erfolgt eine Betrachtung von skalaren Größen.

$$x = -\frac{iv}{\omega} = -\frac{a}{\omega^2} = \hat{x} e^{i(\omega t + \varphi)} = \hat{x} [\cos(\omega t + \varphi) + i \sin(\omega t + \varphi)] \quad (2.9)$$

$$\downarrow \frac{d}{dt} \qquad \qquad \qquad \uparrow \int dt$$

$$i\omega x = v = -\frac{ia}{\omega} = \hat{v} e^{i(\omega t + \varphi + \frac{\pi}{2})} = \hat{v} [\cos(\omega t + \varphi + \frac{\pi}{2}) + i \sin(\omega t + \varphi + \frac{\pi}{2})] \quad (2.10)$$

$$\downarrow \frac{d}{dt} \qquad \qquad \qquad \uparrow \int dt$$

$$-\omega^2 x = i\omega v = a = \hat{a} e^{i(\omega t + \varphi + \pi)} = \hat{a} [\cos(\omega t + \varphi + \pi) + i \sin(\omega t + \varphi + \pi)] \quad (2.11)$$

Dabei kennzeichnen x , v und a die Größen Schwingweg, Geschwindigkeit und Beschleunigung, \hat{x} , \hat{v} und \hat{a} sind die entsprechenden Amplituden und φ die Initialphase. Für sinusförmige Signale können folglich Position, Geschwindigkeit und Beschleunigung ohne numerische Differentiation oder Integration ineinander überführt werden.

2.2 Beschleunigungsübertragungsfunktion

Die komplexwertige Übertragungsfunktion $S(f)$ beschreibt die Sensitivität des zu kalibrierenden Sensors (Device Under Test, DUT) bezogen auf eine bekannte Referenz (REF) bei einer definierten Anregungsfrequenz f . Die Übertragungsfunktion ist nach ISO 16063 – 11 [3] wie folgt definiert.

$$S(f) = \frac{a_{\text{DUT}}(f)}{a_{\text{REF}}(f)} \quad (2.12)$$

Der Betrag der Übertragungsfunktion kann direkt aus den Amplituden berechnet werden.

$$|S(f)| = \frac{\hat{a}_{\text{DUT}}(f)}{\hat{a}_{\text{REF}}(f)} \quad (2.13)$$

Die Phasenverzögerung $\Delta\varphi(f) = \arg(S(f))$ lässt sich aus den initialen Phasen $\varphi_{\text{DUT}}(f)$ und $\varphi_{\text{REF}}(f)$ wie folgt berechnen.

$$\Delta\varphi(f) = \varphi_{\text{REF}}(f) - \varphi_{\text{DUT}}(f) = \arg(S(f)) \quad (2.14)$$

Die in dieser Arbeit kalibrierten MEMS-Beschleunigungssensoren haben oft drei orthogonale Sensorelemente für die drei kartesischen Raumrichtungen x, y und z. Für diese Sensoren können die Sensitivitätskoeffizienten durch die Sensitivitätsmatrix $\mathbf{S}(f)$ beschrieben werden, siehe z. B. [101].

$$\mathbf{S}(f) = \begin{pmatrix} S_{xx}(f) & S_{xy}(f) & S_{xz}(f) \\ S_{yx}(f) & S_{yy}(f) & S_{yz}(f) \\ S_{zx}(f) & S_{zy}(f) & S_{zz}(f) \end{pmatrix} = \begin{pmatrix} \frac{a_{\text{DUT}_x}(f)}{a_{\text{REF}_x}(f)} & \frac{a_{\text{DUT}_x}(f)}{a_{\text{REF}_y}(f)} & \dots \\ \frac{a_{\text{DUT}_y}(f)}{a_{\text{REF}_x}(f)} & \frac{a_{\text{DUT}_y}(f)}{a_{\text{REF}_y}(f)} & \dots \\ \dots & \dots & \frac{a_{\text{DUT}_z}(f)}{a_{\text{REF}_z}(f)} \end{pmatrix} \quad (2.15)$$

In dieser Arbeit wird nur die Sensitivität $S_{zz}(f)$ betrachtet, welche im Weiteren vereinfacht als $S(f)$ bezeichnet wird.

2.2.1 Analoge Kalibrierung der Übertragungsfunktion nach ISO 16063 – 11

Die in dieser Arbeit behandelten einachsigen Kalibrierungen von "rectilinear accelerometers" nach ISO 16063, also Beschleunigungssensoren für geradlinige Beschleunigung (im Gegensatz zur Winkelbeschleunigung), orientierten sich im Wesentlichen an der ISO-Norm 16063-11 1999 *Methods for the calibration of vibration and shock transducers - Part 11: Primary vibration calibration by laser interferometry*. Dabei wird eine **Drei-Parameter-Sinus-Approximation** zur Bestimmung der Amplituden und initialen Phasen verwendet, die als Methode Drei (*sine-approximation method*) in der Norm gelistet ist. Die Übertragungsfunktion wird entsprechend den Gleichungen 2.12 bis 2.14 als komplexwertiger Quotient von Sensorsignal und Referenzsignal berechnet.

Die Abbildung 2.2 zeigt beispielhaft den Aufbau eines einachsigen primären Kalibriersystems mit einem LDV als Referenz. Im Folgenden wird dieses System als analoges Kalibriersystem (ACS) bezeichnet. Eine ausführliche Beschreibung des Kalibriersystems der PTB findet sich im Abschnitt 6.3.

Im ACS werden die Erzeugung des Anregungssignals (orange) und die Messwerterfassung (grün) synchronisiert durchgeführt. Alle Takte und Frequenzen sind von einer gemeinsamen Zeitbasis (violett) t_i abgeleitet.

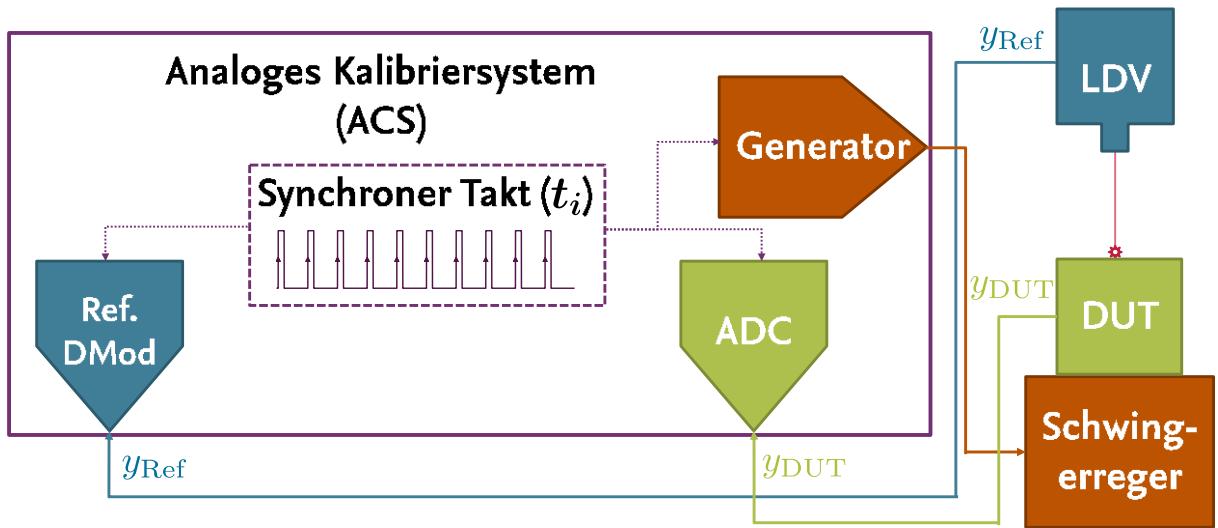


Abbildung 2.2: **Primäres einachsiges Beschleunigungskalibriersystem nach ISO 16063-11 [3] Methode 3: Sinus-Approximation.** Das DUT wird durch einen **Schwingerreger** mit einer sinusförmigen Bewegung mit der Frequenz f angeregt. Mittels **LDV** wird die **momentane Beschleunigung** $y_{Ref,f}(t_i)$ gegen eine **äquidistante Relativzeitbasis** t_i abgetastet. Um eine primäre Kalibrierung zu erreichen, wird das **Heterodyn-Signal** des **LDV** durch einen **ADC** erfasst und anschließend demoduliert (Ref. DMod). Die vom **DUT** erfasste **Beschleunigung** $y_{DUT,f}(t_i)$ wird dazu synchron abgetastet. Der komplexe Übertragungskoeffizient $S(f)$ ist der Quotient aus den durch die **Drei-Parameter-Sinus-Approximation** bestimmten komplexen Beschleunigungen von **DUT** und **Referenz** nach Gleichung 2.12.

Die mechanische Schwingerregerregung erfolgt durch einen membran- oder luftgelagerten elektrodynamischen Schwingerreger (**orange**). Der Schwingerreger hat, wie auch elektrodynamische Lautsprecher, eine sehr niedrige Wechselstromimpedanz, weshalb ein Leistungsverstärker zur Impedanzanpassung an den Signalgenerator des **ACS** erforderlich ist. Um eine Einkopplung von Vibrationen des Schwingerregers auf das Referenz-**LDV** (**blau**) zu verhindern, befinden sich Schwingerreger und **LDV** auf durch Schwingungsdämpfern isolierten, schweren Fundamenten (siehe Abbildung 6.5). Hierbei sollte die Masse der Fundamente laut ISO 16063 mindestens 2000-mal größer sein als die kombinierte Masse von Schwingerreger und Sensor. Die Kalibrierung wird sukzessive für verschiedene Anregungsfrequenzen $f = \frac{\omega}{2\pi}$ durchgeführt. Die Frequenzen einer Reihe werden dabei in der Regel in Anlehnung an die $\frac{1}{3}$ -Oktaven nach ISO 266 [47] gewählt. Die $\frac{1}{3}$ -Oktavfrequenzen bezogen auf eine Referenzfrequenz f_r werden wie folgt berechnet.

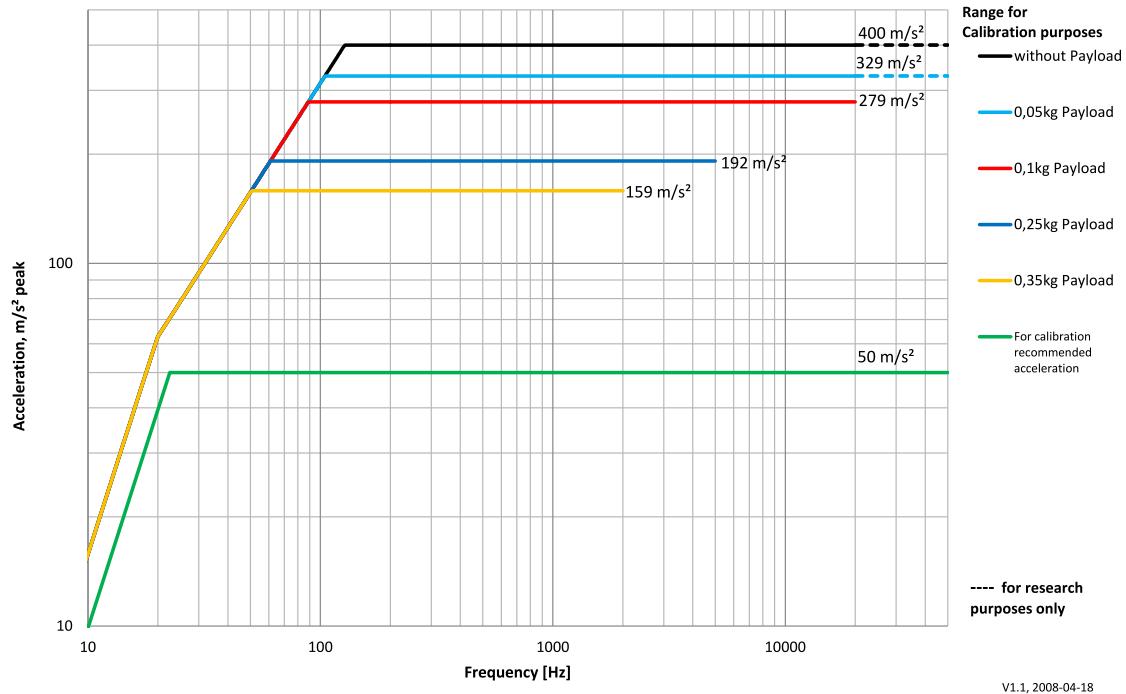
$$f = 10^{\frac{n}{10}} \cdot f_r \quad (2.16)$$

Auf „glatte“ Werte gerundet ergeben sich die 10 in Tabelle 2.1 aufgeführten Frequenzskalierungsfaktoren.

Die Anregungsschwingungen sollen nach Norm so aus den Amplituden $[1, 2, 5] \cdot 10^n \text{ m/s}^2$ ausgewählt werden, dass der Messbereich der Messkette gut ausgesteuert wird. Für tiefe Frequenzen ($f \leq 80 \text{ Hz}$ für die **Hochfrequenz-Beschleunigungs-Normmesseinrichtung (HF-B-NME)** der **PTB**) wird die erreichbare Beschleunigungsschwingung durch den maximalen Schwingweg von 7,5 mm des Erre-

Tabelle 2.1: **Bevorzugte $\frac{1}{3}$ -Oktaven-Frequenzskalierungsfaktoren in Anlehnung an [47]**

Index	n	0	1	2	3	4	5	6	7	8	9
Frequenzskalierungsfaktor	$10^{\frac{n}{10}}$	1	1,25	1,6	2	2,5	3,15	4	5	6,3	8,0

Abbildung 2.3: **Erreichbare Beschleunigungen eines Schwingerreger Typ Spektra SE-09 laut Datenblatt [102].**

gers begrenzt. Die Abbildung 2.3 zeigt grafisch die erreichbaren frequenzabhängigen Beschleunigungen für verschiedene Massenbelastungen für den SE-09 Schwingerreger. Für eine Frequenz von 10 Hz führt schon eine Schwingbeschleunigung von $\approx 1,6 \text{ m/s}^2$ zu einem Schwingweg von 7,5 mm (siehe Gleichung 2.9). Anderenfalls begrenzt die auf die Schwingarmatur wirkende Kraft ($F = ma$) die Beschleunigung des Schwingerregers.

Für niedrige Frequenzen werden die höchstmöglichen Norm-Amplituden verwendet (siehe Tabelle 6.2).

Für die Messdauer je Frequenzpunkt enthält die Norm keine Angaben. Die Messdauer wird entsprechend des Signal-Rausch-Verhältnisses so lang wie nötig, aber so kurz wie möglich gewählt. In dieser Arbeit wurden in der Regel mindestens 30 s Daten pro Frequenzpunkt erfasst.

Das ACS besitzt eine interne Zeitbasis, gegen die die Schwingssignalerzeugung als auch die Datenerfassung synchronisiert sind.

Für die Qualität der Zeitbasis benennt die ISO 16063-11 folgende minimale Anforderungen:

- Relative Frequenzunsicherheit von maximal 0,05 % (50 ppm)
- Frequenzabweichung während einer Messperiode von maximal 0,05 % (50 ppm)
- Amplitudenstabilität während einer Messperiode von maximal 0,05 % (50 ppm)

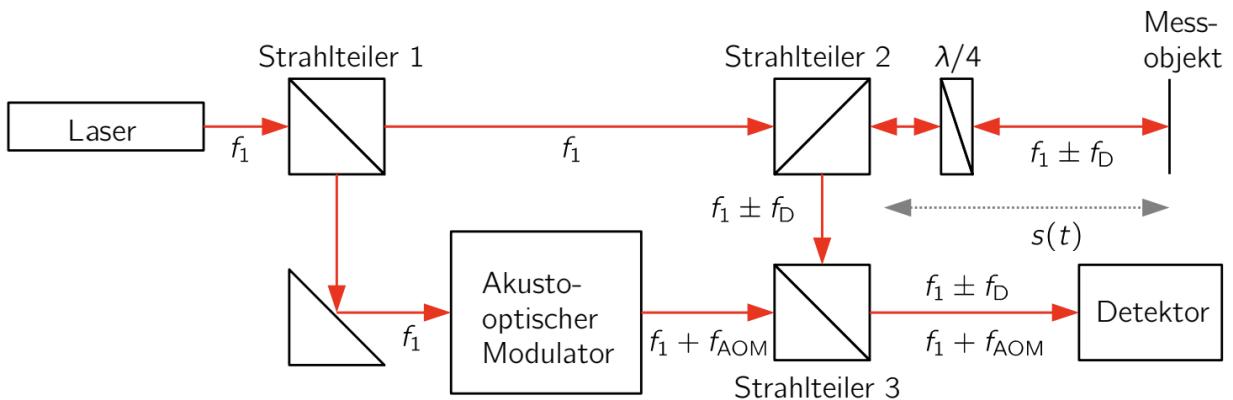


Abbildung 2.4: **Aufbau eines LDV**. Entnommen aus [51].

Durch die synchrone Signalerzeugung und Erfassung führen statische Frequenzabweichungen oder eine Drift der Zeitbasis während der Messung zu vollständig korrelierten Abweichungen der momentanen Beschleunigungsmesswerte. Bei der komplexen Division (Gleichung 2.12) kürzen sich diese korrelierten Einflüsse vollständig heraus und können daher in diesem Fall vernachlässigt werden.

2.3 Laser-Doppler-Vibrometer

Für die primäre Kalibrierung von Beschleunigungssensoren nach ISO 16063-11 [3] werden Laser-Doppler-Vibrometer (LDV) als Referenz verwendet. Ein LDV ist im Wesentlichen ein heterodynes Interferometer mit einem geeigneten Objektiv, welches den Laserstrahl auf das Messobjekt fokussiert und das in das Gerät zurück reflektierte Licht auf den Detektor abbildet. Die Grafik 2.4 zeigt schematisch den Strahlengang innerhalb eines LDV. Als Laserquelle sollen nach der oben genannten Norm Helium-Neon-Laser verwendet werden. Dieser Laser emittiert dominierend Strahlung, die aus einem $3s_2 - 2p_4$ -Übergang im Neon-Atom stammt. Die Energie dieses Übergangs ist eine atomare Konstante, entsprechend ist auch die emittierte Wellenlänge (bzw. Frequenz) konstant. Die Wellenlänge dieses Übergangs ist in Luft als Medium $\lambda_L = 632,81 \text{ nm}$ [15]. Dies entspricht einer Frequenz f_1 von

$$f_1 = \frac{c}{\lambda_L} \approx 473,61 \text{ THz}, \quad (2.17)$$

dabei sind c die Lichtgeschwindigkeit, $n_L = 1,000\,268\,531 \pm 0,000\,000\,024$ [98] der Brechungsindex von Luft bei Normalbedingungen, Temperatur $T = 23 \text{ }^\circ\text{C}$, Druck $p = 1023,25 \text{ hPa}$, und die relative Luftfeuchtigkeit $RH = 50\%$ sowie die CO_2 -Konzentration $C_{\text{CO}_2} = 450 \text{ ppm}$. Durch den polarisierenden Strahlteiler 1 wird das vom Laser emittierte Licht in zwei jeweils linear polarisierte Mess- und Referenzstrahlen aufgeteilt. Der Referenzstrahl wird durch einen akustooptischen Modulator (AOM) geführt, welcher die Frequenz des Laserlichts um die Modulationsfrequenz f_{AOM} verschiebt. In kommerziellen Geräten sind AOM-Modulationsfrequenzen von 40 MHz oder 80 MHz üblich. Der Messstrahl tritt gerade durch den Strahlteiler 2 hindurch und gelangt durch ein $\lambda/4$ -Plättchen auf das Messobjekt. Das linear polarisierte Licht des Messstrahls ist durch die Passage des $\lambda/4$ -Plättchens nun zirkular polarisiert. Die Bewegungskomponente des Messobjekts entlang

der Strahlrichtung erzeugt eine Doppler-Frequenzverschiebung um die Frequenz f_D . Bei der erneuten Passage des $\lambda/4$ -Plättchens auf dem Rückweg wird die zirkulare Polarisation in eine zum Ausgangszustand um 90° phasenverschobene lineare Polarisation geändert. Wegen dieser Polarisation wird der gesamte zurücklaufende Strahl am polarisierenden Strahlteiler 2 in Richtung des Strahlteilers 3 reflektiert. Von diesem Strahlteiler gelangen Referenzstrahl und Messstrahl zu einem Photodetektor. Auf dem Photodetektor kommt es durch die Überlagerung der elektromagnetischen Wellen des Referenzstrahls E_R und des Messstrahls E_M zu einer Amplitudenmodulation infolge von Interferenz [69]. Die elektrischen Feldstärken \hat{E} der Strahlen können wie folgt beschrieben werden:

$$E_R = \hat{E}_R \cdot \sin[2\pi(f_1 + f_{AOM}) \cdot t + \varphi_0] \quad (2.18)$$

$$E_M = \hat{E}_M \cdot \sin[2\pi(f_1 + f_D) \cdot t + \varphi_0] \quad (2.19)$$

Dabei ist φ_0 die Initialphase des Lasers. Wegen der unterschiedlichen Signalfrequenzen kommt es auf dem Detektor zu einer Schwebung. Dabei lässt sich die Feldstärke auf dem Detektor durch folgende Gleichung beschreiben.

$$E = E_R^2 + E_M^2 + \hat{E}_R \hat{E}_M \cos(2\pi[(f_1 + f_{AOM}) - (f_1 + f_D)] \cdot t) \quad (2.20)$$

Die vom Photodetektor erfasste Intensität ist proportional zum Quadrat der Feldstärke und es ergibt sich folgende Gleichung.

$$I = \underbrace{I_R + I_M}_{\text{konstant}} + 2\sqrt{\hat{I}_R \hat{I}_M} \underbrace{\cos(2\pi(f_{AOM} + f_D) \cdot t)}_{\text{moduliert}} \quad (2.21)$$

Dabei wird angenommen, dass die Bandbreite des Detektors klein gegen die Strahlungsfrequenz f_1 ist. Dies ist für die Laserstrahlung mit Frequenzen im Terahertz-Bereich, siehe Gleichung 2.17, immer erfüllt.

$$f_D = \frac{2 \cdot v}{c} f_1 = \frac{2v}{\lambda_L} \quad (2.22)$$

Die Dopplerfrequenz f_D des Messstrahls ist proportional zur Geschwindigkeit des Messobjekts: Durch die Reflexion an der Oberfläche werden sowohl der hin- als auch der zurücklaufende Messstrahl jeweils um den gleichen Faktor frequenzverschoben. Dies erklärt den Faktor zwei in der Berechnung der Dopplerfrequenz. Für ein **LDV** mit $f_{AOM} = 40\text{ MHz}$ und einer Hochpassgrenzfrequenz von $f_{HP} = 1\text{ MHz}$ ist die maximale Geschwindigkeit v_{\max} , die trivial¹ gemessen werden kann, folglich

$$v_{\max} = (f_{AOM} - f_{HP}) \cdot \frac{\lambda_L}{2} = 39\text{ MHz} \cdot \frac{632,8\text{ nm}}{2} = 12\text{ m/s.} \quad (2.23)$$

Infolge der Frequenzverschiebung des Referenzstrahls durch den **AOM** werden die prinzipiellen Vorteile des Heterodyninterferometers erreicht.

1. Es kann ohne eine zusätzliche Quadraturdetektion eine Richtungsinformation erhalten werden.

¹ohne die Verwendung von *Tracking*-Algorithmen oder Quadraturdetektion und mit Hochpassfilterung des Photodetektorsignals.

2. Die tieffrequenten Laseramplitudenschwankungen können von dem hochfrequenten Nutzsignal durch analoge oder digitale Filter getrennt werden.
3. Durch Phasendemodulation können auch sehr kleine Wegänderungen bzw. Schwingwegamplituden aufgelöst werden, wenn entsprechend lange gemessen wird.
4. Die kontinuierliche Modulation führt zur Mittelung von Nichtlinearitätseffekten.

2.4 Sinus-Approximation

Bei der Kalibrierung nach ISO-16063-11 werden sinusförmige Bewegungen verwendet. Aus ihren Amplituden und initialen Phasen wird dann unter linearer Näherung (**LTI**-System) der komplexe Übertragungskoeffizient berechnet. Zur Bestimmung von Amplitude und Phase aus den Messwert-Zeitreihen wird eine **Drei-Parameter-Sinus-Approximation (3PSA)** oder eine **Vier-Parameter-Sinus-Approximation (4PSA)** nach *IEEE 1057 Standard for Digitizing Waveform Recorders* [45] verwendet. Die hier gezeigte Herleitung der Drei- und Vier-Parameter-Sinus-Approximation orientiert sich an *Peter Händels* Veröffentlichung *Properties of the IEEE-STD-1057 four-parameter sine wave fit algorithm* [39]. An den nicht notwendigerweise äquidistanten Punkten $t_1, \dots, t_n, \dots, t_N$ werden die Messwerte $y_1, \dots, y_n, \dots, y_N$ abgetastet. Diese Werte sollen durch die Funktion

$$y_n[A, B, C, \omega] = A \cos \omega t_n + B \sin \omega t_n + C \quad (2.24)$$

mit den Parametern Realteil A , Imaginärteil B und Gleichanteil C modelliert werden. Bei der **3PSA** ist die Kreisfrequenz ω bekannt, bei der **4PSA** wird die Kreisfrequenz ausgehend von einem Startwert iterativ bestimmt.

Äquivalent zu Gleichung 2.24 kann auch unter Verwendung von $A = \alpha \sin \phi$ und $B = \alpha \cos \phi$ folgende Modellgleichung zur Beschreibung der Zeitreihe mit dem Parametervektor θ verwendet werden.

$$y_n[\theta] = \alpha \sin(\omega t_n + \phi) + C \quad (2.25)$$

2.4.1 Drei-Parameter-Sinus-Approximation

Bei bekannter Kreisfrequenz ω können die Sinus-Parameter \mathbf{x} direkt durch eine Berechnung mittels **Ordinary Least Squares (OLS)**, auch bekannt als Methode der kleinsten Abweichungsquadrate erhalten werden. Die abgetasteten Werte

$$\mathbf{y} = (y_1, \dots, y_N)^T \quad (2.26)$$

werden durch das überbestimmte lineare Gleichungssystem

$$\mathbf{y} = \mathbf{D}\mathbf{x} \quad (2.27)$$

mit der $N \times 3$ Designmatrix \mathbf{D} beschrieben.

$$\mathbf{D} = \begin{pmatrix} \cos \omega t_1 & \sin \omega t_1 & 1 \\ \cos \omega t_2 & \sin \omega t_2 & 1 \\ \vdots & \vdots & \vdots \\ \cos \omega t_N & \sin \omega t_N & 1 \end{pmatrix} \quad (2.28)$$

Das **OLS**-Verfahren approximiert die Parameter nun durch folgende Gleichung.

$$\hat{\mathbf{x}} = (\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \mathbf{y} \quad (2.29)$$

Für den Spezialfall von idealem äquidistanten Abtasten der N Messwerte mit der Abtastfrequenz f_s ergibt sich für alle Kreisfrequenzen ω , die die Fourier-Bedingungen

$$\frac{\omega}{2\pi} \bmod \frac{f_s}{2N} = 0, \quad (2.30)$$

$$\text{mit } \omega \bmod f_s \neq 0 \quad (2.31)$$

erfüllen, für die Inverse $(\mathbf{D}^T \mathbf{D})^{-1}$ der Gram-Matrix eine Einheitsmatrix. In diesen Fällen ist die **3PSA** folglich äquivalent zu einer Linie der DFT [45].

2.4.2 Multisinus-Approximation

Die Erweiterung der **3PSA** auf mehrere Frequenzen ist durch das Einfügen der entsprechenden Spalten in die Designmatrix \mathbf{D} einfach möglich.

$$\mathbf{D} = \begin{pmatrix} \cos \omega_1 t_1 & \sin \omega_1 t_1 & \cos \omega_2 t_1 & \sin \omega_2 t_1 & \dots & \cos \omega_k t_1 & \sin \omega_k t_1 & 1 \\ \cos \omega_1 t_2 & \sin \omega_1 t_2 & \cos \omega_2 t_2 & \sin \omega_2 t_2 & \dots & \cos \omega_k t_2 & \sin \omega_k t_2 & 1 \\ \vdots & \vdots \\ \cos \omega_1 t_N & \sin \omega_1 t_N & \cos \omega_2 t_N & \sin \omega_2 t_N & \dots & \cos \omega_k t_N & \sin \omega_k t_N & 1 \end{pmatrix} \quad (2.32)$$

Das Lösen des Gleichungssystems erfolgt analog zur **3PSA**.

$$\hat{\mathbf{x}} = (\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T \mathbf{y} \quad (2.33)$$

Werden alle Frequenzen bis zur halben Abtastfrequenz unter Einhaltung der Fourier-Bedingungen (Gleichung 2.30 und 2.31) approximiert, entspricht die Multisinus-Approximation der Fouriertransformation. Die Multisinus-Approximation kann, wie im Abschnitt ?? beschrieben, zur Untersuchung der Amplituden-Nichtlinearität von Sensoren verwendet werden, indem die resultierenden Harmonischen bei sinusförmiger Anregung ausgewertet werden.

Minimaler Frequenzabstand und maximale Linienzahl

Auch für die Multisinus-Approximation gilt die *Küpfmüllersche Unbestimmtheitsrelation*; daher kann der Frequenzabstand zwischen zwei Frequenzlinien i, j nicht kleiner als

$$\Delta f = \left| \frac{\omega_i}{2\pi} - \frac{\omega_j}{2\pi} \right| \geq \frac{1}{T_W} \quad (2.34)$$

gewählt werden [27]. Wie in Gleichung 43 gezeigt, führt die limitierte Beobachtungsdauer durch die Multiplikation mit dem rechteckigen Beobachtungsfenster der Länge T_W im Zeitbereich zu einer Faltung der Linien im Frequenzraum mit einer Sinc-Funktion der Breite $\frac{1}{T_W}$. Analog zum Abbe-Limit bei der optischen Abbildung können diese verbreiterten Linien nicht beliebig scharf aufgelöst werden [20].

Mit der Multisinus-Approximation können auch zeitgleich direkt und unterabgetastete Signale approximiert werden. Wenn periodisch mit der Abtastrate f_s abgetastet wird, müssen alle Signalfrequenzkombinationen folgende Bedingung erfüllen.

$$\left| \left(f_1 \bmod \frac{f_s}{2} \right) - \left(f_2 \bmod \frac{f_s}{2} \right) \right| \geq \Delta f = \frac{1}{T_W} \quad (2.35)$$

Werden alle Frequenzen so gewählt, dass sie nach dem Heruntermischen am nächsten Vielfachen der halben Abtastfrequenz ins Basisband $f_{\text{Basisband}} = f \bmod \frac{f_s}{2}$ den minimalen Abstand $\Delta f = \frac{1}{T_W}$ haben, können nach dem *Nyquist-Shannon Abtasttheorem* $\frac{N}{2}$ unterschiedliche Frequenzen aufgelöst werden.

2.4.3 Vier-Parameter-Sinus-Approximation

Die *Vier-Parameter-Sinus-Approximation (4PSA)* nach IEEE-1057 [45] ist ein iteratives Verfahren, welches durch Linearisierung mit dem ersten Taylor-Polynom sukzessive die Kreisfrequenz bestimmt.

In jedem Iterationsschritt j der Approximation für die Kreisfrequenz ω_j unter Verwendung des ersten Taylor-Glieds an der Entwicklungsstelle ω ergibt sich

$$\cos \omega t_n \approx \cos \hat{\omega}_j t_n - t_n \sin \hat{\omega}_j t_n \cdot \Delta \omega_j, \quad (2.36)$$

$$\sin \omega t_n \approx \sin \hat{\omega}_j t_n + t_n \cos \hat{\omega}_j t_n \cdot \Delta \omega_j. \quad (2.37)$$

Dabei ist $\Delta \omega_j = \omega - \hat{\omega}_j$. Das Einfügen der Näherungen 2.37 und 2.36 in Gleichung 2.24 liefert

$$y_n[\theta] \approx A \cos \hat{\omega}_j t_n + B \sin \hat{\omega}_j t_n + C - A t_n \Delta \omega_j \sin \hat{\omega}_j t_n + B t_n \Delta \omega_j \cos \hat{\omega}_j t_n \quad (2.38)$$

mit dem Parametervektor $\theta = \mathbf{x}_j$.

$$\mathbf{x}_j = (A, B, C, \Delta \omega_j)^T \quad (2.39)$$

Die Gleichung 2.38 ist für die Parameter weiterhin nichtlinear, kann aber mit der Annahme, dass $\Delta \omega_j \approx 0$ sei, linearisiert werden. Dafür werden anstelle der letzten beiden Terme in 2.38 die Schätzwerte aus der vorherigen Iteration für A und B verwendet, also \hat{A}_{j-1} und \hat{B}_{j-1} . Dies führt zu einer

Linearisierung bezogen auf die Komponenten des Parametervektors x_j .

Entsprechend zu Gleichung 2.27 werden die erfassten Messwerte \mathbf{y} mit der $N \times 4$ Matrix $\hat{\mathbf{D}}_j$ für jeden Iterationsschritt approximiert als

$$\hat{\mathbf{D}}_j = \begin{pmatrix} \cos \hat{\omega}_j t_1 & \sin \hat{\omega}_j t_1 & 1 & -\hat{A}_{j-1} t_1 \sin \hat{\omega}_j t_1 + \hat{B}_{j-1} t_1 \cos \hat{\omega}_j t_1 \\ \cos \hat{\omega}_j t_2 & \sin \hat{\omega}_j t_2 & 1 & -\hat{A}_{j-1} t_2 \sin \hat{\omega}_j t_2 + \hat{B}_{j-1} t_2 \cos \hat{\omega}_j t_2 \\ \vdots & \vdots & \vdots & \vdots \\ \cos \hat{\omega}_j t_N & \sin \hat{\omega}_j t_N & 1 & -\hat{A}_{j-1} t_N \sin \hat{\omega}_j t_N + \hat{B}_{j-1} t_N \cos \hat{\omega}_j t_N \end{pmatrix}. \quad (2.40)$$

In der Tabelle 2.2 wird das iterative Vorgehen bei der Optimierung zusammengefasst. Die initialen Parameter für die Approximation müssen entweder durch Vorwissen, z. B. Nominalwerte der Anregung, oder durch ein anderes Verfahren, wie z. B. die Position des DFT-Maximums, ermittelt werden.

Tabelle 2.2: **Algorithmus zur Vier-Parameter-Sinus-Approximation nach Methode der kleinsten Quadrate (OLS) [39]**

-
- a) $\hat{\mathbf{x}}_0 = (A_0, B_0, C_0, 0)^T$, $\hat{\omega}_j = \hat{\omega}_0$, $j = 0$
 - b) Nächste Iteration $j = j + 1$
 - c) $\hat{\omega}_j = \hat{\omega}_{j-1} + \Delta \hat{\omega}_{j-1}$
 - d) Erstelle $\hat{\mathbf{D}}_j$, siehe (2.40)
 - e) $\hat{\mathbf{x}}_j = (\hat{\mathbf{D}}_j^T \hat{\mathbf{D}}_j)^{-1} \hat{\mathbf{D}}_j^T \mathbf{y}$
 - g) Wiederhole Schritte b) ... e) bis zur Konvergenz

Bei der iterativen Durchführung des Algorithmus nach Tabelle 2.2 wird sukzessive folgende Summe der Abweichungsquadrate minimiert.

$$\sum_{n=1}^N (y_n - y_n[\theta])^2 \quad (2.41)$$

Die Parameter A , B und ω bzw. \hat{A} , φ und ω sind nicht informationsorthogonal und können daher nicht unabhängig voneinander minimiert werden [39].

2.5 MEMS-Sensoren

Die Abkürzung **MEMS** steht für *micro-electro-mechanical-systems*, also miniaturisierte elektromechanische Systeme, wie zum Beispiel Beschleunigungssensoren, Gyroskope, Mikrofone, Oszillatoren oder Mikrospiegel-Arrays. In dieser Arbeit werden als Sensor nur die innerhalb eines Chipgehäuses verbauten Komponenten betrachtet. Zum Auslesen und Weiterverarbeiten der Daten ist zusätzlich noch ein Host-Prozessor nötig. Durch die fortschreitende Miniaturisierung erscheinen immer mehr Sensoren, die einen Host-Prozessor enthalten, auf dem Markt. Bei diesen Sensoren handelt sich um vollständige Messsysteme. Die Kombination von Mechanik und Elektronik erlaubt die Integration einer oder mehrerer vollständiger Messketten aus Sensorelementen, Signalkonditionierung, Analog-to-Digital Converter (ADC), digitaler Filterung/Verarbeitung sowie Takt- und Interrupterzeugung in einem **MEMS**-Sensor.

Abbildung 2.5 zeigt den grundlegenden Aufbau von **MEMS**-Sensoren mit ausschließlich digitalem Interface.

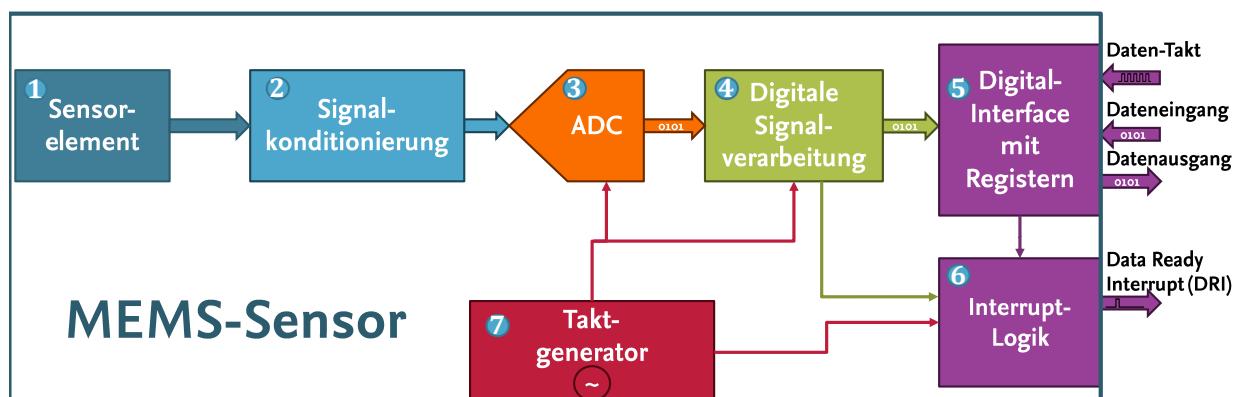


Abbildung 2.5: **Grundlegender Aufbau von MEMS-Sensoren mit digitaler Schnittstelle.** Ein Sensorelement ① wandelt die physikalische Messgröße (z. B. Beschleunigung) in eine elektrische Größe (z. B. Kapazität) um. Dieses elektrische Signal wird durch analoge Schaltungen ② konditioniert (verstärkt, Ladungs-Spannungs-Konversion), um das Eingangssignal für den ADC ③ zu erzeugen. Der Takt des ADC wird durch einen internen Taktgenerator ⑦ erzeugt; einige wenige Sensoren erlauben auch die Verwendung einer externen Taktquelle. Die digitalisierten Messdaten werden mittels digitaler Signalverarbeitung behandelt (Hochpass, Tiefpass, Dezimierung uvm.). Anschließend werden Daten in Registern abgelegt, aus denen, die Daten über das digitale Interface ⑤ einzeln oder als Block aus einem **FIFO**-Puffer ausgelesen werden können. Der Takt des digitalen Interfaces und des ADC sind verschieden. Das digitale Interface wird vom Host getaktet und der Takt liegt nicht kontinuierlich an. Durch die Interrupt-Logik ⑥ können die Sensoren verschiedene Ereignisse, z. B. das Anliegen neuer Daten in den Registern (**Data Ready Interrupt**) oder das Über/Unterschreiten von Schwellwerten (**Window-Interrupt**), über eine Signalleitung signalisieren. Auch veröffentlicht in [8o]

Ein elektromechanisches Sensorelement ① konvertiert die physikalische Messgröße zum Beispiel Beschleunigung in ein elektrisches Signal. Bei Beschleunigungssensoren werden oft kapazitive Auswerteverfahren in Kombination mit einer kapazitiven Brückenschaltung verwendet. Aber auch Widerstandsmessbrücken oder frequenzerzeugende Messaufnehmer sind möglich. In den **MEMS**-

Sensoren sind analoge Signalkonditionierungsschaltungen ② integriert, die ein Spannungssignal für die **Analog-to-Digital Converter** ③ erzeugen. Die in der Abbildung blau gekennzeichneten analogen Signale (Ladung/Spannung) sind nicht von außen zugänglich.

Durch einen oder mehrere **ADC** ③ werden die konditionierten Signale in die digitale Domäne überführt. Über die Art der Analog-Digital-Wandlung liegen in der Regel keine oder nur indirekte² Informationen vor. Bei den meisten Sensoren findet eine digitale Signalkonditionierung ① statt, beispielsweise Tiefpassfilterung, Dezimierung und Offset-Kompensation bei Beschleunigungssensoren. Bei schwingenden Gyroskopen (*vibrating structure gyroscope*) wird oft auch noch die Arbeitsfrequenz mit einem Bandsperrfilter unterdrückt [107]. Bei einigen Sensoren kann die digitale Signalverarbeitung deaktiviert werden, sodass direkt die unveränderten **ADC**-Werte (orange) in die Interface-Register ⑤ übertragen werden. Bei dem in dieser Arbeit verwendeten Sensor *TDK InvenSense MPU-9250* ist dies nicht möglich und nur die tiefpassgefilterten Werte (grün) können ausgelesen werden. Das Auslesen der Messwerte aus dem Sensor erfolgt über ein digitales Interface. Für einfache Sensoren sind dies oft **I²C**, **I²S**, **UART** oder **SPI**. Bei allen diesen Bussystemen gibt das Host-System den Interface-Takt vor (**Datentakt**). Dabei sind der Datentakt und der **ADC**-Takt in der Regel unabhängig. Der Daten-Takt wird vom Host nur so lange angelegt, wie tatsächlich Daten übertragen werden, um den Energieverbrauch zu reduzieren und weniger elektromagnetische Interferenzen (auch im Sensor selbst) zu verursachen. Der **ADC**/Sensortakt-Takt (in der Abbildung rot) wird daher von einem im Sensor integrierten Taktgenerator ⑦ erzeugt. Einige wenige Sensoren ermöglichen das Einspeisen eines externen Taktes als Referenz für den internen Taktgenerator. Um Zustandsänderungen im Sensor zum Host zu signalisieren, verfügen die Sensoren über eine Interrupt-Logik ⑥. In der Regel können ein oder mehrere Digitalsignale beim Eintreten bestimmter Ereignisse geschaltet werden. Durch einen **Data Ready Interrupt** signalisieren die Sensoren, dass neue Messwerte in den Registern ⑤ bereitstehen. Der Host kann die Daten anschließend über das digitale Interface auslesen. Eine Interruptsteuerung ist nötig, da Sensor- und Host-Takt asynchron sind. Ohne Interruptsignale müsste das Hostsystem die Daten aus den Registern mit einer höheren Datenrate lesen als diese aktualisiert werden und die doppelten Werte löschen, um sicher zu stellen, dass keine Messwerte verpasst werden. Sensoren für dynamische Anwendungen verfügen auf Grund der höheren Datenraten oft über einen Ringpuffer (**First In, First Out, FIFO**), um die Messwerte am Stück und nicht einzeln auszulesen und so Kommunikations-Overhead im Hostsystem zu vermeiden. Verfügen die Sensoren über ein **FIFO**-Puffer, kann in der Regel ein Interrupt beim Erreichen eines bestimmten Füllstandes (z. B. 50 %) ausgelöst werden. Anschließend liest der Host die entsprechende Anzahl von Messwerten aus dem **FIFO**. Oft ist es außerdem möglich, einen Interrupt auszulösen, wenn ein bestimmter Messwertbereich verlassen wird. Anwendungsbeispiele hierfür sind unter anderem die Freifall-Detektion bei Geräten mit Festplatten. Dies kann vom Sensor selbst detektiert werden, da im Freifall die Inertial-Beschleunigung auf fast Null sinkt. Ein anderes Beispiel ist die schnelle Detektion von Stoßbeschleunigungen zum Auslösen von Airbags [26]. Die wohl häufigste Anwendung ist die Detektion von Bildschirmdrehungen in Smartphones.

²Im Datenblatt des *AXDL355* wird für einen externen **ADC**-Takt von 1,024 MHz eine Datenrate von $\frac{1,024 \text{ MHz}}{4 \cdot 64 \cdot \text{ODR}}$ spezifiziert. Dabei ist **ODR** der Wert eines Konfigurations-Registers, der die **Output Data Rate** konfiguriert, was auf einen **Sigma-Delta-ADC** hindeutet, ohne dies je explizit zu erwähnen.

2.5.1 MEMS-Beschleunigungssensor

Zur Erklärung der Funktion und des Aufbaus von **MEMS**-Beschleunigungssensoren wird nun beispielhaft ein dreiachsiger Beschleunigungssensor betrachtet. Die Abbildung 2.6 zeigt beispielhaft Mikroskopaufnahmen eines *STMicroelectronics AIS328DQ* Sensors, um die grundsätzliche Funktionsweise dieser Art von Sensoren zu verdeutlichen.

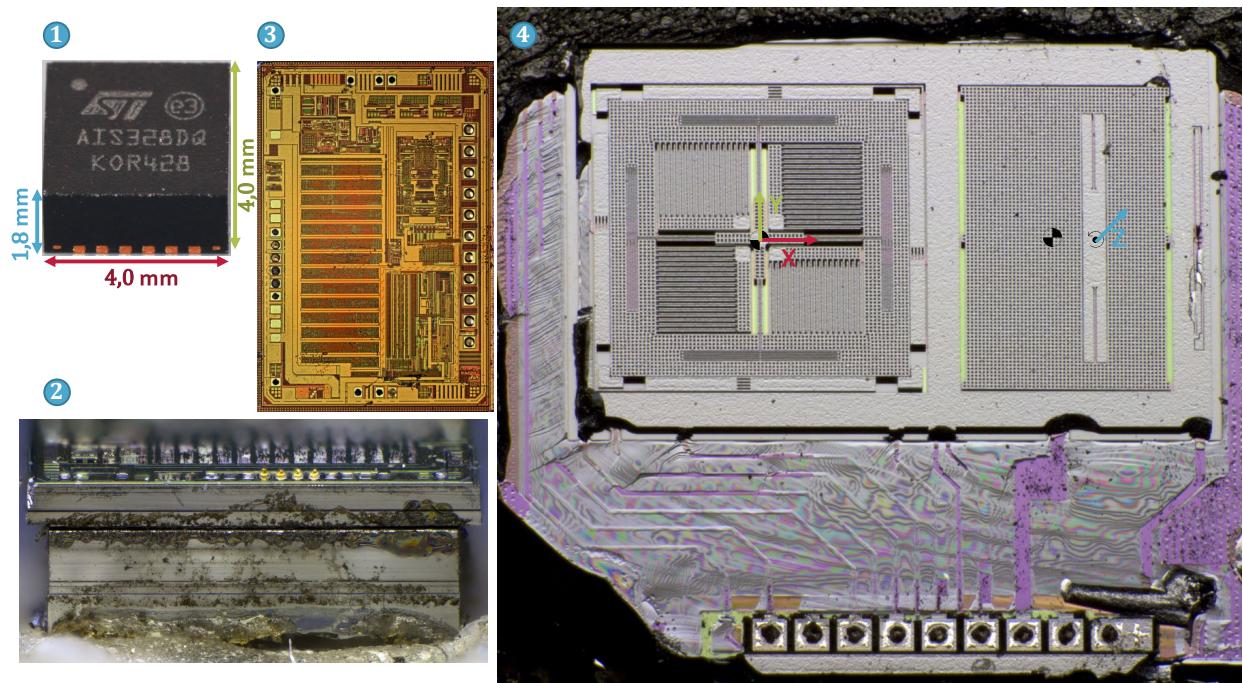


Abbildung 2.6: **Beispiel: Triaxial-MEMS-Beschleunigungssensor AIS328DQ von STMicroelectronics.** Das QFN-Gehäuse des Sensors ① hat die Maße $4\text{ mm} \cdot 4\text{ mm} \cdot 1,8\text{ mm}$. Der Sensor ② besteht aus zwei verschweißten Siliziumchips. Ein Chip ④ enthält die Sensorelemente für die X- und Y-Achse (links) und die Z-Achse (rechts). Die seismische Masse ist perforiert, um die *Squeezed-Film*-Dämpfung zu reduzieren [7]. Der zweite Chip ③ enthält die Ladungsverstärker und **ADCs** (rechts) sowie die digitalen Filter, Interruptlogik und Kommunikationsschnittstellen (links). Mit freundlicher Genehmigung von Richard Kraußler [48].

Der gesamte **MEMS**-Sensor ist in einem $4\text{ mm} \cdot 4\text{ mm} \cdot 1,8\text{ mm}$ großen **QFN**³-Gehäuse untergebracht ①. Die elektrischen Kontakte, die auch der Befestigung dienen, befinden sich auf der Unterseite des Gehäuses. In dem Kunststoffgehäuse ist der aus zwei Halbleiterstücken zusammenge-schweißte Sensor ② eingeschmolzen. Dieser Aufbau ist üblich, da so **MEMS**-Sensorelement und Ansteuerungselektronik in unterschiedlichen, jeweils auf die Aufgabe spezialisierten Halbleiter-fertigungsverfahren produziert werden können. Durch das Verschweißen wird außerdem eine gas-dichte Kammer erzeugt, in welcher sich das bewegliche Sensorelement befindet.

Die Elektronik des Sensors ist in der Abbildung durch ③ gekennzeichnet. Die Abbildung 2.7 zeigt den aus dem Datenblatt entnommenen schematischen Aufbau dieses Sensors.

³QFN steht für **Quad-Flat No-Leads**, also ein oberflächenmontiertes Gehäuse ohne Anschlusspins.

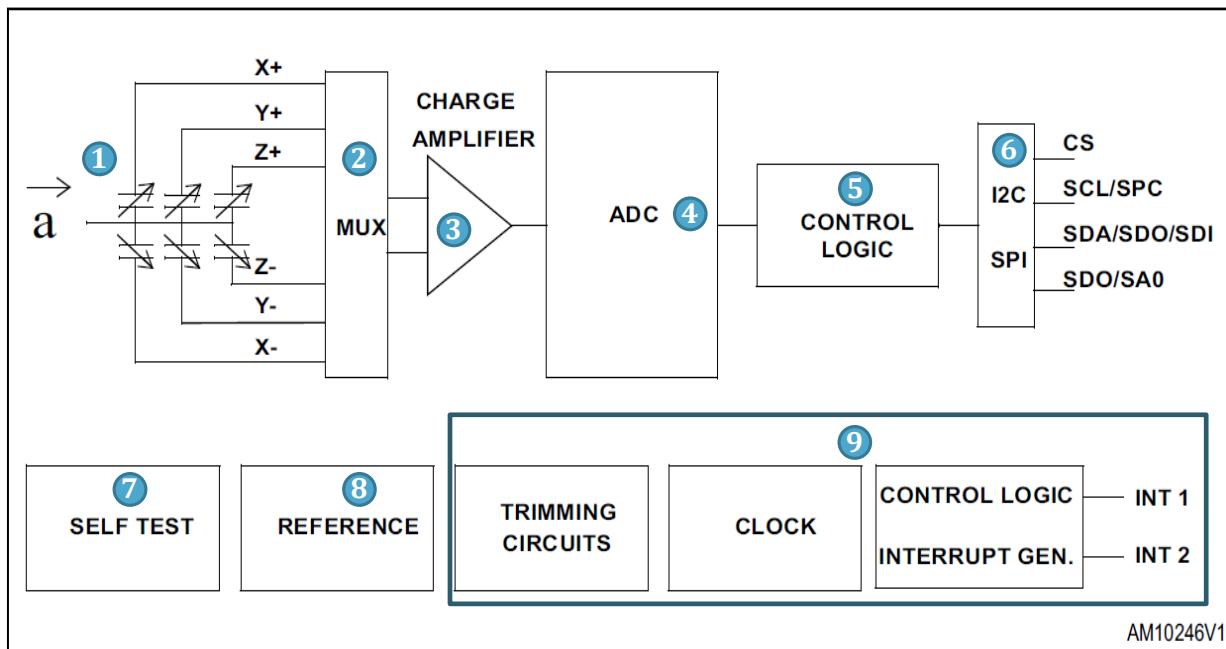


Abbildung 2.7: **Blockdiagramm eines triaxialen MEMS-Beschleunigungsaufnehmers AIS328DQ von STMicroelectronics.** Das MEMS-Sensorelement ① arbeitet kapazitiv. Ein Multiplexer ② schaltet die Verbindung zum Ladungsverstärker ③, welcher ein Spannungssignal für den ADC ④ erzeugt. Die Kontrolllogik ⑤ steuert die digitalen Filter und bedient die Kommunikationsschnittstelle ⑥. Interne Spannungsreferenzen ⑧ und ein Oszillatator ⑨ erzeugen Referenzspannung und Takt für den ADC. Die Selbsttest-Funktionalität kann steuerbar eine konstante Auslenkung der seismischen Masse erzeugen. Verändert aus [96].

Die Differentialkondensator-Signale des Sensorelements werden über einen Multiplexer ② in einen Ladungsverstärker ③ geführt. Die von diesem ausgegebene Spannung wird von einem ADC ④ digitalisiert. In anderen Sensortypen ist ggf. für jede Sensorachse ein eigener Verstärker und ADC verbaut (siehe Abbildung 6.1). Dies führt jedoch zu größerer Chipfläche (größere Kosten) und auch erhöhtem Stromverbrauch. Die Empfindlichkeit des Ladungsverstärkers kann üblicherweise stufenweise konfiguriert werden. Die Steuerung der im Sensor ablaufenden Prozesse wird bei einfachen Sensoren durch eine Kontrolllogik übernommen, welche als Zustandsautomat agiert. Komplexe Sensoren beinhalten ein Mikroprozessorsystem (beispielsweise *TDK Invensense ICM-20948-P*) mit einem integrierten *Digital Motion Processor*TM mit RAM und ROM, welches die Signalerfassung steuert und komplexe Berechnungen, wie z. B. Positionsschätzungen zur Trägheitsnavigation, durchführt.

Die Kommunikation mit der Außenwelt erfolgt über digitale Schnittstellen 6 (z. B. I²C, SPI oder UART). Die Kommunikationsprotokolle sind im Gegensatz zu den Schnittstellen nicht herstellerübergreifend standardisiert. Bei einfachen Sensoren finden Kommunikation und Konfiguration üblicherweise über Registeradressen statt. Bei jedem Lese- oder Schreibvorgang wird eine Adresse an den Sensor gesendet. Der Sensor schreibt entweder die Daten in das adressierte Register oder liest die Werte aus. Aufeinanderfolgende Register können üblicherweise auch konsekutiv gelesen werden, ohne dass eine neue Adresse übergeben werden muss. Einige Sensoren enthalten auch einen FIFO-Speicher für einige hundert Messwerte. Diese können effizient blockweise mittels Di-

rect Memory Access (DMA) ausgelesen werden.

Der Sensor erzeugt üblicherweise den Abtasttakt sowie alle internen Takte selbst aus einer integrierten Zeitbasis 9. Durch die Interruptlogik des Sensors kann in der Regel ein Data Ready Interrupt (DRI) generiert werden, welcher die vollständige Erfassung eines Messdatensatzes signalisiert. Die Referenzspannungen für den ADC werden ebenso wie der Takt intern erzeugt 8.

Zur Funktionsüberprüfung der Sensorelemente haben die Sensoren oft eine Selbsttestschaltung integriert, die eine elektrostatische Auslenkung der Sensorelemente durch Anlegen einer elektrischen Spannung erzeugt 7.

2.5.2 Modell des Sensorelements

Sowohl das Messprinzip analoger piezoelektrischen Beschleunigungssensoren als auch das von MEMS-Beschleunigungssensoren basiert auf der Messung der Auslenkung einer seismischen Masse m , welche über eine Feder mit der Steifigkeit k mit dem Ort der Krafteinwirkung verbunden ist. Nach dem Newtonschen Trägheitsgesetz erfährt ein Körper mit der Masse m , der in einem Inertialsystem mit der Beschleunigung a beschleunigt wird, eine Kraft F in entgegengesetzter Richtung.

$$F(t) = m \cdot a(t) \quad (2.42)$$

In dem Modell des einfachen Feder-Masse-Systems (siehe Abbildung 2.8), mit der Federkonstante k kann die Auslenkung $x(t)$ der Masse m entlang der X-Achse als Folge der Beschleunigung $a_x(t)$ mit folgender Differentialgleichung beschrieben werden.

$$m \frac{d^2x(t)}{dt^2} + kx(t) = ma_x(t) \quad (2.43)$$

Durch den piezoelektrischen Effekt kommt es in Piezo-Aufnehmern bei einer elastischen Längenänderung des Piezoelements als Folge der einwirkenden Kraft zu einer Ladungstrennung im Piezokristall. Die Ladungsänderung ist dabei proportional zur Längenänderung. In einem MEMS-Beschleunigungssensor wird die Auslenkung in der Regel über einen kammförmigen Differentialkondensator kapazitiv erfasst.

Die Kapazität C der Kondensatoren an der Position x kann für Auslenkungen, die klein gegen den Plattenabstand sind ($x \ll d$), durch folgende Gleichungen beschrieben werden [9].

$$C_{\text{top}} = \frac{\varepsilon \cdot N_f \cdot L_f \cdot h}{(d + x)} = \frac{\varepsilon \cdot N_f \cdot L_f \cdot h}{d \cdot (1 + x/d)} \approx \frac{\varepsilon \cdot N_f \cdot L_f \cdot h}{d} \cdot \left(1 - \frac{x}{d}\right) \quad (2.44)$$

$$C_{\text{bottom}} = \frac{\varepsilon \cdot N_f \cdot L_f \cdot h}{(d - x)} = \frac{\varepsilon \cdot N_f \cdot L_f \cdot h}{d \cdot (1 - x/d)} \approx \frac{\varepsilon \cdot N_f \cdot L_f \cdot h}{d} \cdot \left(1 + \frac{x}{d}\right) \quad (2.45)$$

Dabei ist L_f die Länge eines Kammfingers, h die Strukturhöhe, N_f die Anzahl der Finger pro Kamm (in Abb. 2.6 ist $N_f = 17$), ε die Permittivität des Mediums im Sensor und d der Abstand der Kondensatorfinger in Ruheposition. Für kleine Auslenkungen gilt dann folgende Näherung.

$$\Delta C = C_{\text{bottom}} - C_{\text{top}} = \frac{2 \cdot \varepsilon \cdot N_f \cdot L_f \cdot h}{d} \cdot \left(\frac{x}{d}\right) = 2 \cdot C \cdot \left(\frac{x}{d}\right) \quad (2.46)$$

Die Kapazitätsdifferenz ΔC zwischen den Differential-Kondensatoren wird nun in eine Spannung

konvertiert (siehe Abb. 2.7) und durch einen ADC digitalisiert. In dem Blockschaltbild Abb. 2.7 ist für diese Konversion ein Ladungsverstärker eingezeichnet; dabei bleibt die genaue Funktionsweise dieser Komponente ungeklärt. In vielen Sensoren erfolgt die Auswertung der Kapazitätsdifferenz durch Wechselstrommessbrücken in Kombination mit einem Lock-in-Verstärker [79]. In einigen MEMS-Sensoren werden auch direkt Charge-to-Digital Converter (CDC) zur Digitalisierung der Kapazitätsdifferenz verwendet [70].

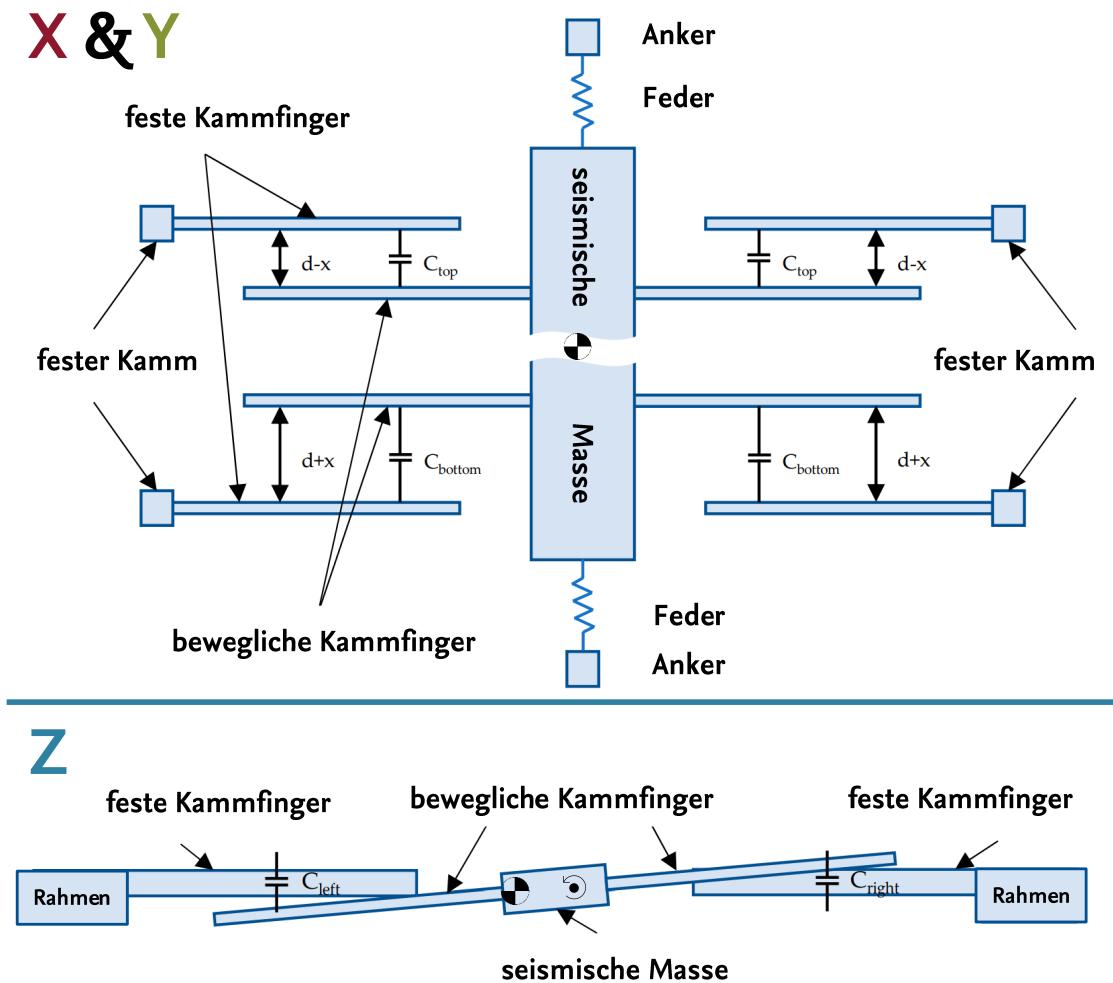


Abbildung 2.8: **Funktionsprinzip eines kapazitiven MEMS-Beschleunigungsaufnehmers.** Wirkt eine Beschleunigung auf die seismische Masse, führt die resultierende Kraft $F = m \cdot a$ zu einer Verschiebung der elastisch gelagerten seismischen Masse. In der XY-Ebene kann diese Verschiebung durch die Differentialkondensatoren direkt kapazitiv detektiert werden. Für die senkrecht zum Substrat stehende Z-Achse können keine Kammfinger gefertigt werden. Daher wird hier die Verkippung der drehbar gelagerten Masse kapazitiv detektiert. Der Drehpunkt ↗ befindet sich außerhalb des Massenschwerpunkts ⚡, sodass eine Verkippung zu unterschiedlichen Kapazitäten auf zu- und abgewandter Seite führt. In Abb. 2.6 ist ein geöffneter Sensor mit ähnlichem Aufbau zu sehen. Grafik entnommen aus [99].

Die Sensoren sind in der Regel mit einem Inertgas (z. B. Stickstoff oder Argon) gefüllt [11, 33]. Die höchste mechanische Güte und somit auch Empfindlichkeit wird im Vakuum erreicht. Durch die mikroskopischen Dimensionen der Aufnehmer liegt das Gewicht der seismischen Massen im Bereich einiger Milligramm [11]. Daher führen Stöße mit dem umgebenden Gas zu geringfügigen Positionsänderungen der Masse ohne von außen wirkender Kraft. Dieses ist die dominierende Rauschquelle bei gasgefüllten Sensoren [70].

Soll das Sensorelement zur Reduzierung des Eigenrauschen im Vakuum arbeiten, ist ein hermetisch abgeschlossenes Gehäuse nötig (z. B. ein CLCC), um das Eindringen von Gas und damit eine Veränderung der Sensoreigenschaften zu verhindern [35].

Die Abbildung 2.6 zeigt in ④ ein Mikroskopbild des Sensorelements des dreiachsigen **MEMS**-Beschleunigungssensors vom Typ AIS328DQ des Herstellers STMicroelectronics als Beispiel. Dieser Sensor ist mit Inertgas gefüllt. Dabei ist im linken Bildteil das Sensorelement für die innerhalb des Sensorsubstrates liegenden X- und Y-Achsen erkennbar. Das Sensorelement für die Z-Achsen-Beschleunigung befindet sich rechts im Bild. Die seismische Masse des Z-Achsen-Sensorelements ist drehbar mit einer Federachse (↷) gelagert, welche sich außerhalb des Schwerpunktes (◐) befindet. Eine durch Beschleunigung verursachte Kraft führt folglich zu einer Drehung des Sensorelements um seine Federachse. Dadurch kommt es zu einer Abstandsänderung zum Sensorgehäuse, welche kapazitiv ausgewertet wird.

Es ist zu erkennen, dass alle seismischen Massen des Sensors mit einer Gitterstruktur versehen sind. Diese ermöglicht es dem Inertgas, das sich im Sensorgehäuse befindet, durch die Strukturen zuströmen. Dennoch kommt es gerade bei der Z-Achse zur sogenannten *Squeezed-Film*-Dämpfung [7]. Diese Dämpfung wird durch die erzwungene Strömung verursacht und führt zu einer Reduzierung der Empfindlichkeit des Sensors sowie zu höherem Rauschen durch turbulente Anteile der Strömung. Das Feder-Masse-System der **MEMS**-Sensoren zeigt Resonanzen, die ohne Dämpfung (hohe mechanische Güte) zu einem starken Aufschwingen der Sensorelemente und somit zu Störungen der Signalerfassung oder Zerstörung des Sensors führen können [6]. Über den Druck des Inertgases und die Geometrie und Anordnung der Löcher in den Strukturen lässt sich die Dämpfung des Feder-Masse-Systems so einstellen, dass eine kritische Dämpfung des mechanischen Systems erreicht wird. Dies macht die Sensoren robust gegen Stoßbeschleunigungen und parasitäre Vibrationen, erhöht aber das Eigenrauschen.

2.6 Abtastphasenrauschen und Absolutzeitstempelung

Wie schon im vorigen Abschnitt 2.5 erwähnt, erzeugen die **MEMS**-Sensoren ihren Abtasttakt für die Datenerfassung in der Regel selbst. Die nominelle Datenrate der Sensoren, $F_{s\text{Nom}}$ oder auch **Output Data Rate (ODR)** genannt kann dabei meist mittels Konfigurationsregister im Sensor programmiert werden. Je nach Sensor führt eine Änderung der Datenrate zu veränderter digitaler Filterung. Bei einigen Sensortypen werden bei niedrigerer Datenrate mehr Messwerte gemittelt. Diese Mittlung vermindert das Rauschen durch die Reduktion der Messbandbreite.

Eine andere Variante zur Reduktion der Datenrate ist die Vergrößerung der Zeitabstände zwischen den einzelnen Messwerterfassungen, was lediglich zu einem reduzierten Stromverbrauch ohne Verringerung des Rauschens führt. Die digitalen **MEMS**-Sensoren signalisieren in der Regel durch einen **Data Ready Interrupt (DRI)** die abgeschlossene Erfassung der Messwerte.

Wegen der internen Erzeugung des Abtasttaktes, engl. *Sampling Clock Frequency f_s* , kommt es zu Abweichungen der tatsächlichen Abtastzeitpunkte vom erwarteten ideal äquidistanten Verlauf (**Ideal Time Course, ITC**). Diese Abweichung wird auch als **Time Interval Error (TIE)** bezeichnet [76]. Durch die Absolutzeitstempelung der **DRI** können für die Sensoren Informationen über den vermutlichen Abtastzeitpunkt gewonnen werden. Diese Zeitstempel werden für die Sinus-Approximationen verwendet, um die Einflüsse durch das Abtastphasenrauschen zu minimieren. In Kapitel 4 werden die Effekte von nichtäquidistantem Abtasten erklärt und für einige Sensoren charakterisiert. Außerdem werden die daraus resultierenden Amplituden- und Phasenabweichungen simuliert.

3 Digitale Datenerfassungseinheit

Zur Erfassung, Verarbeitung und Zeitstempelung der digitalen Sensordaten wurde eine Datenerfassungseinheit entwickelt, die als **Data Acquisition Unit (DAU)** bezeichnet wird. Die Grundkomponenten der Datenerfassungseinheit sind der Mikrocontroller **STM32F767ZI** in Kombination mit einem **GNSS**-Empfänger vom Typ **U-Blox NEO N7**. Die Verwendung des Echtzeitbetriebssystems **FreeRTOS** ermöglicht die asynchrone Verarbeitung der Sensordaten. Die **DAU** konvertiert die von den digitalen Sensoren erfassten Messwerte in **SI**-Einheiten und erfasst die Absolutzeitstempel der Messwerte. Über eine **Ethernet**-Schnittstelle werden die Messdaten als kontinuierlicher Datenstrom mit Metadaten zu einem Empfangs-PC gesendet.

Durch die Verwendung standardisierter **Internet Protocol (IP)** Technik ist die Integration der **DAU** in andere Messsysteme einfach möglich, wie z. B. in das *Machine Learning Testbed* des **ZeMA** in Saarbrücken [28] oder in die Radialschmiede des *Advanced Forging Research Centre* der University of Strathclyde. Aktuell kann die **DAU** Sensoren mit **SPI** oder **I²C** Interface ansteuern. Für Messaufnehmer, die mittels **CAN**, **I²S** oder **USART** kommunizieren, ist bei Bedarf eine einfache Anpassung der **DAU** möglich, da der Mikrocontroller diese Schnittstellen in der Hardware unterstützt. Je nach Hersteller und Sensormodell erfolgt das Auslesen der Daten und die Sensorkonfiguration unterschiedlich. Es existiert kein gemeinsamer Standard. In der Regel werden die Sensoren über das Setzen einzelner Bits in den Konfigurationsregistern programmiert.

In der **DAU**-Software wird ein gemeinsames C++ **Application Programming Interface (API)** für alle Sensoren verwendet, welches sich nur im Konstruktor unterscheidet (siehe Abschnitt 3.1.3). Dadurch können Treiber für neue Sensoren leicht in das System integriert werden. Abbildung 3.1 zeigt den Datenfluss bei der Erfassung digitaler Sensordaten mittels **DAU**.

3.1 Hardware

Das Herzstück der **DAU** ist ein Mikrocontroller vom Typ **STM32F767ZI** mit einer **ARM Cortex-M7** **CPU**-Architektur auf einem **NUCLEO-F767ZI** Evaluationsboard des Herstellers. Der **CPU**-Takt kann bis zu 216 MHz gewählt werden. Der Controller verfügt über 512 **KiB**¹ **SRAM**, 2 **MiB** **Flash-ROM** sowie 4 **KiB** **Backup-SRAM** mit Batteriepuffer. Es sind eine große Auswahl an Peripheriegeräten, unter anderem mehrere Schnittstellen für die Standards **SPI**, **USART**, **CAN**, **I²C**, **I²S** sowie ein **Ethernet-MAC** in den Controller integriert. Zur Erfassung analoger Messgrößen stehen drei **Successive-Approximation-ADCs** mit bis zu 12 **Least Significant Bit** Auflösung und einer maximalen Konversionsrate von bis zu 2,4 MS/s bei einer minimalen Abtastzeit der Abtast- und Haltestufe von 84 ns zur Verfügung.

¹Bei der Angabe von Speichergrößen können auch Vielfache von $2^{10}=1024$ verwendet werden. **Kibibyte** 1 **KiB**= $2^{10}=1024$ **Byte**; **Mebibyte** 1 **MiB**= $2^{20}=1048576$ **Byte**

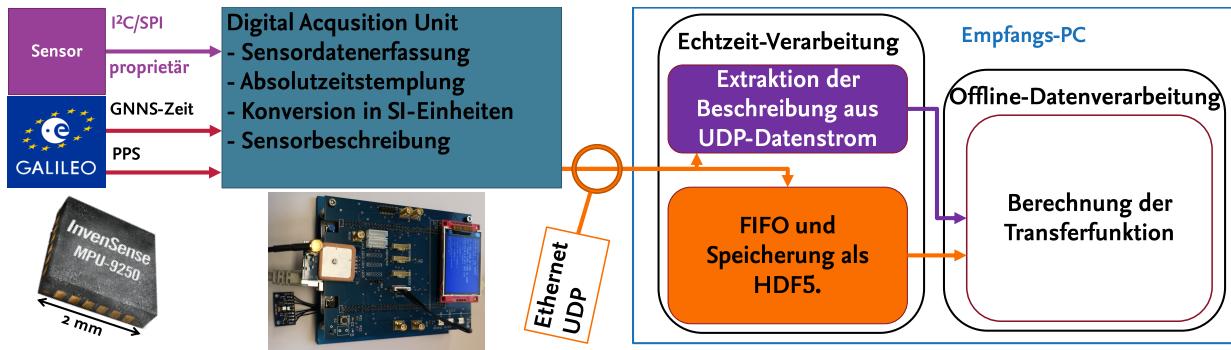


Abbildung 3.1: **Datenerfassung mit der DAU.** Der Prozessor der **Data Acquisition Unit (DAU)** liest und absolviert die Sensordaten und konvertiert diese in **SI-Einheiten**. Die Daten und Beschreibungen werden über eine **Ethernet**-Schnittstelle mittels **UDP**-Protokoll an einen Empfangs-PC versendet. Dort werden die Datenströme entgegengenommen und verarbeitet, z. B. Speicherung als **HDF**-Datei.

Zur Signalkonditionierung der in den analogen Kalibriersystemen (**ACS**) üblichen ± 10 V besitzt die **DAU** eine aktive differentielle Eingangsstufe (siehe auch 3.3.3).

Für Zeitstempelung, Frequenzmessungen, Erzeugung von **PWM**-Signalen, Einlesen von Rotationsencodern und Erzeugen von Interrupts oder Hardware-Triggern stehen zwei 32-Bit und fünfzehn 16-Bit-Timer/Counter zur Verfügung. Die maximale Zeitbasisfrequenz der 32-Bit-Zähler beträgt 108 MHz, die der 16-Bit-Zähler 216 MHz. Zwei **General Purpose DMA**-Controller ermöglichen Datenübertragung zwischen Peripheriegeräten und **SRAM** ohne Beteiligung der **CPU**. Der **Ethernet-MAC** verfügt über einen eigenen **DMA**-Controller.

Zur Steuerung des Programmflusses innerhalb verschachtelter **Interrupt Request (IRQ)** besitzt der **ARM-Cortex M7** Mikrocontroller einen **Nested Vectored Interrupt Controller (NVIC)** mit bis zu 240 individuell priorisierbaren Interrupt-Kanälen.

Zur Synchronisierung in verteilten Systemen und zur Kompensation der internen Zeitbasis wird ein **Global Navigation Satellite System (GNSS)**-Modul der Firma **U-Blox** des Typs **NEO-M7** als **Pulse-per-Second (PPS)** Zeitquelle verwendet.

Die Stromversorgung der **DUT**-Sensor-**Breakout Boards (BoB)** erfolgt wahlweise durch die 5 V oder 3,3 V Spannungen des **NUCLEO-F767ZI**-Boards. Die in dieser Arbeit kalibrierten Sensoren wurden mit 5 V gespeist. Die 3,3 V Versorgungsspannung der Sensoren wird direkt auf den **BoBs** mittels **Linearregler (LDO)** erzeugt; so wird der Einfluss der Stromversorgungsqualität auf die Sensoren minimiert.

Die Anzeige von Status-Informationen, z. B. der **GNSS**-Zeit oder der automatisch mittels **DHCP** bezogenen **IP**-Adressen, wird durch ein über **SPI** angebundenes 320×240 Pixel großes Farb-**LCD** realisiert.

Zum Aufspielen und Debuggen der Mikrocontroller-Firmware kann der integrierte **ST-Link-v2-Debugger** des **NUCLEO-F767ZI**-Evaluationsboards verwendet werden, hierbei ist das Aktivieren von bis zu fünf Hardware-Breakpoints möglich. Für komplexere Debugging-Aufgaben besitzt die **DAU** noch einen zwanzigpoligen **Debugging-Connector**, der sowohl das **SWO** als auch das **JTAG**-Trace Debugginginterface der **STM32F767ZI-CPU** zur Verfügung stellt.

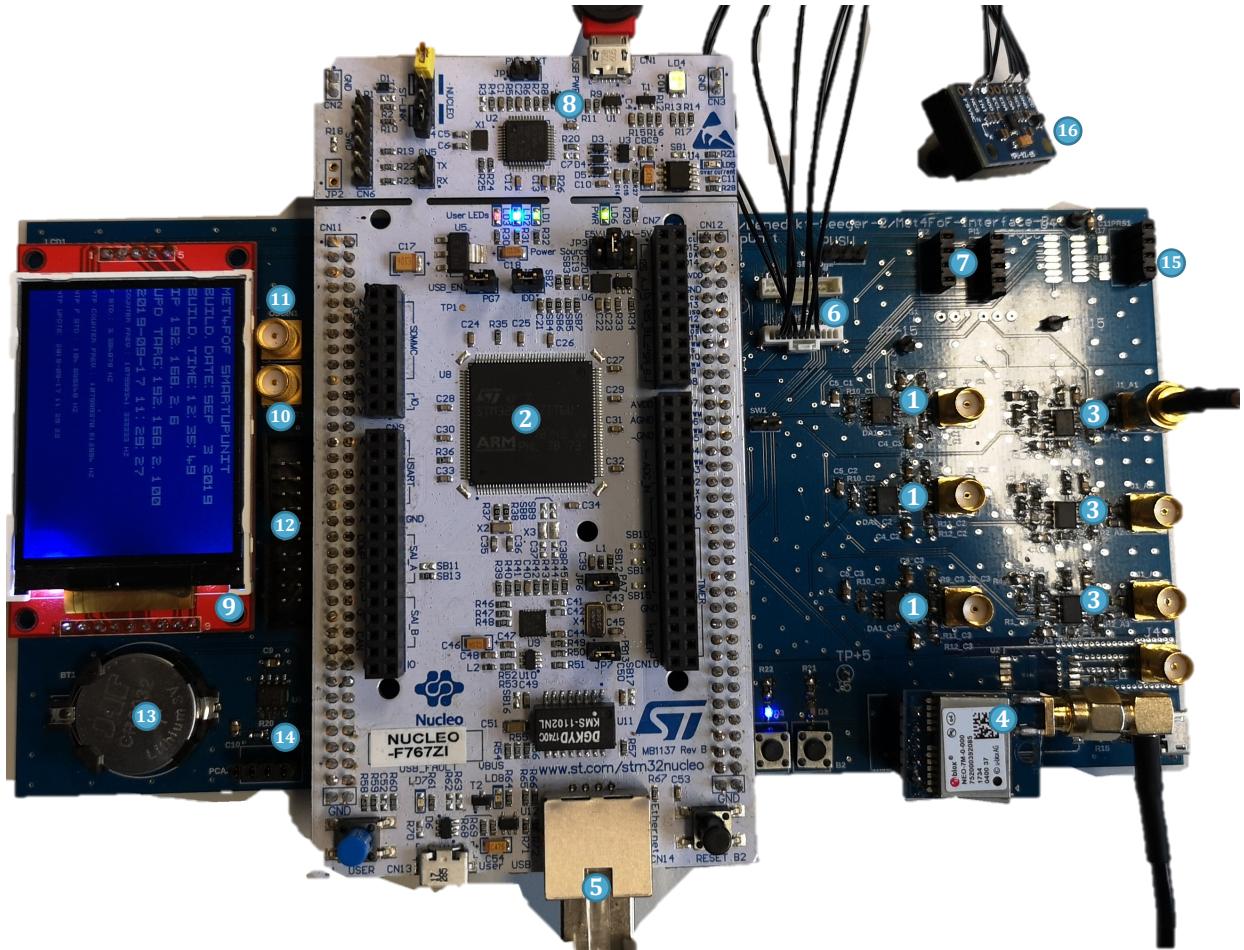


Abbildung 3.2: **Data Acquisition Unit mit angeschlossenem Sensor.** ① Komparator-Eingänge, ② STM3276ZI Mikrocontroller, ③ ADC-Eingang, ④ GNSS-Empfänger, ⑤ Ethernet-Anschluss, ⑥ SPI-Sensoranschlüsse, ⑦ I²C-Sensoranschlüsse, ⑧ ST-Link-v2-Debugger, ⑨ Info-Display, ⑩ Takteingang, ⑪ Taktausgang, ⑫ JTAG/SWO-Debugging-Anschluss, ⑬ Backup-SRAM-Batterie, ⑭ CAN-Anschluss, ⑮ RS422-USART, ⑯ Sensor-BoB.

Abbildung 3.2 zeigt die für die Kalibrierungen verwendete Version der DAU-Hardware mit analoger Signalkonditionierung und drei Differential-Komparator-Eingängen für 10 V (peak to peak) zur Zeitstempelung externer Signale. Die Abbildung 3.3 zeigt die rein digitale Version der DAU. Das NUCLEO-F767ZI-Evaluationsboard befindet sich unter der DAU-Platine. Die ADC-Eingänge sind direkt über SMA-Steckverbinder zugänglich. Eine externe Signalkonditionierung auf den Spannungsbereich von 0 V bis 3,3 V, bezogen auf das GND-Potential der DAU, ist zwingend nötig, da anderenfalls der Mikrocontroller beschädigt werden kann. Es gibt vier parallel verwendbare digitale SPI-Sensorinterfaces sowie bis zu fünf I²C-Anschlüsse ohne Hardware-Absolutzeitstempelung und DRI-IRQ-Eingang. Diese Version der DAU wurde am ZeMA und der Radialschmiede des Advanced Forging Research Centre der University of Strathclyde in Kombination mit verschiedenen Sensoren verwendet.

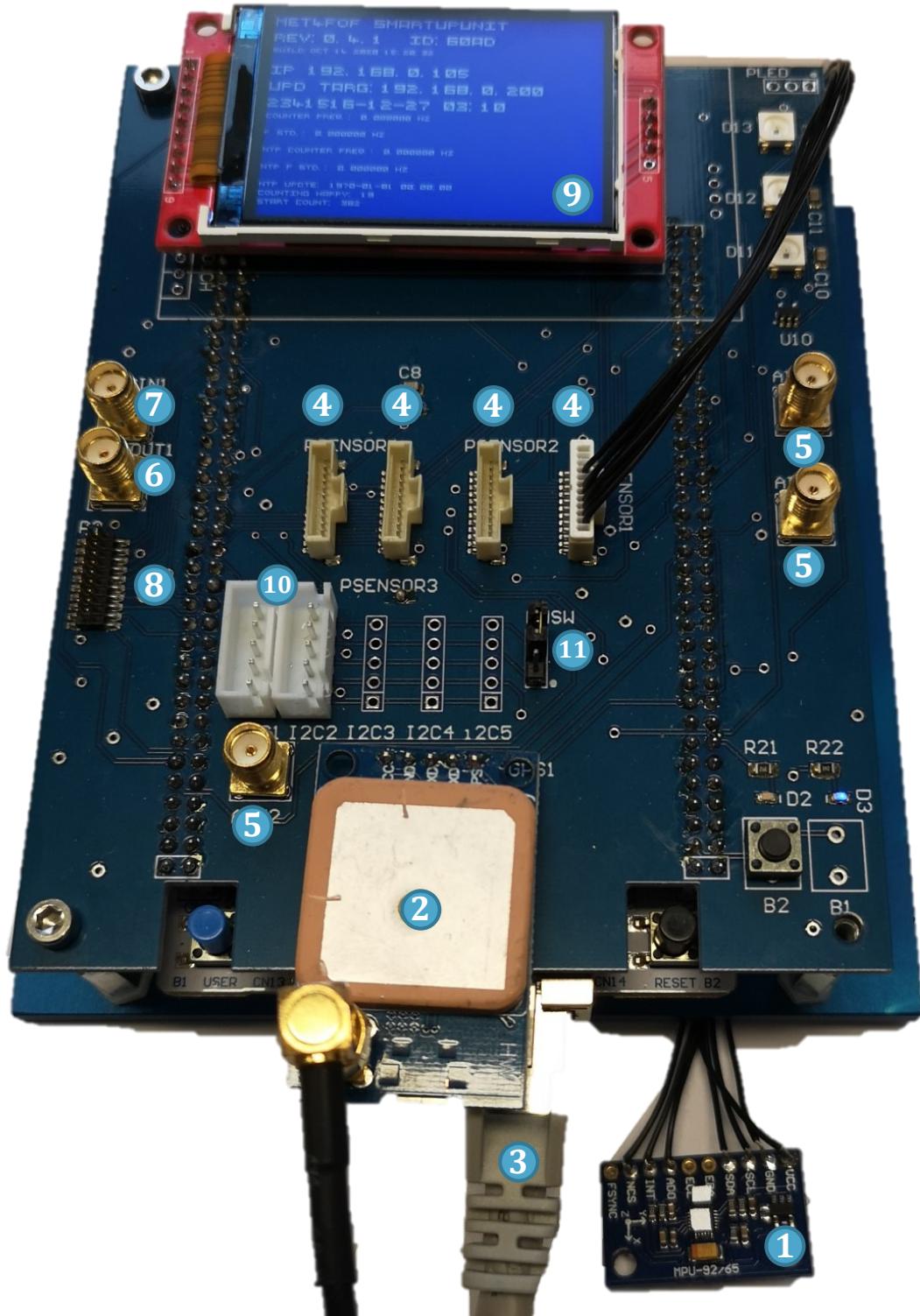


Abbildung 3.3: **Data Acquisition Unit ohne analoge Eingangsschaltungen.** ① MPU-9250-Sensor, ② GNSS-Empfänger, ③ Ethernet-Anschluss, ④ SPI-Sensoranschlüsse, ⑤ ADC-Anschluss, ⑥ Takteingang, ⑦ Taktausgang, ⑧ JTAG/SWO-Debugging-Anschluss, ⑨ Infodisplay, ⑩ I²C-Sensoranschlüsse, ⑪ Sensorspannungsumschalter.

3.1.1 Embedded-Software

Die DAU muss verschiedene zeitkritische Aufgaben, wie z. B. Sensordaten lesen, GNSS-Zeitsynchronisierung oder Ethernet-Datenübertragung, parallel abarbeiten können. Echtzeit-Betriebssysteme mit präemptivem Multitasking, so wie das in der DAU verwendete FreeRTOS, sind für solche Anwendungsfälle entwickelt worden. Beim präemptiven Multitasking wird jeder Prozess präventiv von einem Timer unterbrochen, damit alle Prozesse ausgeführt werden, auch wenn sie gleiche Priorität haben. FreeRTOS ermöglicht die parallele Ausführung von Prozessen mit unterschiedlicher Priorität. Jeder aktuell ausgeführte Prozess wird periodisch (alle 1 ms) vom *Scheduler* unterbrochen und die Register- und *Stack*-Daten werden gesichert. Diese Unterbrechung wird *Kontext-Switch* genannt. Der *Scheduler* prüft nun, ob ein Prozess mit höherer Priorität CPU-Zeit benötigt, weil z. B. ein *Timeout* abgelaufen oder ein Paket in einer *FreeRTOS-Mail-Queue* eingetroffen ist. Sollte dies der Fall sein, wird der höher priorisierte Task ausgeführt, anderenfalls wird der zuvor unterbrochene Task fortgesetzt.

Die CPU der DAU besitzt einen Nested Vectored Interrupt Controller (NVIC), der die konfigurierbare Priorisierung von IRQs ermöglicht. Der Begriff *Nested*, auf Deutsch verschachtelt, bedeutet, dass auch Unterbrechungen von ISRs durch höher priorisierte ISRs möglich sind. Der dabei ausgelöste *Kontext-Switch* benötigt 12 Instruktionsszyklen, d. h. bei maximaler CPU-Frequenz ≈ 56 ns.

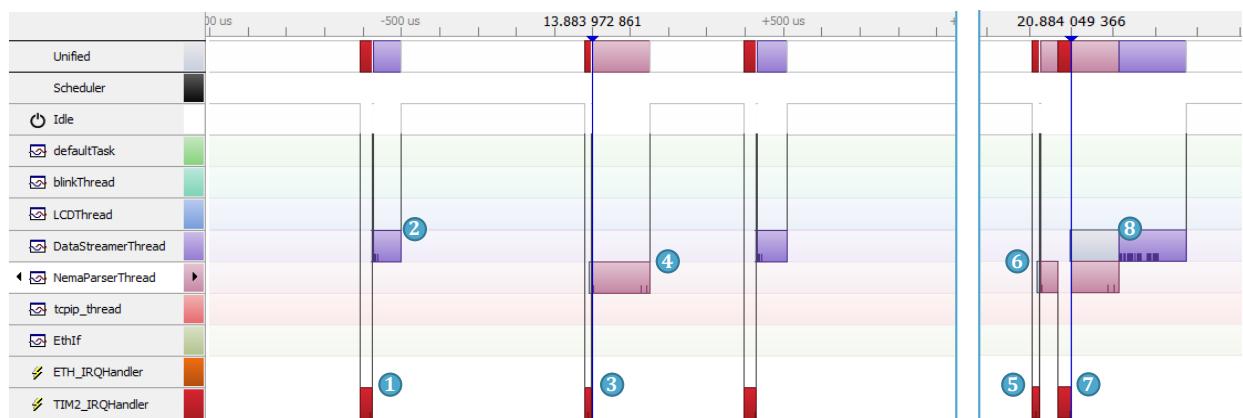


Abbildung 3.4: **Zeitverlauf der DAU-Prozesse nach der Sensor-IRQ.** Links: ① Die steigende DRI-Flanke des Sensors löst eine Timer-CCIRQ aus, die den Sprung in die *Timer-ISR* auslöst, nun wird der entsprechende Sensor ausgelesen. ② Der DataStreamer-Prozess liest anschließend die Daten aus der *Queue* und verarbeitet sie. ③ Die *Timer-ISR* wird angesprungen, eine *NMEA*-Nachricht wird aus dem *USART-DMA*-Puffer gelesen. ④ Der *NMEAParser*-Prozess liest die Daten aus der *Queue* und aktualisiert die *GNSS*-Informationen.
Rechts: ⑤ Die *Timer-ISR* des *GNSS*-Empfängers wird ausgelöst. Anschließend wird ⑥ der *NMEAParser*-Prozess ausgeführt und eine Sensor-ISR ⑦ unterbricht den *NMEAParser*-Prozess. Nach der Unterbrechung wird der *NMEAParser*-Prozess zu Ende ausgeführt und danach wird der DataStreamer-Prozess ⑧ angesprungen.

Die Abbildung 3.4 zeigt die mit dem *Realtime-Recording-Tool SEGGER SystemView* aufgezeichneten Prozess- und ISR-Laufzeiten. Im linken Teil des Bildes ist der normale Programmablauf ohne verschachtelte Unterbrechungen zu sehen. Der rechte Teil des Bildes zeigt eine Unterbrechung der

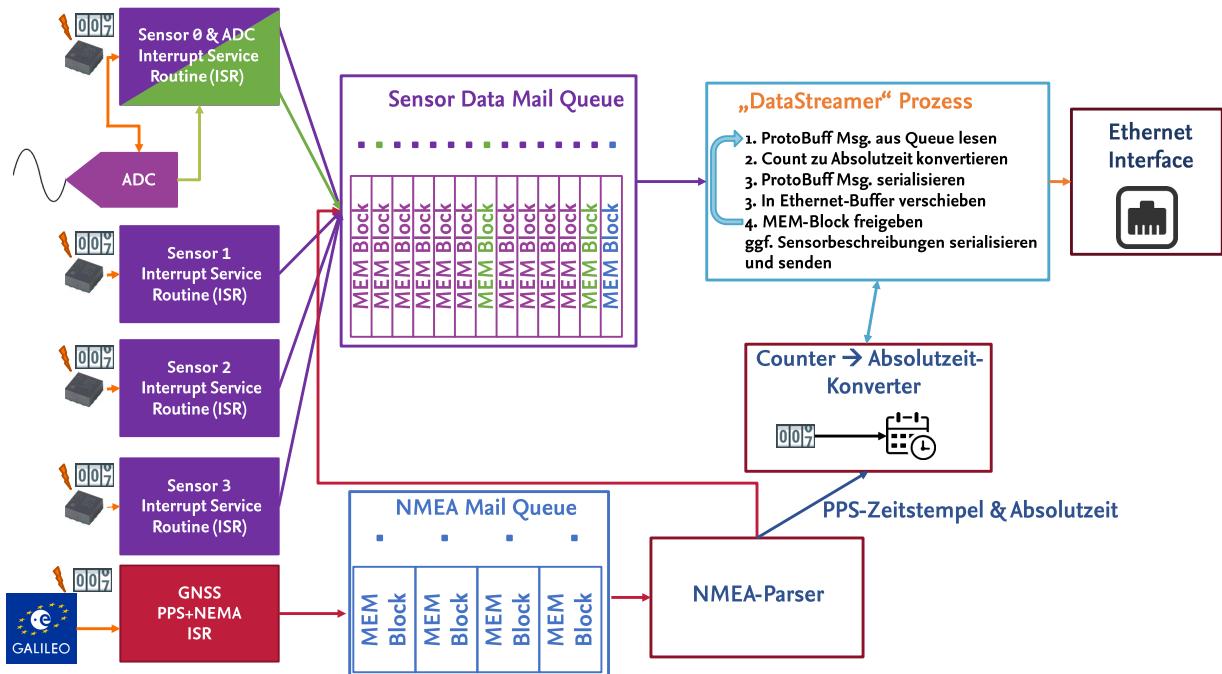


Abbildung 3.5: **Datenfluss innerhalb der DAU.** Jede DAU kann bis zu vier Sensoren ansteuern. Die DRI-Flanken der Sensoren werden individuell durch **Input Capture Counters** zeitgestempelt. Die Messdaten werden sofort innerhalb einer **ISR** vom Sensor gelesen und in **SI-Einheiten** konvertiert. Einer der Sensoranschlüsse startet über seinen **DRI** die Abtastung der drei **ADCs**. Die ausgelesenen Daten werden in einer für alle Sensoren identischen **FreeRTOS-Mail-Queue** zwischengespeichert. Die **NMEA**-Nachrichten des **GNSS**-Moduls werden zusammen mit ihren **ICC**-Zeitstempeln in einer eigenen **FreeRTOS-Mail-Queue** abgelegt. Aus dieser **Queue** liest der **NMEA-Parser-Prozess** die **GNSS**-Daten und aktualisiert den **Counter->Absolutzeit-Konverter**. Die in der **NMEA**-Nachricht enthaltenen Informationen werden als **Protobuf**-Nachrichten in die **Sensor-FreeRTOS-Mail-Queue** geschrieben. Der **DataStreamer**-Prozess liest die Sensor-Nachrichten aus der **FreeRTOS-Mail-Queue**, konvertiert die **ICC**-Werte in Absolutzeitstempel und serialisiert die **Protobuf**-Nachrichten im Arbeitsspeicher. Aus Speicher werden die Daten per **DMA** in den **Ethernet-Media-Access-Controller (MAC)** übertragen und anschließend von **MAC** an den **PC** übertragen. Außerdem versendet dieser Prozess alle zwei Sekunden die **Description**-Nachrichten der Sensoren.

NMEA-Datenverarbeitung durch eine Sensor-**ISR**. Nach der **ISR** wird die **NMEA**-Datenverarbeitung, also die Verarbeitung der **GNSS**-Zeit und Positionsdaten, fortgesetzt, da dieser Prozess eine höhere Priorität hat. Im **Scheduler** ist der **DataStreamerThread** als *pending*, also zur Ausführung anstehend, markiert und wird nach Ende der **NMEA**-Datenverarbeitung sofort ausgeführt.

Abbildung 3.5 zeigt schematisch den Datenfluss in der DAU. Sowohl die Sensordatenerfassung als auch die **GNSS/PPS**-Datenerfassung erfolgen interruptgesteuert. In den **ISR** werden ausschließlich die Rohzeitstempel und Messdaten gelesen und die Konversion in 32-Bit **float-SI**-Werte durchgeführt. In der **ISR** des Sensors 0 werden zusätzlich noch die durch die externe **IRQ**-Flanke getriggerten **ADC**-Werte gelesen. Die Messdaten werden als **DataMessage-Struct** in den Memory-Pool

der *FreeRTOS-Mail-Queue* geschrieben (osMailPut). Die **NMEA**-Rohdaten (ASCII-String) werden zusammen mit ihrem Zeitstempel in einer eigenen *FreeRTOS-Mail-Queue* gespeichert. Der **NMEAParser**-Prozess extrahiert die **GNSS**-Zeit aus den **ASCII**-Daten und aktualisiert mit dem Tupel (Zeitstempel, **Unixzeit**) den *Counter-Absolutzeit-Konverter* (siehe hierzu Abschnitt 3.3). Die vom **GNSS**-Modul gelieferten Informationen über Position, Höhe, Satelliten in Sicht, die theoretische Positionsunsicherheit (engl. **Dilution of Precision (DOP)**), Rohzeitstempel und die Oszillator-Frequenz der **DAU** werden vom **NMEAParser**-Prozess als **DataMessage-Structs** in die **Data-FreeRTOS-Mail-Queue** geschrieben. Der **DataStreamer**-Prozess liest (osMailGet) kontinuierlich **DataMessage-Structs** aus der *FreeRTOS-Mail-Queue*. Die Rohzeitstempel werden mit dem *Counter-Absolutzeit-Konverter* in Absolutzeiten mit Unsicherheitsangabe konvertiert. Anschließend werden die **Protobuf**-**DataMessage-Structs** im **Ethernet**-Paketpuffer binär serialisiert. Wenn die maximale Paketgröße (1 **KiB**) erreicht ist, werden die serialisierten Daten mittels **DMA** in den **Ethernet-Media-Access-Controller** transferiert und von diesem über **UDP** an einen Empfänger versendet. In regelmäßigen Abständen (alle 2 s) versendet der **DataStreamer**-Prozess **Protobuf DescriptionMessages** für jeden in der **DAU** konfigurierten Sensor. Datenpakete, die während des **DescriptionMessage**-Versandes in der *FreeRTOS-Mail-Queue* auflaufen, werden anschließend verarbeitet.

Die Konfiguration der **UDP**-Zieladresse erfolgt über ein **Webinterface**, welches von der **DAU** bereitgestellt wird. Die **UDP**-Adresse wird zusammen mit einigen anderen Parametern im **Backup-SRAM** abgespeichert. In Zukunft soll eine Sensorkonfiguration über dieses **Webinterface** realisiert werden. Die **Ethernet-IP**-Adresse wird über **DHCP** bezogen und auf dem **Status-LCD** angezeigt. **Basis-ID** und **Ethernet-MAC**-Adresse werden mittels **Cyclic Redundancy Check (CRC)** aus der 96 Bit langen **Unique Device Identifier (UID)** der **CPU** berechnet. Zur Ansteuerung der **STM32**-Peripherie wird der **STM32 Hardware Abstraction Layer** [61] verwendet. Für die Implementierung der **Ethernet**-Protokolle wird der quelloffene **LwIP**-Stack verwendet [29].

3.1.2 Konversion und Übertragung der Messwerte in SI-Einheiten

Die **DAU** erfasst die Messwerte digitaler Sensoren in der Regel als Ganzahlwerte (engl. *Integer*), die aus einer im Sensor stattfindenden AD-Konversion und Signalkonditionierung stammen. In der Industrie hat sich die Einheit **LSB**, also Vielfache der Integerquantisierung, für solche Ganzahlwerte etabliert. Für die Verwendung dieser Messwerte müssen sie in die Einheit der physikalischen Messgröße konvertiert werden. Die Sensitivitäten der Sensoren in **LSB** pro Messgröße sind Teil der Spezifikationen und werden daher im Datenblatt angegeben.

Oft kann der Messbereich der Sensoren konfiguriert werden, was den Skalierungsfaktor entsprechend verändert. Bei der in der **DAU** stattfindenden Konversion in **SI**-Einheiten wird der Messbereich entsprechend berücksichtigt. Zur Darstellung und Übertragung dieser Werte werden die *Unit Strings* des **Digital System of Units (D-SI)**-Formats verwendet. Tabelle 3.1 gibt einen Überblick über die verwendeten Einheiten und ihre Darstellung im **D-SI**. Für die Übertragung der Messdaten und Metainformationen wird ein zustandsloses Protokoll, basierend auf **Protobuf**, verwendet. Abbildung 3.6 zeigt schematisch den Inhalt der von der **DAU** versendeten Daten- und Beschreibungsdatagramme für eine **MPU-9250 Inertial Measurement Unit (IMU)**. Die **Protobuf**-Nachrichtendefinition für die Datagramme kann dem Listing 3.1 entnommen werden.

Tabelle 3.1: Verwendete Einheiten im D-SI-Format [44]

Größe	Einheit	D-SI
Länge	m	\metre
Frequenz	Hz	\hertz
Beschleunigung	$\frac{\text{m}}{\text{s}^2}$	\metre\per\second\tothe{2}
Geschwindigkeit	$\frac{\text{m}}{\text{s}}$	\metre\per\second
Winkel	°	\degree
Winkel	rad	\radian
Winkelgeschwindigkeit	$\frac{\text{rad}}{\text{s}}$	\radian\per\second
Winkelgeschwindigkeit	$\frac{\circ}{\text{s}}$	\degree\per\second
Temperatur	°C	\degreecelsius
Magnetische Flussdichte	μT	\micro\tesla
Spannung	v	\volt
Zeit	s	\seconds
Zeit	ns	\nano\seconds

Tabelle 3.2: Zusammensetzung der IDs

MSB [31:16]	[15:8]	[7:0] LSB
DAU-Base-ID	Sensor-ID	Configuration-ID

Die einzelnen Felder einer **Protobuf**-Nachricht werden immer mit einem Label (`required`, `optional`, `repeated`), einem Datentypen (z. B. `uint32`, `uint64`, `float` oder `string`) sowie einem Namen und einer Feldnummer definiert. Jede **Protobuf**-Nachricht der **DAU** enthält zwingend (`required`) das Feld (`required uint32 id`) für die **ID** des Senors von dem die Nachricht stammt. Jede 32-Bit-ID enthält eine 16-Bit-Basis-ID zur Identifikation der **DAU**, weitere 8 Bit für den Sensor sowie 8 Bit für die Konfiguration des Sensors. Die Zusammensetzung der **ID** ist in Tabelle 3.2 gezeigt. Die Datagramme enthalten als zweites notwendiges Feld `required uint32 sample_number` die in der Sensor-**ISR** fortlaufend inkrementierte Anzahl der bisher erfassten Datenpakete. **UDP** ist ein sehr schnelles und schlankes Datenübertragungsprotokoll, bietet im Gegenzug aber keine Paketverlustdetektion und folglich auch nicht die Möglichkeit, Pakete ggf. noch einmal zu versenden. Sollte die Daten-**FreeRTOS-Mail-Queue** in der **DAU** volllaufen, kommt es auch zum Verlust von Paketen. In beiden Fällen wird die `sample_number` jedoch korrekt hochgezählt, was eine nachträgliche Detektion des Paketverlusts ermöglicht. Die Felder `required uint32 unix_time`, `required uint32 unix_time_nsecs` und `required ← uint32 time_uncertainty` enthalten die Absolutzeitinformationen zusammen mit der abgeschätzten Unsicherheit der Zeitstempelung. Danach folgen in der Nachricht das Feld (`required float Data_← 01`), welches die Messdaten des ersten Sensorkanals enthält. Zusätzlich existieren noch fünfzehn weitere optionale Datenfelder, z. B. `optional float Data_02` für Sensoren, die verschiedene Messgrößen simultan erfassen, wie beispielsweise dreiachsig Beschleunigung und Temperatur. Das Feld `optional uint64 time_ticks` enthält optional die Rohzeitstempel in **Counts**.

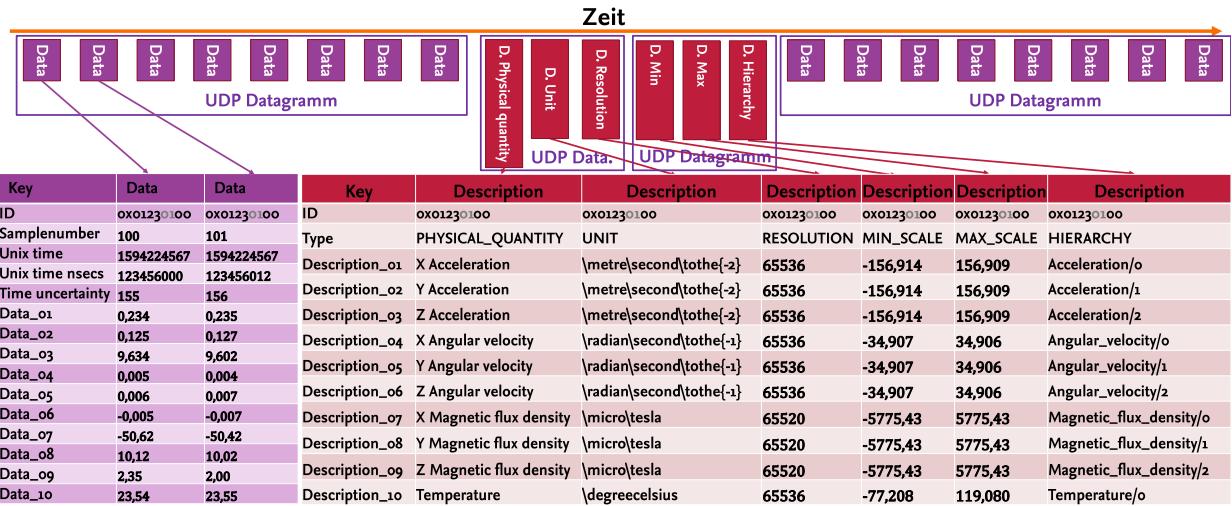


Abbildung 3.6: **Zustandsloses Daten- und Beschreibungsprotokoll.** Die Datenpakete (violett) werden kontinuierlich in Blöcken zu je 1 KiB, also mit etwa acht Sensordaten-Paketen, versendet. Alle zwei Sekunden werden für jeden an der DAU angeschlossenen Sensor Beschreibungspakete (rot) versendet.

Die Beschreibung der Messdatenströme in den einzelnen Kanälen erfolgt über verschiedene `DescriptionMessages`. Im Listing 3.2 ist die `Protobuf`-Nachrichtendefinition gezeigt. Die Abbildung 3.6 zeigt in der roten Tabelle beispielhaft den Inhalt der verschiedenen `DescriptionMessages` für eine `MPU-9250 IMU`. Jede `DescriptionMessage` enthält die Sensor-ID und den Sensornamen als `string` mit maximal 40 Zeichen Länge. Im Feld `required DESCRIPTION_TYPE Description_Type` wird der Typ der `DescriptionMessages` als einer von sechs möglichen `enum`-Keys kodiert.

Nachrichten des Typs `PHYSICAL_QUANTITY` enthalten in den Feldern `optional string str_Data` den Namen der physikalischen Messgröße in englischer Sprache. Die `SI`-Einheit der Messgrößen wird mit dem `UNIT`-Key gekennzeichnet, die Felder `str_Data` enthalten dann die `D-SI`-Repräsentation der jeweiligen Einheit. Die einzelnen Datenfelder haben untereinander keine Beziehung. Um Beziehungen der Datenfelder untereinander darzustellen, dienen die `HIERARCHY`-Nachrichten. Die Zahlen in der tieferen Ebene werden zum Indizieren der Datenströme verwendet. Es wird mit Null-Indizierung gearbeitet. Für kartesische Vektorgrößen ist die Indizierungskonvention $x=0$, $y=1$, $z=2$. Die Nachrichten mit den Keys `MIN_SCALE` und `MAX_SCALE` kodieren in den Feldern `optional float f_Data` die minimalen bzw. maximalen Messbereiche. In `RESOLUTION`-Nachrichten wird die Gesamtzahl der möglichen Messwerte (256 für einen 8-Bit-Sensor) übertragen. Allein die Auflösung in Bit ist nicht ausreichend, da einige Sensoren bestimmte Wertebereiche nicht nutzen, z. B. das Magnetometer im `MPU-9250`. Die Breite ΔX der als äquidistant angenommenen Quantisierungsintervalle der Messgröße x kann mit folgender Gleichung errechnet werden.

$$\Delta X = \frac{\text{MAX_SCALE} - \text{MIN_SCALE}}{\text{RESOLUTION} - 1} \quad (3.1)$$

In den **UDP**-Paketen ist jeder **Protobuf**-Nachricht ein Header vorangestellt, welcher eines der beiden **ASCII**-Keywords (DATA → Daten, DSCP → Beschreibungen) sowie die Nachrichtenlänge in Bytes als **Protobuf**-Variable vom Typ **VarInt** enthält, da **Protobuf** die Übertragung der einzelnen serialisierten Nachrichten nicht spezifiziert und daher die einzelnen Nachrichtentypen für die Deserialisierung ansonsten nicht richtig unterschieden werden können.

Code Listing 3.1: **ProtoBuf-Code DataMessage**

```

1  message DataMessage {
2      required uint32 id = 1;           //32 bit ID of the sensor
3      required uint32 sample_number = 2; //Number of Packets send
4      required uint32 unix_time = 3;    //Absolutetimestamp in unix seconds
5      required uint32 unix_time_nsecs = 4; //Absolutetimestamp ns fraction
6      required uint32 time_uncertainty = 5; //Uncertainty of time stamp in ns
7      required float Data_01 =6;        //Data of 1st channel as float number
8      optional float Data_02 =7;        //Data of 2nd channel as float number
9      optional float Data_03 =8;        //Data of 3rd channel as float number
10     optional float Data_04 =9;        //Data of 4th channel as float number
11     optional float Data_05 =10;       //Data of 5th channel as float number
12     optional float Data_06 =11;       //Data of 6th channel as float number
13     optional float Data_07 =12;       //Data of 7th channel as float number
14     optional float Data_08 =13;       //Data of 8th channel as float number
15     optional float Data_09 =14;       //Data of 9th channel as float number
16     optional float Data_10 =15;       //Data of 10th channel as float number
17     optional float Data_11 =16;       //Data of 11th channel as float number
18     optional float Data_12 =17;       //Data of 12th channel as float number
19     optional float Data_13 =18;       //Data of 13th channel as float number
20     optional float Data_14 =19;       //Data of 14th channel as float number
21     optional float Data_15 =20;       //Data of 15th channel as float number
22     optional float Data_16 =21;       //Data of 16th channel as float number
23     optional uint64 time_ticks=22;    //Raw ticks of input capture counter
24 }
25 }
```

Code Listing 3.2: **ProtoBuf-Code DescriptionMessage**

```

1 message DescriptionMessage {
2     required uint32 id = 1;      //32 Bit ID of the sensor [0x1FE40100]
3     required string Sensor_name = 2 [(nanopb).max_size = 40];
4     //Name of the Sensor [MPU-9250]
5     enum DESCRIPTION_TYPE{
6         PHYSICAL_QUANTITY = 0; //String fields describe the unit of the values
7         UNIT = 1;             //String fields describe the unit of the values in DSI
8         UNCERTAINTY_TYPE = 2; // RESERVED
9         RESOLUTION = 3;      // Steps between min and max val. [256] for 8 bit Sensor
10        MIN_SCALE = 4;       // Maximal possible value
11        MAX_SCALE = 5;       // Minimal possible value
12        HIERARCHY = 6;       //Information about hierarchical structure of data see ←
13        examples
14    }
15    required DESCRIPTION_TYPE Description_Type =3;
16    optional string str_Data_01 =4 [(nanopb).max_size = 40];
17    optional string str_Data_02 =5 [(nanopb).max_size = 40];
18    optional string str_Data_03 =6 [(nanopb).max_size = 40];
19    optional string str_Data_04 =7 [(nanopb).max_size = 40];
20    optional string str_Data_05 =8 [(nanopb).max_size = 40];
21    optional string str_Data_06 =9 [(nanopb).max_size = 40];
22    optional string str_Data_07 =10 [(nanopb).max_size = 40];
23    optional string str_Data_08 =11 [(nanopb).max_size = 40];
24    optional string str_Data_09 =12 [(nanopb).max_size = 40];
25    optional string str_Data_10 =13 [(nanopb).max_size = 40];
26    optional string str_Data_11 =14 [(nanopb).max_size = 40];
27    optional string str_Data_12 =15 [(nanopb).max_size = 40];
28    optional string str_Data_13 =16 [(nanopb).max_size = 40];
29    optional string str_Data_14 =17 [(nanopb).max_size = 40];
30    optional string str_Data_15 =18 [(nanopb).max_size = 40];
31    optional string str_Data_16 =19 [(nanopb).max_size = 40];
32    optional float f_Data_01 =20;
33    optional float f_Data_02 =21;
34    optional float f_Data_03 =22;
35    optional float f_Data_04 =23;
36    optional float f_Data_05 =24;
37    optional float f_Data_06 =25;
38    optional float f_Data_07 =26;
39    optional float f_Data_08 =27;
40    optional float f_Data_09 =28;
41    optional float f_Data_10 =29;
42    optional float f_Data_11 =30;
43    optional float f_Data_12 =31;
44    optional float f_Data_13 =32;
45    optional float f_Data_14 =33;
46    optional float f_Data_15 =34;
47    optional float f_Data_16 =35;
48    required bool has_time_ticks =36; }      // has 64bit raw timestamp or not

```

3.1.3 C++-Schnittstelle für Sensoren

Zur Vereinheitlichung der Sensoransteuerung wird die gemeinsame abstrakte Basisklassen-Deklaration **Met4FoFSensor** verwendet. Jede davon erbende Implementierung eines SensorTreibers auf der **DAU** muss mindestens die Memberfunktionen der **Met4FoFSensor**-Klasse als nicht-virtuelle Funktionen implementieren (siehe Code Listing 3.3).

Code Listing 3.3: **Virtuelle Klasse als gemeinsame Treiber-API in C++**

```

1 class Met4FoFSensor {
2     public:
3         // virtual type var=0 is needed to generate vtable for linking with only ←
4         // virtual functions
5         virtual int getData(DataMessage * Message, uint64_t RawTimeStamp)= 0; //data ←
6             // getter function handles sensor communication
7         virtual int getDescription(DescriptionMessage * Message, ←
8             DescriptionMessage_DESCRIPTION_TYPE DESCRIPTION_TYPE)= 0; // get the ←
9             // protobuf description
10        virtual uint32_t getSampleCount()= 0; // get sample count
11        virtual float getNominalSamplingFreq()= 0; // get nominal sampling freq
12        virtual int setBaseID(uint16_t BaseID)= 0; // set base id
13        protected:
14            bool _publish_time_ticks=false;
15

```

Die Funktion `int getData(DataMessage * Message, uint64_t RawTimeStamp)` wird verwendet, um die Messdaten des Sensors über das Hardware-Interface zu lesen. Die Funktion gibt `1` bei Erfolg und `0` im Fehlerfall zurück.

Der Zeiger (engl. *pointer*) `DataMessage * Message`, welcher der Funktion übergeben wird, zeigt auf einen durch die FreeRTOS-Funktion `dataMsgPointer = (DataMessage*) osMailAlloc(DataMail, 0)` allokierten Speicherbereich in einer *FreeRTOS-Mail-Queue*.

Die Funktion `getData(...)` schreibt den Protobuf-Nachrichten-Struct direkt in diesen Speicherbereich. Die ebenfalls übergebene Variable `uint64_t RawTimeStamp` enthält den in der **ISR** ausgelesenen Zählerwert der **DAU**-Zeitbasis zum Zeitpunkt der Interruptauslösung. Dieser Wert wird in die Felder `uint32_t unix_time` und `uint32_t unix_time_nsecs` des **Structs** als 32-Bit-Ganzzahl gespeichert. Das Feld `uint32_t time_uncertainty` wird auf `0xFFFFFFFF` gesetzt, um zu kennzeichnen, dass es sich um einen Rohzeitstempel und keine Absolutzeit handelt. Bei jedem Aufruf dieser Funktion wird die Membervariable `uint32_t _SampleCount` des korrespondierenden Sensors um `1` inkrementiert; ihr Wert wird zudem in das Feld `uint32_t sample_number` des **Protobuf-Structs** kopiert. Ist die Nachrichten-*FreeRTOS-Mail-Queue* voll, kann die Funktion `osMailAlloc(DataMail, 0)` keinen Speicher allozieren und gibt einen NULL-Zeiger zurück. Wird die `getData(...)`-Funktion mit einem NULL-Zeiger aufgerufen, wird ausschließlich die Sensor-Klassenvariable `uint32_t _SampleCount` um `1` inkrementiert und die Funktion sofort verlassen, um Rechenzeit zu sparen, da ein Auslesen der Daten ohne Ziel-Speicherort nicht sinnvoll ist; der Rückgabewert ist in diesem Fall `0`. Die Funktion `int getDescription(DescriptionMessage * Message, Description\-\Message_DESCRIPTION_TYPE ←` `DESCRIPTION_TYPE)`, welche mit einem Zeiger auf ein **Protobuf-DescriptionMessage-Struct** sowie einem der **ENUM-Keys** `Description\-\Message_DESCRIPTION_TYPE_.....PHYSICAL_QUANTITY, ...UNIT, ...RESOLUTION←`

, ...MIN_SCALE, ...MAX_SCALE, ...HIERARCHY aufgerufen wird, erzeugt die entsprechenden Einträge in der `Description`-Message. Die Funktion `uint32_t getSampleCount()` gibt die Anzahl der schon erfassten Messwerte zurück. Die Funktion `float getNominalSamplingFreq()` liefert die aktuell konfigurierte nominelle Abtastrate des Sensors zurück. Diese beiden Funktionen können verwendet werden, um einen *Watchdog* zu implementieren, welcher den Sensor nach Fehlern (z. B. einer kurzzeitigen Unterbrechung der Versorgungsspannung) reinitialisiert. Die Funktion `int setBaseID(uint16_t ← BaseID)` wird verwendet, um die *Basis-ID* des Sensors entsprechend der *DAU-ID* zu konfigurieren (siehe Tabelle 3.2). Die *DAU-ID* ist erst zur Laufzeit des Systems bekannt, da sie für jede *DAU* individuell zusammen mit der *Ethernet-MAC*-Adresse durch *CRC-16* aus der 96 Bit langen Hardware-*UID* des *STM32F767zi*-Microkontrollers generiert wird. Die Konfiguration der Sensoren erfolgt aktuell über sensorspezifische Übergabeparameter im Konstruktor der Sensorklasseninstanz und/oder spezifische Funktionen zur Laufzeit. Für Konfigurationsänderungen sind folglich Änderungen des *DAU*-Quellcodes nötig. Dies ist für Nutzer ohne Programmierkenntnisse nur schwer möglich. In Zukunft soll die Konfiguration der Sensoren mittels *JSON*-Steuerdateien erfolgen. Die Sensorkonfiguration könnte dann über das Webinterface der *DAU* erfolgen.

3.2 Software für den Empfang der Protobuf-Nachrichten auf einem PC

Für den Empfang und die Weiterverarbeitung der von den *DAUs* erfassten Sensordaten wird eine in *Python* programmierte *DataReceiver*-Software verwendet. Abbildung 3.7 zeigt schematisch den Datenfluss innerhalb der *DataReceiver*-Applikation beim Speichern der Messdaten in eine gemeinsame *HDF*-Datei. Die Applikation ist nebenläufig (engl. *multithreaded*) programmiert. Da die Verarbeitung der Sensordaten voneinander unabhängige Aufgaben sind, kann hierbei jede Sensor-datenverarbeitung in einem eigenen *Thread* innerhalb der Applikation ausgeführt werden. Durch das *Multithreading* blockieren langsame Datenverarbeitungsschritte für einen Sensor nicht die der anderen Sensoren. Abbildung 3.8a zeigt das Flussdiagramm der Endlosschleife des Hauptthreads. Die *UDP*-Nachrichten werden aus dem *Ethernet*-Puffer des Betriebssystems gelesen und anschlie-ßend werden die einzelnen *Protobuf*-Nachrichten deserialisiert. In den *UDP*-Paketen ist jeder einzelnen *Protobuf*-Nachricht ein Header vorangestellt, welcher eines der beiden *Keywords* [DATA → Daten, DSCP → Beschreibungen] sowie die Nachrichtenlänge in Bytes als *Protobuf* Variable vom Typ *VarInt* enthält. Mit diesen Informationen kann jede *UDP*-Nachricht verarbeitet werden. Die Nachrichten werden deserialisiert und die *Sensor-ID* ausgelesen. An dieser Stelle existieren zwei verschiedene Möglichkeiten.

Erstens, die *ID* des Sensors ist unbekannt: in diesem Fall wird eine neue Instanz der `class Sensor` erzeugt und dem `Dict AllSensors` hinzugefügt. Bei der Initialisierung der Sensorklasseninstanz wird ein neuer Thread gestartet, in dem die komplette Verarbeitung der *Protobuf*-Nachrichten dieses Sensors stattfindet. Es wird außerdem mit dem Aufruf `multi\processing.Queue(int(BufferSize))` eine `Queue` mit der Standardgröße `BufferSize=25e5` in der Sensorklasseninstanz erstellt, über die die *Protobuf*-Nachrichten an den Sensorthread gesendet werden.

Zweitens, die *Sensor-ID* ist schon bekannt: die *Protobuf*-Nachricht wird sofort über die `multi\← processing\-.Queue` an die Sensorklasseninstanz gesendet.

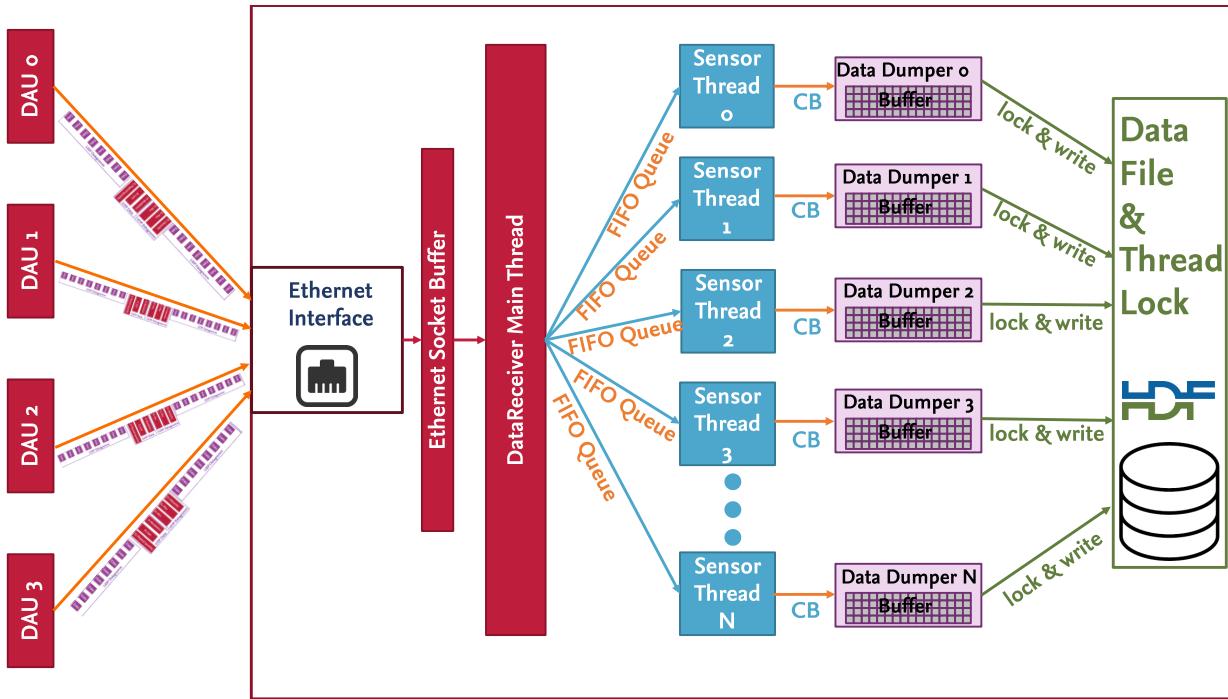


Abbildung 3.7: **Datenfluss in der DataReceiver Software auf dem Empfangs-PC.** Verschiedene DAUs können an den selben Empfänger (DataReceiver) Daten senden. Im Ethernet-Stack des Betriebssystems werden die auf dem für die DataReceivers Software konfigurierten Port ankommenen Pakete gepuffert. Im Hauptthread werden die Sensor-IDs der Protobuf-Nachrichten aus den UDP-Datagrammen geparsert. Für jeden Sensor wird ein Sensorthread gestartet. Die Protobuf-Nachrichten werden über eine FIFO-Queue an die Sensorthreads übertragen. In den Threads findet nach der Deserialisierung die komplette Verarbeitung der Beschreibungen und Daten statt. Zum Speichern der Sensordaten werden DataDumper-Threads verwendet. Diese verwenden einen Ringpuffer und schreiben nach der Pufferung die Daten blockweise in eine mit einem Thread Lock geschützte HDF-Datei. Die Fluss-Diagramme für die einzelnen Threads finden sich in Abb. 3.8a und 3.8b.

Im Haupthread wird die Sensor-ID ohne Deserialisierung direkt mittels Offsetadressierung aus der binärkodierten Nachricht gelesen. Die zeitintensive Deserialisierung findet anschließend im Sensorthread statt. Innerhalb der Sensorthreads läuft eine Endlosschleife, welche die Protobuf-Nachrichten aus der Queue verarbeitet. Die Abbildung 3.8b zeigt das Flussdiagramm dieser Endlosschleife. Jede Sensorinstanz stellt die Meta-Informationen über ein Dict-Interface bereit, wobei die Datenkanäle entweder über ihren Index [0, 1, 2 ... 15] oder über PHYSICAL_QUANTITY aufgerufen werden können. Die Sensorbeschreibung wird beim Empfang neuer Description-Nachrichten aktualisiert. Wegen des zustandslosen Protokolls ist die Beschreibung der Sensoren bis zum Erhalt aller Description-Nachrichten unvollständig.

Die Verarbeitung der empfangenen Daten erfolgt flexibel durch Callbacks mit der Funktionssignatur `callback(message["ProtMsg"], self.Description)`. Das Abspeichern der Messdaten und Metadaten in HDF-Dateien erfolgt auch über die Callbacks. Für jeden Sensor wird eine Instanz der Klasse `class HDF5Dumper` erzeugt und die Funktion `HDF5Dumper.pushmsg` als Callback für den entsprechenden Sensor registriert. Jeder `HDF5Dumper` hat einen `chunksize=2048` Elementen großen Buffer für die Messda-

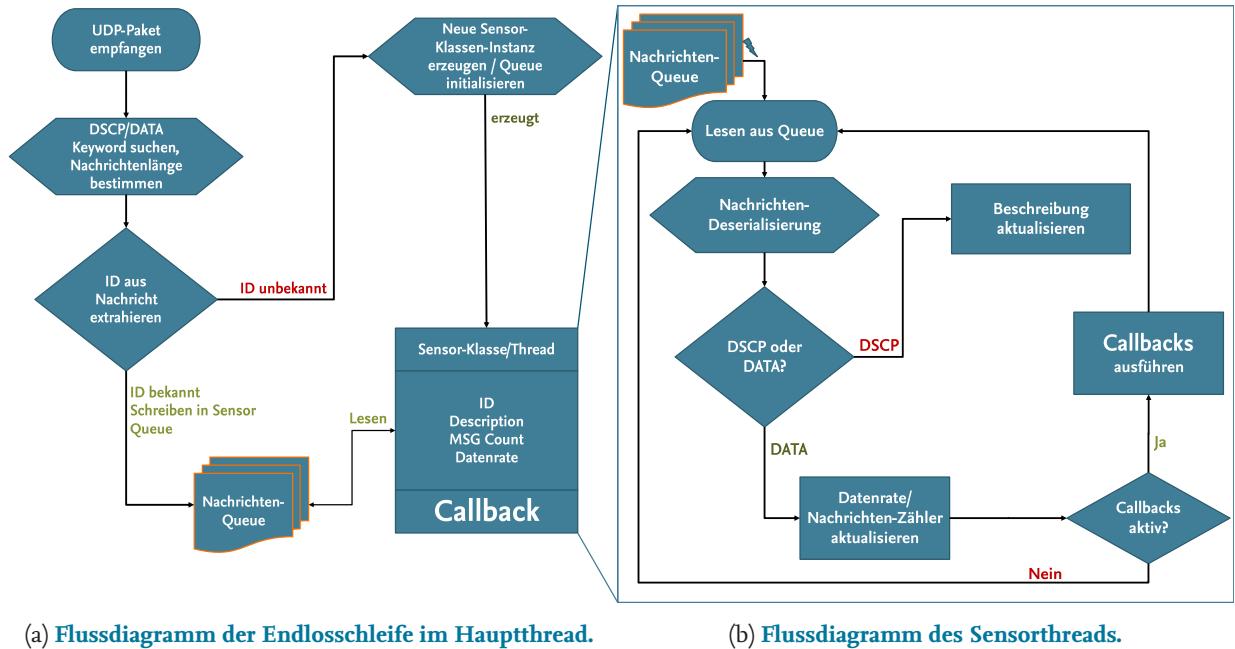


Abbildung 3.8: **Flussdiagramme des Datenempfangs.** (a) Im Hauptthread wird in den **UDP**-Paketen nach den Keywords **DSCHP** oder **DATA** gesucht. Direkt hinter diesen befindet sich die Länge der folgenden **Protobuf**-Nachricht als **uint32** Variable. Aus dem **UDP**-Paket wird ohne komplette Deserialisierung die Sensor-ID gelesen. Ist die **ID** unbekannt, wird eine neue Sensor-Klasseninstanz initialisiert. In jedem Fall wird die Nachricht in die **Queue** (Ringpuffer mit Signalisierung) des Sensors geschrieben (push) und das Lesen des **UDP**-Pakets hinter den verarbeiteten Daten fortgesetzt, bis das gesamte **UDP**-Paket verarbeitet ist. (b) Im Sensorthread werden die Nachrichten aus der **Queue** gelesen (pop). **DSCHP**-Nachrichten (Beschreibungen) führen zu einer Aktualisierung der Beschreibung. Datenpakete lösen direkt die **Callbacks** (Rückruffunktionen) für die Datenverarbeitung und/oder Speicherung aus.

ten. Wenn der **Buffer** vollständig gefüllt ist, wird das zur **HDF** gehörende **threading.Lock** akquiriert und die Daten werden entsprechend der Hierarchie blockweise und **gzip**-komprimiert in die **HDF**-Datei geschrieben. Ein unsynchronisierter Schreibzugriff auf die **HDF**-Datei kann diese korrumperen, daher ist ein exklusiver Zugriffsschutz durch Synchronisierung der Schreibzugriffe zwischen den Threads mittels **ThreadLock** oder eines **Mutex** zwingend nötig. Paralleler Lesezugriff hingegen ist möglich.

3.2.1 HDF-Dateiformat

Sowohl bei den Messungen als auch bei den Berechnungen im *Postprocessing* fallen große heterogene Datensätze an, welche strukturiert gespeichert werden müssen. Das **Hierarchical Data Format**, [100] wurde für solche Anwendungsfälle entwickelt, in dieser Arbeit wird konkret das **HDF5** Dateiformat verwendet. In **HDF**-Dateien können Datensätze (*Datasets*) strukturiert in Gruppen abgelegt werden, ähnlich wie Dateien in Ordnern von Dateisystemen. Sowohl *Datasets* als auch *Groups* können Metadaten in Form von Attributen tragen. Abbildung 3.9 zeigt schematisch den Aufbau einer **HDF**-Datei. In *Datasets* können bis zu vierdimensionale *Arrays* binär gespeichert werden. Viele Datentypen wie

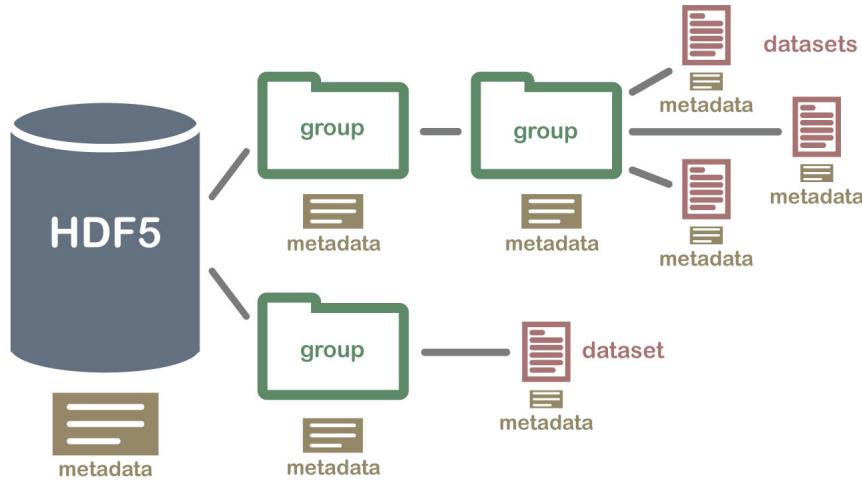


Abbildung 3.9: **Datenstrukturierung in einer HDF-Datei.** Die Datensätze *Datasets* werden in Gruppen gespeichert. Dabei können Metadaten in den Attributen von *Datasets* und Gruppen abgelegt werden [106].

`float32`, `float64`, `int32`, `int64`, `uint32` oder `uint64` werden nativ von **HDF** unterstützt. Die *Datasets* können in Blöcke (*Chunks*) mit z. B. [2048, 3] Datenpunkten aufgeteilt werden. Die einzelnen *Chunks* können unabhängig voneinander und parallel ausgelesen werden. Außerdem ist eine für den Nutzer transparente Kompression der *Chunks* mit verschiedenen Algorithmen, wie z. B. `gzip`, möglich. Bei den in dieser Arbeit aufgezeichneten Datensätzen wird durch Kompression eine Verkleinerung der Dateigröße um etwa 90 % erreicht. Die Zugriffszeiten werden durch die Kompression nur minimal verlängert. **HDF** besitzt ähnlich wie ein Dateisystem auch die Möglichkeit, Links auf Teile von *Datasets* zu erstellen und auf diese wie auf normale *Datasets* zuzugreifen.

Die Beziehung der *Datasets* untereinander kann durch die Attribute `DIMENSION_LABELS` und `DIMENSION_LIST` beschrieben werden. In den in dieser Arbeit aufgezeichneten Rohdaten sind die Zeitstempel der Messwerte im *Dataset* `Absolutetime` gespeichert; jedem Vektor aus dem `Acceleration-Dataset` wird der `Absolutetime`-Wert mit gleichem Index zugeordnet. Diese Zuordnung wird in den `DIMENSION_LIST`-Attributen als Referenz auf das `Absolutetime-Dataset` gespeichert. Für die zweite Array-Dimension wird keine Dimension verwendet, da diese die x-, y-, z-Vektorkomponenten der Beschleunigung enthält. Das Attribut `DIMENSION_LABELS` speichert die Anzeigenamen der Dimensionen.

Im Folgenden sollen die einzelnen Gruppen und *Datasets* der **HDF**-Dateien genauer erklärt werden.

Die Abbildung 3.10 zeigt die meisten Gruppen und einige *Datasets* einer **HDF**-Datei mit der Aufzeichnung der Kalibrierung einer **MPU-9250 IMU** auf der Normalmesseinrichtung des **CEM**.

RAWDATA: Diese Gruppe enthält die Rohdaten, welche von den **DAUs** geliefert und durch die **HDF5-Dumper-Klassen** abgespeichert wurden. Für jeden Sensor wird eine Gruppe erstellt. Der Gruppenname besteht aus der Hexadezimaldarstellung der **ID** und des Sensornamens, wobei Leerzeichen durch Unterstriche ersetzt werden. Die Sensorgruppen besitzen die Attribute `Sensor_ID`, `Sensor-`

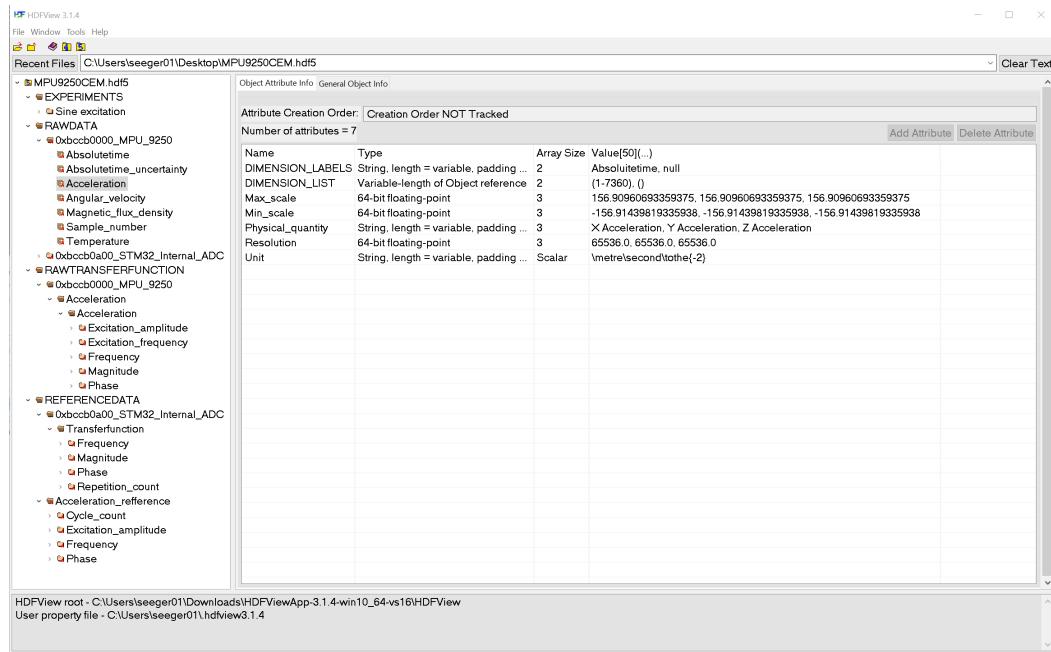


Abbildung 3.10: **Struktur der HDF-Datei einer Beschleunigungskalibrierung in HDFView.** Auf der linken Seite wird eine Übersicht der Gruppen und *Datasets* angezeigt. Auf der rechten Seite sind die Attribute des *Datasets* `RAWDATA/0xbccb0000_MP9250/Acceleration` zu sehen.

`_name` sowie `Start_time` und `Start_Time_uncertainty`, welche den Zeitpunkt des ersten Datenpunktes enthalten. Zusätzlich findet sich die als **JSON** serialisierte Beschreibung des Sensors im Attribut `Data_description_json`. Die *Datasets* in dieser Gruppe enthalten die entsprechend der Hierarchie abgelegten Messdaten.

EXPERIMENTS: Diese Gruppe enthält für jede von der *Postprocessing*-Software erkannte Kalibrierungsmessung eine Untergruppe. Der Gruppenname setzt sich aus einer fortlaufenden Zahl und dem Typ des Experiments zusammen, z. B. `00015_Sine_Excitation` für eine Kalibrierung mit monofrequenter Sinusanregung. Die Gruppe trägt als Attribute die Start- und Endzeitpunkte der Experimente. In den Experimentgruppen finden sich für jeden Sensor und jedes Sensor-*Dataset* Gruppen, welche die für diese Messgröße bestimmten Koeffizienten enthalten. Im Falle der Sinusanregung sind dies die durch die **Sequentielle-Drei-Parameter-Sinus-Approximation** bestimmten Parameter inklusive Kovarianzen.

REFERENCE DATA: In dieser Gruppe werden die Daten von externen **ACSS** sowie bekannten Transferfunktionen, z. B. für den internen **ADC** der **DAUs**, abgelegt.

RAWTRANSFERFUNCTION: In dieser Gruppe werden die aus den **ACS**-Daten und Sinuskoeffizienten der Experimente bestimmten Transferkoeffizienten abgelegt.

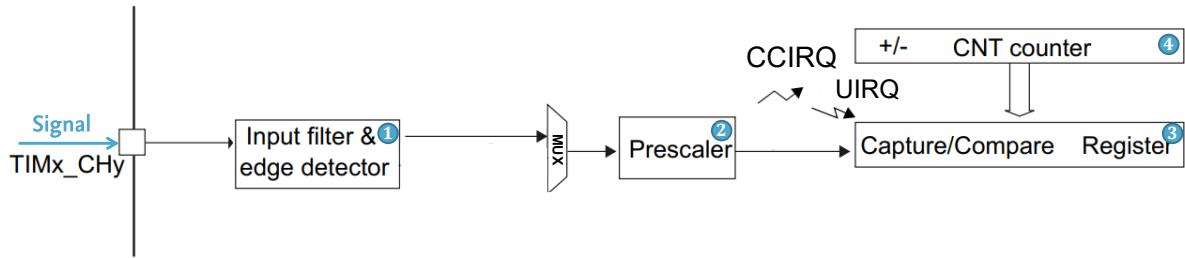


Abbildung 3.11: **Aufbau eines STM32-Timer/Counter-Eingangs.** Durch den Eingangsfilter ① werden transiente Signale unterdrückt. Die Signale können optional mit einem *Integerteiler* variabel heruntergeteilt werden ②. Im *Capture/Compare-Register* ③ wird der aktuelle Zählerstand ④ gespeichert und eine *Timer Capture/Compare Interrupt Request* (‘CCIRQ’) ausgelöst. Läuft der *Timer* über, wird eine *Timer Update Interrupt Request* (‘UIREQ’) ausgelöst und der Zähler auf den Wert des *Reload/Preload-Registers* (0x00000000) zurückgesetzt. Verändert nach [97].

3.3 GNSS-Zeitbasis

Für die Absolutzeitstemplung der Sensor-IRQ-Signale wird eine **GNSS**-Absolutzeitbasis verwendet. Das Akronym **GNSS** steht dabei für **Global Navigation Satellite System**, also z. B. **GPS** oder **Gallileo**. Da die Satellitennavigation im Wesentlichen auf der Bestimmung der Laufzeitdifferenz von Satellitensignalen basiert, liefern **GNSS**-Empfänger nicht nur eine gute Positionsbestimmung, sondern auch eine hochgenaue Zeitreferenz. Im Folgenden wird die Realisierung dieser Zeitbasis erläutert. Der Mikrocontroller **STM32F767ZI** besitzt mehrere 16- und 32-Bit-*Multipurpose-Timer/Counter*, die unter anderem auch den *Input-Capture*-Modus für jeweils bis zu vier Kanäle unterstützen. Abbildung 3.11 zeigt den Signalfluss von der Detektion einer externen Signalflanke bis zum Auslösen der *Timer Capture/Compare Interrupt Request* durch die *Timer*-Hardware.

Im *Input-Capture*-Modus wird der *Counter*-Wert des *Timers* bei der Detektion einer externen Signalflanke in den *Capture/Compare-Registern* gespeichert. Erst bei der nächsten externen Flanke wird dieser Wert überschrieben. Die Zeitbasen der *Counter* können hardwareynchronisiert gestartet werden. Die Zeitbasen der *Timer* werden in der **DAU** mit 108 MHz betrieben. Die 32-Bit-*Counter* laufen folglich nach $\frac{2^{32}}{108 \cdot 10^6 \text{ s}^{-1}} \approx 40 \text{ s}$ über, die 16-Bit-*Counter* hingegen schon nach $\frac{2^{16}}{108 \cdot 10^6 \text{ s}^{-1}} \approx 600 \mu\text{s}$. Um Probleme durch Zählerüberlauf zu vermeiden, werden die *InputCaptureCounts*-Variablen in 64-Bit-Zeitstempel konvertiert. Dabei entsprechen die oberen 32/48 Bit der Anzahl der Timerüberläufe seit dem Timerstart. Für die 32-Bit-*Timer* können die oberen 32 Bit des 64-Bit-*Counter*-Werts direkt in der **Timer Update Interrupt Request (UIREQ)-ISR** des korrespondierenden *Timers* behandelt werden. Für die 16-Bit-*Timer* ist dies in Software aufgrund der hohen Interruptfrequenz von 1647 Hz nicht ohne starke **CPU**-Auslastung möglich. Deshalb werden jeweils zwei 16-Bit-*Timer* seriell verwendet, und der Zähler des einen *Counters* wird bei jedem Überlauf des anderen *Counters* inkrementiert. Zusammen können diese beiden *Counter* wie ein 32-Bit *Counter* verwendet werden (siehe auch [97]). Die so erhaltenen 64 Bit langen *InputCaptureCounts* tragen allein noch keine Absolutzeitinformation. Zusätzlich zu den Sensorinterrupt-Flanken wird auch das **PPS**-Signal eines **GNSS**-Moduls durch

die *Counter* erfasst. Zusammen mit den aus den **NMEA**-Nachrichten extrahierten Zeitinformationen werden so Tupel aus **UTC**-Zeit $t(n)$ und korrespondierendem Zählerstand c_n erhalten. Mit zwei aufeinander folgenden Datenpaaren kann die aktuelle Zählerfrequenz direkt wie folgt errechnet werden:

$$f_{\text{cnt}} = \frac{c_n - c_{n-1}}{t_n - t_{n-1}} \quad (3.2)$$

Die Konversion eines Zählerwertes c in eine Absolutzeit ermöglicht folgende Formel.

$$t(c) = \frac{c - c_n}{f_{\text{cnt}}} + t_n \quad (3.3)$$

Unter der Verwendung des Mittelwertes der Zählerfrequenz $\overline{f_{\text{cnt}}}$ und des Zählerwerts c_n beim letzten (n -ten) **PPS**-Signal sowie der **UTC**-Zeit t_n dieses Pulses können weitere Berechnungen vorgenommen werden. Für die Konversion von Zählerwerten in Absolutzeit kann mit der Standardabweichung der Zählerfrequenz $\sigma(f_{\text{cnt}})$ und der Standardabweichung der **PPS**-Signale $\sigma(t_{\text{PPS}})$ die Zeitunsicherheit $u_t(c)$ wie folgt abgeschätzt werden.

$$u_t(c) = \sqrt{\left[-\frac{c - c_n}{\overline{f_{\text{cnt}}}^2} \cdot \sigma(f_{\text{cnt}}) \right]^2 + \left[\sigma(t_{\text{PPS}}) \right]^2} \quad (3.4)$$

Informationen zur Standardabweichung der **PPS**-Signale $\sigma(t_{\text{PPS}})$ können von den in dieser Arbeit verwendeten **NEO-7M GNSS**-Modulen der Firma **U-Blox** nicht entnommen werden. Bei der Bestimmung der Zählerfrequenz nach Gleichung 3.2 werden Jitter und Drift des **PPS**-Signals und Zählerfrequenzabweichungen untrennbar miteinander vermischt. Daher kann als Näherung $\frac{\sigma(f_{\text{cnt}})}{\overline{f_{\text{cnt}}}} \cdot \Delta t_{\text{PPS}}$ mit $\Delta t_{\text{PPS}} = 1 \text{ s}$ als obere Abschätzung für die Unsicherheit der **PPS**-Signale verwendet werden. Es ergibt sich dann folgende Näherungsformel.

$$u_{t_{\text{UTC}}}(c) = \sqrt{\left[-\frac{c - c_n}{\overline{f_{\text{cnt}}}^2} \cdot \sigma(f_{\text{cnt}}) \right]^2 + \left[\frac{\sigma(f_{\text{cnt}})}{\overline{f_{\text{cnt}}}} \cdot \Delta t_{\text{PPS}} \right]^2} \quad (3.5)$$

In der **DAU** wird $\sigma(f_{\text{cnt}})$ aus einem 60 Werte großen Ringpuffer mit den letzten Werten der Zählerfrequenz nach Gleichung 3.2 berechnet. Nach jedem **PPS**-Signal wird der Ringpuffer aktualisiert und $\sigma(f_{\text{cnt}})$ neu berechnet. Für den Fall, dass die Taktquelle der **DAU** extrem stabil und rauscharm sein sollte, z. B. bei einem beheizten Quarzoszillatator (engl. **Oven-Controlled Crystal Oscillator, OC-XO**) oder der Referenztaktdistribution der **PTB**, bietet die Verwendung eines adaptiven Kalman-Filters zur Abschätzung des erwarteten Zählerstandes bei Sekundenbeginn, statt des letzten gemessenen Wertes, eine deutliche Verbesserung der Zeitstempelung [32].

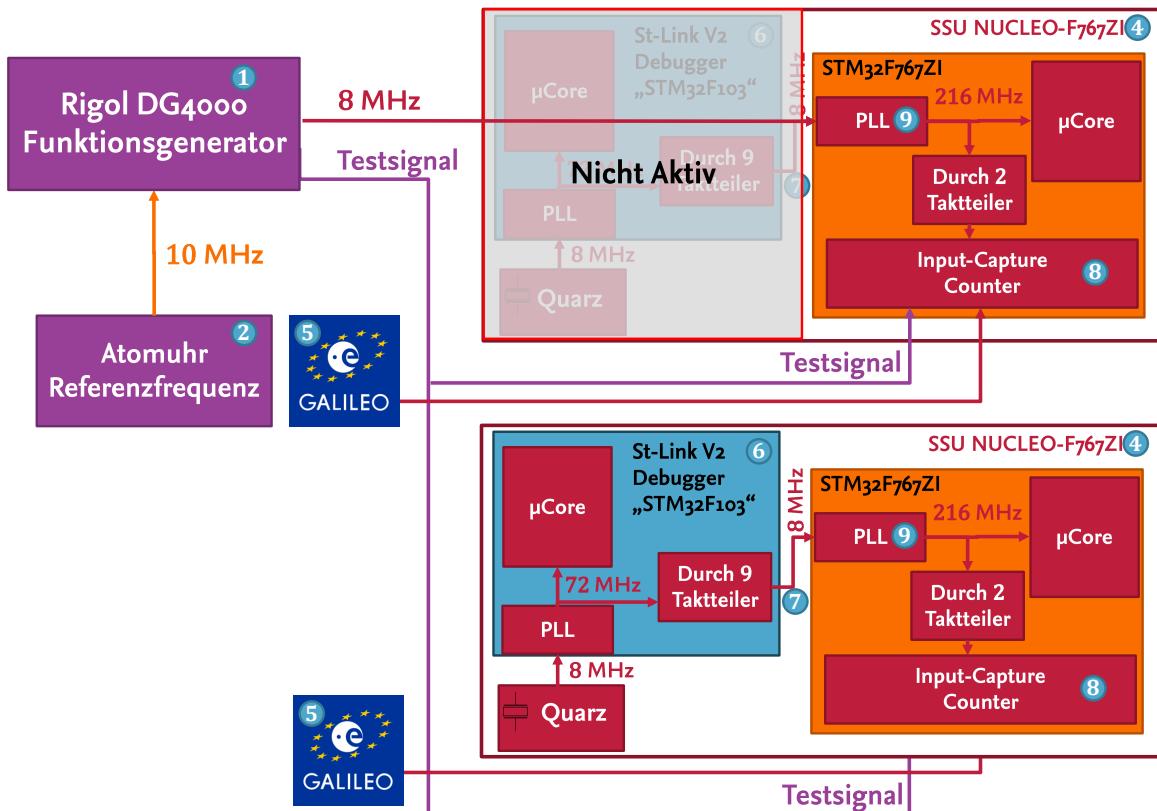


Abbildung 3.12: **Testaufbau für die Absolutzeitbasis.** Der *Rigol DG4162* Funktionsgenerator (1) wird von der 10-MHz-Normfrequenzdistribution (2) mit einem Referenztakt versorgt; der Generator erzeugt ein **LVTTL**-Testsignal mit der Frequenz 10 Hz und ein 8-MHz-Takt für eine der beiden DAUs (3). Die andere DAU (4) verwendet den 8-MHz-Taktausgang (7) des Debuggers (6) als Taktquelle für die Systemtakt-PLL (9). Jede DAU besitzt einen eigenen **GNSS**-Empfänger. Mittels **Input-Capture Counter** wird das 10-Hz-Signal gezeitstempelt.

3.3.1 Charakterisierung der GNSS-Zeitbasis

Um die Absolutzeitstempelung nach Gleichung 3.3 sowie die Unsicherheitsabschätzung nach Gleichung 3.5 zu validieren, wurden verschiedene Experimente durchgeführt. Die Abbildung 3.12 zeigt schematisch den Testaufbau für die Absolutzeitstempelung.

Mit einem Funktionsgenerator vom Typ *Rigol DG4162* wird ein **LVTTL**-Rechtecksignal mit einem Pegel von 3,3 V mit einer Frequenz von 10 Hz und einem Tastverhältnis von 50 % erzeugt. Die Zeitbasis des Generators wird mit den 10 MHz der **PTB**-Normalfrequenzdistribution synchronisiert. Die steigenden Flanken dieses Signals werden von zwei verschiedenen DAUs zeitgestempelt. Dabei verwendet eine DAU (unten in Abb. 3.12) den 8-MHz-Taktausgang des auf dem NUCLEO-Board integrierten Debuggers als **PLL**-Eingangstakt zur Erzeugung des Systemtakts von 216 MHz. Die andere DAU (oben in Abb. 3.12) wurde so modifiziert, dass ein externer Takt von 8 MHz als Referenz für die **PLL** der **CPU** verwendet wird. Dieses externe Taktsignal wird ebenfalls von dem Funktionsgenerator erzeugt. Der *Jitter* des 10-Hz-Signals wird im Wesentlichen durch den Funktionsgenerator verur-

sacht. Dabei ist die RMS-Abweichung des *Jitters* mit 500 ps spezifiziert. Der Funktionsgenerator ist ein auf **Direct Digital Synthesis (DDS)** basierender **Arbitrary Waveform Generator (AWG)**. Daher ist zu erwarten, dass der *Jitter* für Frequenzen, die ganzzahlige Teiler der **AWG**-Abtastfrequenz sind, niedriger ist [18]. Da der Generator auf die Normalfrequenzdistribution der **PTB** synchronisiert ist, wird angenommen, dass die aus der Frequenzdrift resultierenden *Jitter*-Komponenten vernachlässigbar klein sind. Die Einflüsse des *Jitters* des Signalgenerators auf die Zeitstempel sind somit deutlich kleiner als die Zeitquantisierung der **DAUs**.

Für die Validierung der Absolutzeitstempel wird die Abweichung $\Delta t(n)$ des Zeitstempels t_n vom theoretischen Wert $t_{\text{ITC}}(n)$ berechnet. Dabei werden die theoretischen Zeitstempel mit dem Akronym **ITC (Ideal Time Course)** bezeichnet, die Abweichung der Zeitstempel vom erwarteten Wert wird nach *IEEE STD 1139-2008* [1] als **Time Interval Error (TIE)** bezeichnet.

$$TIE = \Delta t(n) = t_n - t_0 - t_{\text{ITC}}(n) = t_n - t_0 - n \cdot \frac{1}{f_{\text{test}}} \quad (3.6)$$

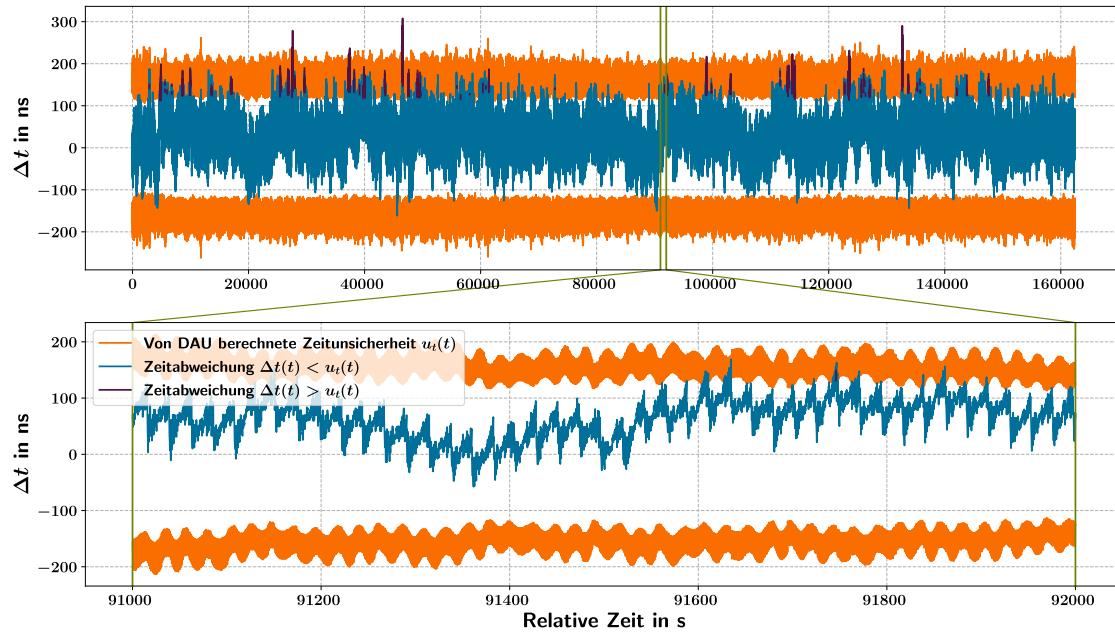
Zur Validierung der Unsicherheitsberechnung wurde der Anteil der von der **DAU** berechneten Unsicherheitswerte $u(n)$ bestimmt, für die die berechnete Unsicherheit größer oder gleich der Zeitabweichung vom **ITC** war: $\frac{|u(n)| \geq |\Delta t(n)|}{N}$, hierbei bezeichnet n den Index der Signalflanke und N die Gesamtzahl aller Signalflanken.

Die Abbildung 3.13a zeigt die Größen $\Delta t(n)$ (blau) und $u(n)$ (orange) der **DAU** mit dem *Debugger-Oszillat*or als Taktquelle. 99,5 % der von der **DAU** berechneten Unsicherheiten sind größer oder gleich der tatsächlichen Zeitabweichung. In der Vergrößerung ist eine periodische Modulation der Zeitstempel zu erkennen. Die Abbildung 3.13b zeigt dieselben Signale bei der Verwendung des vom Funktionsgenerator erzeugten 8-MHz-Takts als **PLL**-Referenztakt. In diesem Fall sind nur 71,5 % der von der **DAU** berechneten Unsicherheiten größer oder gleich der tatsächlichen Zeitabweichung, d. h. die Zeitunsicherheit wird unterschätzt.

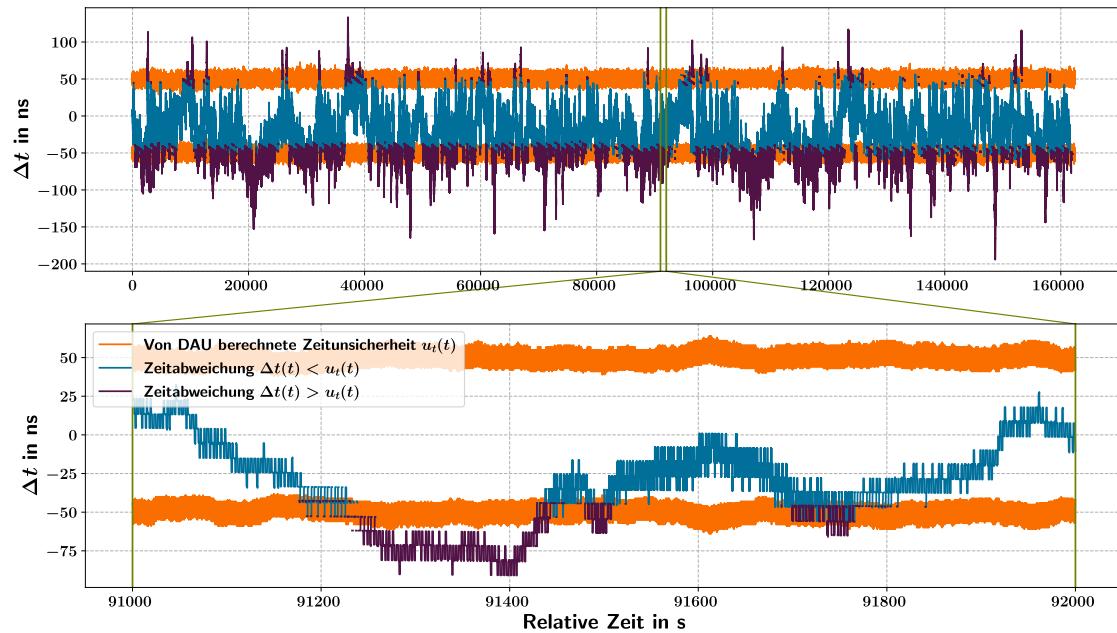
In der Vergrößerung sind sowohl die Zeitquantisierung durch den Zählertakt von $\frac{1}{108 \text{ MHz}} = 9,25 \text{ ns}$ als auch eine langsame Drift der Zeitabweichungen mit einer Periodendauer im Bereich einiger 100 s zu erkennen. Diese langsame Drift stammt nahezu vollständig aus dem **GNSS**-Modul. Der nur 60 Werte umfassende Pufferspeicher zur Berechnung von f_{cnt} und $\sigma(f_{\text{cnt}})$ ist zu klein, um diese Drift zu erfassen.

Deshalb wird die Unsicherheit nach Formel 3.6 deutlich unterschätzt. Bei einer zukünftigen Weiterentwicklung der **DAU** sollte für die Zählerstand-zu-Absolutzeit-Konversion und die Bestimmung der Unsicherheit ein adaptiver Kalman-Filter, wie in [50] vorgeschlagen, verwendet werden. Hiermit lässt sich mit einem guten Oszillatator eine bessere Schätzung für die Zählerstände an den **PPS**-Signalen berechnen, sodass der *Jitter* des **GNSS**-Moduls nicht mehr ungefiltert auf die Messwerte wirkt.

Zur weiteren Untersuchung der Zeitstempelung wurde die Zeitabweichung zwischen den Zeitstempeln der beiden **DAUs** im Versuchsaufbau nach Abb. 3.12 untersucht. Dabei wurde für jede steigende 10-Hz-Testsignalflanke von beiden **DAUs** die Absolutzeit bestimmt und die Differenz der Zeitstempel berechnet. Das Histogramm der Zeitstempeldifferenzen ist in Abbildung 3.14 gezeigt. Das Histogramm zeigt eine gaußförmige Wahrscheinlichkeitsdichteverteilung mit einem Mittelwert von $-20,72 \text{ ns}$ und einer Standardabweichung von $42,48 \text{ ns}$. Es sind deutliche Spalten im Abstand von 10 ns zu erkennen, welche durch die Zeitquantisierung auf ganze Ticks (9,25 ns) verursacht werden.



(a) Debugger-Oszillator als Taktquelle.



(b) Externer Oszillator als Taktquelle.

Abbildung 3.13: **Abweichung der Zeitstempel (TIE) vom idealen Verlauf.** Beide Grafiken (a, b) zeigen oben die Abweichung der Zeitstempel der steigenden Flanken des 10-Hz-Testsignals (siehe Abb. 3.12) vom erwarteten Wert (ITC) in ns (siehe Gleichung 3.6) und darunter jeweils eine Vergrößerung. Bei der Verwendung des internen Oszillators (a) ist in 99,5 % der Fälle die Zeitabweichung kleiner als die von der DAU berechnete Zeitunsicherheit. Die rampenförmige Frequenzmodulation, die durch die Wechselwirkung der PLL des Debuggers und der PLL der CPU zustande kommt, führt wegen der daraus resultierenden großen Streuung von f_{cnt} zu einer Überschätzung der Zeitabweichung.

Wird ein externer stabilisierter Takt verwendet (b), sind in nur 71,5 % der Fälle die Zeitabweichung kleiner als die von der DAU angegebene Zeitunsicherheit. Dies liegt daran, dass der GNSS-Jitter nicht gesondert erfasst wird; außerdem sind externer Takt und Testsignal sehr stark korreliert, da sie aus demselben Generator stammen. Die langsame Drift der GNSS-Zeitbasis wird durch die Mittelungszeit von 60 s nicht erfasst.

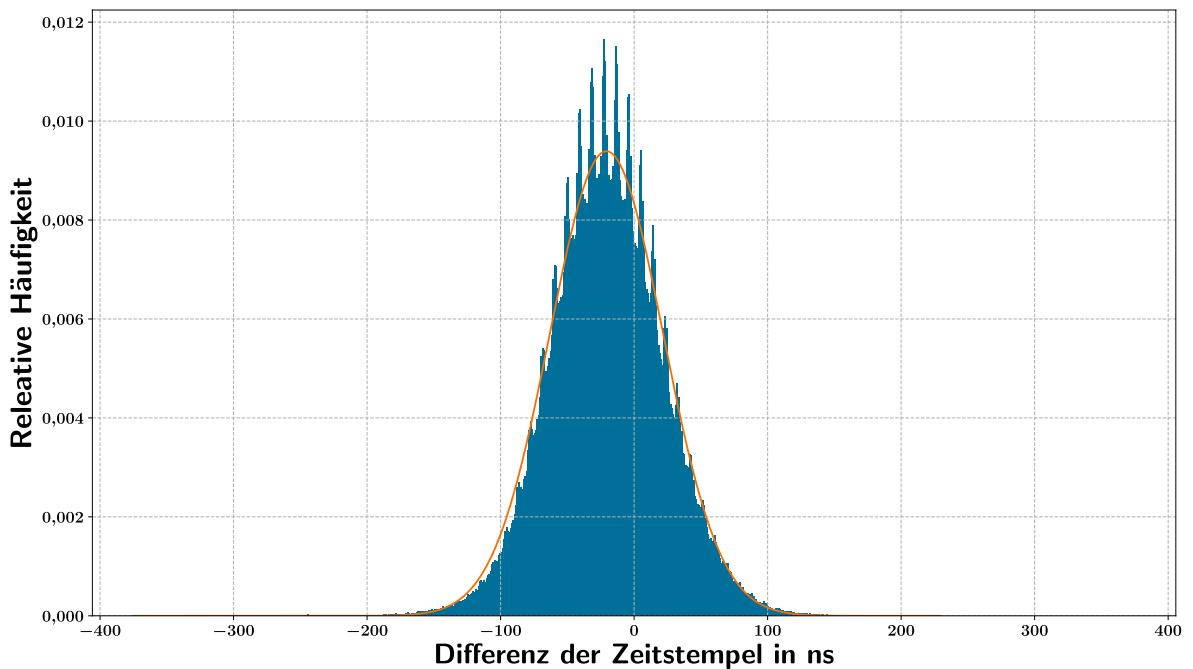


Abbildung 3.14: **Abweichung der zwei DAU-Zeitstempel voneinander.** Abweichung der Zeitstempel des 10-Hz-Signals zwischen den beiden DAUs im Aufbau nach Abb. 3.12. Es gibt eine konstante Abweichung von -21 ns sowie eine normal verteilte Streuung mit $\sigma = 43\text{ ns}$. Die Zeitquantisierungsauflösung von 10 ns ist erkennbar. Aus der Normalverteilung des Histogramms kann und darf eine spektral-weiße Frequenzverteilung der Zeitabweichungen nicht abgeleitet werden! Siehe auch Abb. 3.15.

Bei einer Verwendung der DAU in einem auf Absolutzeit basierenden Kalibriersystem sollte die systematische Zeitabweichung entsprechend in den Phasenberechnungen berücksichtigt werden. In der Abbildung 3.13 am deutlichsten in der Vergrößerung 3.13b ist zu erkennen, dass die Zeitabweichungen (TIE) zeitlich deutlich autokorriert sind. Zum Beispiel erzeugt die Kombination aus Debugger-PLL und PLL des STM32-Prozessors den sägezahnförmigen Verlauf der Zeitabweichungen in 3.15 oben. Der GNSS-Empfänger selbst erzeugt prinzipbedingt auch eine sägezahnförmige zeitlich autokorrierte Zeitabweichung vom idealen Wert [91], die aber eine deutlich größere Autokorrelationszeit als das der PLL-PLL-Kombination hat. Aus der gaußförmigen Verteilung der Zeitabweichungen im Histogramm (Abb. 3.14) kann auf Grund der zeitlichen Autokorrelation keine spektral-weiße Verteilung im Frequenzraum abgeleitet werden! Eine spektral-weiße Verteilung der Zeitabweichungen würde für die Betrachtung des Zeitstempelphasenrauschen (siehe Abschnitt 4.4) deutliche Vereinfachungen ermöglichen, da für spektral-weiße Zeitabweichungen eine Schätzung des Phasenrauschleistungsdichtspektrum aus der Standardabweichung des TIE einfach möglich ist. Um die sägezahnförmige Autokorrelation genauer zu untersuchen, wurde ein LVTTL-Signal mit einer Frequenz von 1 kHz von zwei unabhängigen DAUs mit jeweils eigenem GNSS-Modul und Verwendung des Debugger-Oszillators der DAU zeitgestempelt. Abbildung 3.15 zeigt oben in Blau und Orange die Zeitabweichungen bei der Zeitstempelung des Testsignals für beide DAUs. Diese zeigen jeweils die sägezahnförmige Autokorrelation in den TIE-Daten. Wenn Störungen im GNSS, z. B. Atmosphäreneinflüsse hierfür ursächlich wären, müssten diese Einflüsse bei beiden DAUs auftreten.

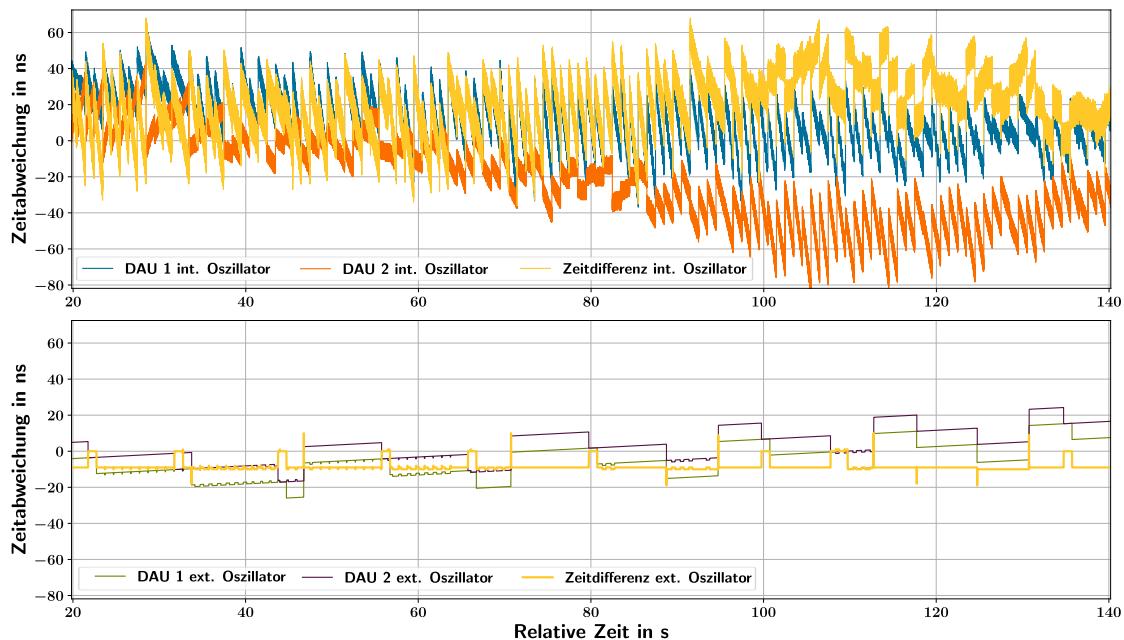


Abbildung 3.15: **Abweichung der Zeitstempel von zwei DAUs mit eigenem GNSS-Modul und internem Oszillator bzw. mit gemeinsamem GNSS-Modul und externem Oszillator.** In beiden Grafiken werden die Zeitabweichungen (TIE) bei der Zeitstempelung eines 1-kHz-Signals gezeigt. Oben verwenden die DAUs jeweils ihren Debugger-Oszillator und ein eigenes GNSS-Modul. Es ist eine sägezahnförmige zeitliche Autokorrelation in beiden Signalen erkennbar. Die Differenz der Zeitabweichungen (gelb) der beiden DAUs zeigt keine kleinere Varianz als die einzelnen Signale, die Zeitabweichungen sind folglich unkorreliert. Unten werden beide DAUs mit einem externen stabilen Takt von 8 MHz versorgt. Beide DAUs sind hier bei mit dem selben GNSS-Modul verbunden, es zeigt sich eine langsame Drift. Durch die starke Korrelation der Zeitbasen der DAUs fällt die Differenz der Zeitstempel (gelb) auf die Quantisierungsauflösung von 10 ns.

ten. Deshalb wurde zusätzlich die Differenz der Zeitstempel (gelb) berechnet. Die Standardabweichung der Differenz (gelb) ist dabei nur marginal kleiner als die der einzelnen TIE-Messungen. Folglich sind die Zeitabweichungen der DAUs untereinander fast unkorreliert. Die sägezahnförmigen Zeitabweichungen stammen daher vermutlich aus der PLL-PLL-Kombination des *Stm32-Nucleo*-Boards. Um diese Vermutung zu validieren, wurden beide DAUs mit demselben von der Referenztaktdistribution abgeleiteten Taktsignal von 8 MHz versorgt und über ein gemeinsames GNSS-Modul mit der Absolutzeit synchronisiert. Unten in Abbildung 3.15 sind in Grün und Violett die Zeitabweichungen der so modifizierten DAUs gezeigt. Anstatt der sägezahnförmigen Autokorrelation sind nun eine langsame Drift und Sprünge um die Quantisierungsauflösung von ≈ 10 ns erkennbar. Die Differenz der Zeitstempel untereinander (gelb) ergibt sich aus der Quantisierungsauflösung von ≈ 10 ns; sie beträgt im Mittel -10 ns. Die langsame Drift stammt vermutlich aus dem GNSS-Modul.

3.3.2 Analoge Phasenreferenz

Um die DAUs in ein bestehendes, nach ISO 16063-11 [3] akkreditiertes, analoges Kalibriersystem (ACS) integrieren zu können, wird die Phasenreferenz nicht über eine gemeinsame Absolutzeitbasis, sondern über ein analoges Referenzsignal hergestellt. Dieses analoge Referenzsignal (siehe Abbildung 5.2) wird sowohl von dem ACS als $y_{Sync_{ACS}}$ als auch von der DAU als $y_{Sync_{DAU}}$ erfasst. Die Digitalisierung kann sowohl von externen ADCs als auch vom internen ADC des STM32 übernommen werden, solange der Samplezeitpunkt genau bestimmt und extern getriggert werden kann. Dies ist unter anderem bei allen ADCs mit *Sample-and-Hold*-Schaltung der Fall. Abbildung 3.16 zeigt den internen Aufbau und die externe Beschaltung der ADCs der DAU. Als Phasenreferenz kommen alle zur Anregung phasensynchronen Signale in Frage. Dies sind zum Beispiel das Anregungssignal des Schwingerregers der HF-B-NME der PTB, oder ein von Signalgeneratoren erzeugtes Constant Output Level Amplitude (COLA)-Signal, wie bei der Winkelgeschwindigkeitskalibrierung an der PTB oder bei dem am CEM verwendeten Beschleunigungskalibriersystem CS-18 der Firma Spektra. Üblicherweise haben diese bipolaren Signale eine maximale Amplitude von $\pm 10\text{ V}$.

Der interne ADC des STM32 hat eine Auflösung von 12 Bit im Bereich zwischen $\text{GND}=0\text{ V}$ und der Analogreferenzspannung $\text{AV}_{CC} = 3,3\text{ V}$. Eingangsspannungen außerhalb dieses Bereichs können zur Zerstörung des ADCs oder des gesamten Controllers führen. Aus diesem Grund enthält die DAU eine analoge Signalkonditionierungsschaltung, die analoge Signale auf diesen Messbereich umsetzt. Die Eingangsschaltung und der ADC enthalten keinen Anti-Alias-Filter, um eine geringe Phasenverzögerung zu gewährleisten und eine Unterabtastung der periodischen Synchronisationssignale zu ermöglichen. Das durch den fehlenden Anti-Alias-Filter erhöhte Rauschen führt bei der Auswertung der schmalbandigen Synchronisationssignale mittels 3PSA nur zu geringen Fehlern. Eingangssignale werden mit einem differentiellen Vorverstärker ① um den Faktor 0,165 verstärkt, um ein Signal von 10 V zu $3,3\text{ V}$ konvertieren zu können. Anschließend wird ein konstanter Offset von $1,65\text{ V}$ addiert, um ein unipolares Signal für den ADC-Messbereich zu erhalten. Der Vorverstärker hat eine Eingangsimpedanz von $40\text{ k}\Omega$. Dieser Wert stellt einen Kompromiss aus geringer Belastung der im Allgemeinen niederimpedanten Signalquellen und wenig Eigenrauschen dar. Außerdem werden so parasitäre Ströme im Falle von unterschiedlichen GND-Potentialen auf ein unschädliches Maß begrenzt. Der Eingang des SAR-ADC ② ist im STM32 an einen Multiplexer mit *Sample-and-Hold*-Schaltung ④ angeschlossen. Da der Vorverstärker eine niedrige Ausgangsimpedanz hat und direkt an den ADC angeschlossen ist, kann eine kurze Samplezeit verwendet werden. Die Samplezeit muss aber ausreichen, damit die Signalquelle die Kapazität der *Sample-and-Hold*-Stufe auf die Eingangsspannung umladen kann. Je kleiner die Ausgangsimpedanz des Vorverstärkers, desto schneller ist dies möglich [94].

Der ADC in der DAU arbeitet mit einem Viertel des Systemtakts ($f_{ADC\text{Clock}} = 216\text{ MHz}/4 = 54\text{ MHz}$) und benötigt mindestens drei Taktzyklen $n_{\text{SampleCycle}}$ zum Stabilisieren der *Sample-and-Hold*-Stufe. Damit ergibt sich eine Abtastzeit $T_{\text{ADC}\text{Sample}}$ von

$$T_{\text{ADC}\text{Sample}} = \frac{n_{\text{SampleCycle}}}{f_{\text{ADC}\text{Clock}}} = \frac{3}{54\text{ MHz}} = 55,5\text{ ns} \quad (3.7)$$

Aus der Abtastzeit kann die maximal durch Unterabtastung erfassbare Signalfrequenz unter der

Annahme eines rechteckigen Abtastfensters $\text{rect}(\frac{t}{T_{\text{ADCsample}}})$ abgeschätzt werden, in dem die Fourier-Transformierte des Abtastpulses betrachtet wird. Die Fourier-Transformierte des Rechteckfensters ist eine Sinc-Funktion.

$$\mathfrak{F}\left(\text{rect}\left(\frac{t}{T_{\text{ADCsample}}}\right)\right) = \frac{1}{|T|} \cdot \text{sinc}\left(\frac{\omega}{2T}\right) \quad (3.8)$$

Zur theoretischen Abschätzung der maximalen Signalfrequenz wird oft der 3-dB-Punkt herangezogen. Dieser kann durch Lösen des Sinc-Terms in Gleichung 3.8 für die Magnitude an diesem Punkt von $\frac{1}{\sqrt{2}}$ berechnet werden. Hierbei ergibt sich eine Grenzfrequenz von

$$f_{\text{grenz}} = 1/(2 \cdot T_{\text{ADCsample}}) = \frac{1}{2 \cdot 55,5 \text{ ns}} = 9,01 \text{ MHz.} \quad (3.9)$$

Da die Grenzfrequenz der *Sample-and-Hold*-Stufe groß gegen die Signalfrequenz sein soll, sind Messungen bis etwa 2 MHz möglich. Die maximale Zeitunsicherheit der Absolutzeitstempelung liegt in der Größenordnung von 100 ns, was durch das Abtastphasenrauschen bei sehr hohen Frequenzen auch zu Signalstörungen führt. Da die Vorverstärkerschaltung für Frequenzen bis etwa 4 MHz ausgelegt ist, könnte der **ADC** bis etwa 2 MHz verwendet werden. Ab 100 kHz beginnt die Phasenstreuung deutlich anzusteigen, daher sollten nur niedrigere Frequenzen mit dieser Schaltung gemessen werden. Die AD-Konvertierung soll synchron zu den **Data Ready Interrupts** der digitalen Sensoren erfolgen. Der interne **ADC** des STM32 bietet einen Triggermultiplexer, bei dem auch ein externes Signal an einem **GPIO**-Pin als *Conversion-Trigger* ③ ausgewählt werden kann. Dieser **GPIO**-Pin wird parallel an den *Input-Capture* Eingang eines Timers gelegt. Je nach Konfiguration wird bei jeder steigenden oder fallenden Flanke des externen Triggers die AD-Konvertierung ausgelöst. Gleichzeitig wird ein Sprung in die **Interrupt Service Routine (ISR)** des *Timers* ausgelöst. In der **ISR** werden der Wert des Timers und die **DUT**-Sensordaten erfasst. Anschließend wird das Ergebnis der AD-Konvertierung aus den entsprechenden Registern im Mikrocontroller gelesen. Die Timerwerte werden später zu Absolutzeitstempeln konvertiert. Dies wird in Abschnitt 3.3 näher beschrieben.

3.3.3 Kalibrierung der analogen Phasenreferenz

Die Transferfunktion (insbesondere die Phasenverzögerung) des **ADCs** muss kalibriert werden, um die errechnete **DAU**-Phasenantwort nicht negativ zu beeinflussen. Da mit dem **ADC** die Phase eines monofrequenten Sinussignals gemessen werden soll, ist eine Unterabtastung möglich. Dies gilt für alle monofrequenten Signale mit der Frequenz f , die die Bedingung

$$f \neq \frac{N}{2} f_s \quad (3.10)$$

erfüllen; dabei sind N ein ganzzahliges Vielfaches und f_s die Abtastfrequenz. Im Falle $f = \frac{N}{2} \cdot f_s$ mit ungeradem N werden nur an zwei verschiedenen Phasenlagen die Amplituden der periodischen Sinusfunktion abgetastet, was für eine Rekonstruktion nicht ausreicht. Im Falle $f = N \cdot f_s$ wird immer die Amplitude an der gleichen Phasenlage des Sinussignals erfasst.

Die Abbildung 3.17 zeigt den **ADC**-Kalibrierungsaufbau. Die Kalibrierung des **ADC** der **DAU** ② wurde mit einem zweikanaligen Funktionsgenerator (Rigol DG 4062) durchgeführt. Ein PC ③ mit

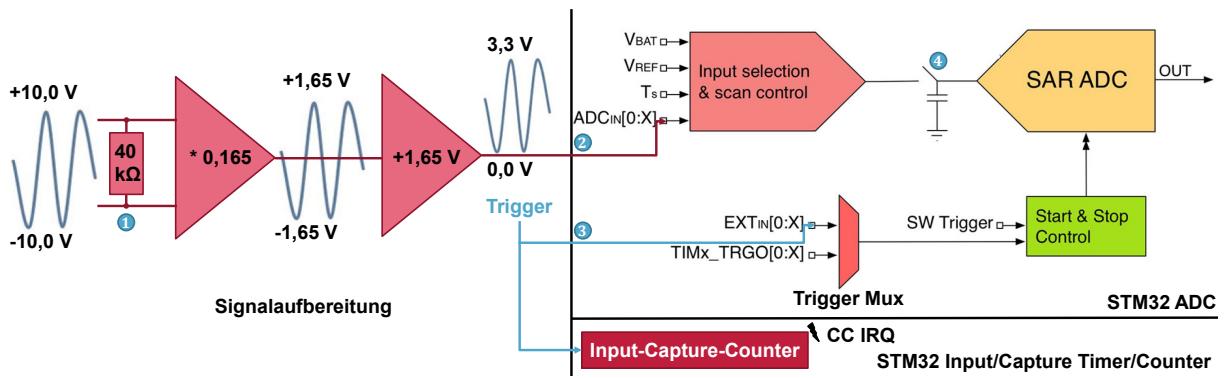


Abbildung 3.16: **Analoge Signalaufbereitung für den STM32-ADC.** Der Signalaufbereitungsverstärker ist differenziell mit einer Eingangsimpedanz von $40\text{ k}\Omega$ 1 ausgeführt und konvertiert das Eingangssignal von $\pm 10\text{ V}$ auf den Messbereichs (0 V bis 3,3 V) des ADC-Eingangs 2. Weder ADC noch Verstärkerschaltung beinhalten einen Anti-Alias-Filter. Der externe Trigger 3 des ADC ist zusätzlich mit einem Input-Capture-Eingang verbunden, um den Triggerzeitpunkt mit einem Absolutzeitstempel zu versehen. Durch die niedrige Ausgangsimpedanz des Verstärkers kann die Kapazität der Sample-and-Hold-Stufe 4 sehr schnell umgeladen werden, was Unterabtastung mit kurzen Abtastzeiten ermöglicht. Verändert und ergänzt aus [65].

Python-Software übernimmt die Steuerung des Kalibrierorgangs und die Erfassung und Auswertung der DAU-Messdaten. Es werden alle drei ADC-Kanäle gleichzeitig mit demselben Signal kalibriert.

Der Funktionsgenerator 1 erzeugt das Sinuskalibriersignal (blau) mit z. B. 1050 Hz und ein phasen-synchrones Trigger-Signal (rot) mit 999 Hz und n Pulsen. Bei einer Abtastfrequenz von 999 Hz ist für die Frequenzen nach ISO 226:1997 [47] eine Rekonstruktion nach Unterabtastung immer möglich, siehe Ungleichung 3.10. Durch eine Drei-Parameter-Sinus-Approximation mit der bekannten Anregungsfrequenz f_{cal} können die von den ADCs zu den Zeitpunkten t_{ADC} erfassten Werte $y[t_{\text{ADC}}]$ die Parameter der Funktion

$$y[t_{\text{ADC}}] \approx \hat{A}_{\text{ADC}} \cdot \sin(2\pi \cdot f_{\text{cal}} \cdot t_{\text{ADC}} + \varphi_{\text{ADC}}) + C_{\text{ADC}} \quad (3.11)$$

approximiert werden. Die Magnitudenantwort $|S(f_{\text{cal}})|$ kann mit der bekannten Anregungsspannung $\hat{A}_{\text{ADC}}(f_{\text{cal}})$ direkt als Quotient berechnet werden.

$$|S_{\text{DAUADC}}(f_{\text{cal}})| = \frac{\hat{A}_{\text{ADC}}(f_{\text{cal}})}{\hat{A}_{\text{ADCNominal}}(f_{\text{cal}})} \quad (3.12)$$

Aus der Phasenverschiebung $\varphi_{\text{Ref-Trigger}}$ zwischen Anregungssignal und Trigger und der Initialphase der ADC-Messung bezogen auf ihr erstes Sample kann direkt die Phasenantwort des ADC durch Subtraktion erhalten werden.

$$\Delta\varphi_{\text{DAUADC}}(f_{\text{cal}}) = \varphi_{\text{Ref-Trigger}}(f_{\text{cal}}) - \varphi_{\text{ADC}}(f_{\text{cal}}). \quad (3.13)$$

Abbildung 3.18b zeigt die Transferfunktion des ADC in dem für diese Arbeit relevanten Frequenz-

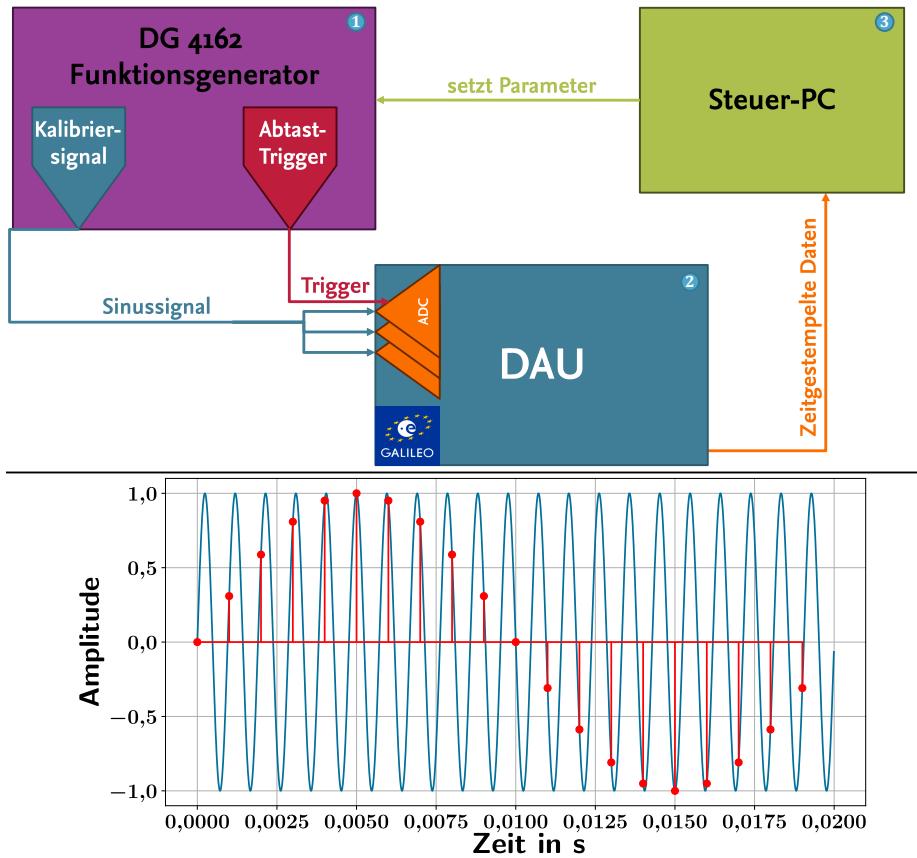


Abbildung 3.17: **Kalibrierung der analogen Phasenreferenz mittels Unterabtastung.** Der Funktionsgenerator ① erzeugt sowohl das analoge Sinussignal (blau), z. B. 1050 kHz ③, als auch den Abtasttrigger ② (hier 999 Hz). Durch das Unterabtasten wird das Signal an dem nächsten Vielfachen der Abtastfrequenz heruntergemischt (hier $1 \cdot 999$ kHz). So ergibt sich für das vom ADC erfasste Signal (rot) eine Frequenz von scheinbar ④ 51 Hz. Mittels 3PSA und den Absolutzeitstempeln kann das ursprüngliche Signal rekonstruiert werden. Ähnlich veröffentlicht in [86].

bereich bis 10 kHz. Kleinere Signalamplituden, z. B. 0,195 V (\approx 40 LSB) (blaue Transferfunktion Abb. 3.18), mit kleinerem SNR führen im gesamten Frequenzbereich zu einer Erhöhung der Phasenunsicherheit. Da für die 3PSA die Zeitstempel t_{ADC} verwendet werden, erzeugt der TIE der DAU-Zeitbasis einen Unsicherheitsbeitrag in dieser Kalibrierung. Da die tatsächlichen Abtastzeitpunkte bekannt sind, könnte dieser Beitrag kompensiert werden. Diese Kompensation ist bei realen Messungen aber nicht möglich und wurde daher hier nicht angewendet.

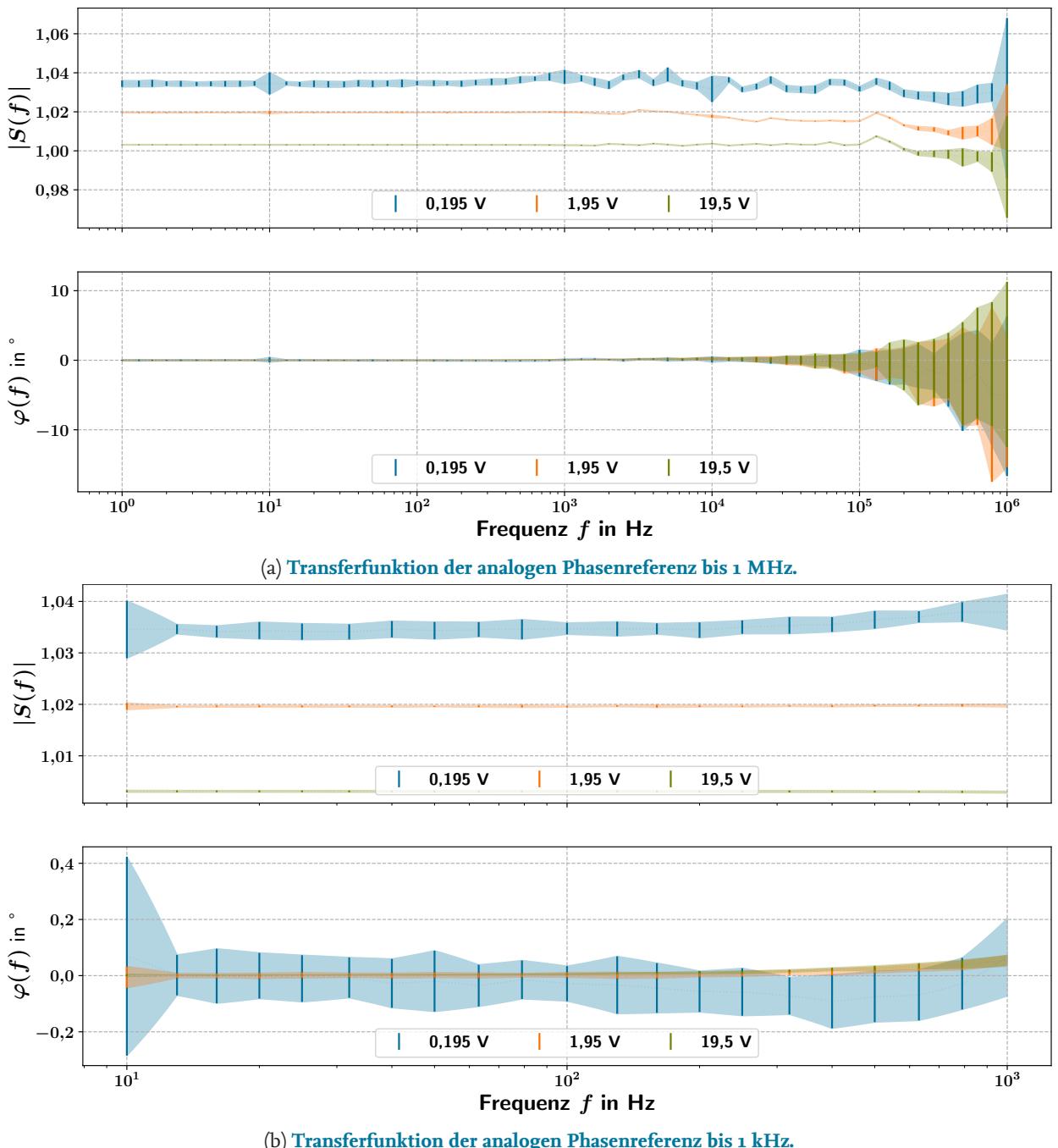


Abbildung 3.18: **Transferfunktion der analogen Phasenreferenz in Betrag und Phase.** Die Abbildung (b) zeigt den für diese Arbeit relevanten Frequenzbereich bis 1 kHz als Ausschnitt aus Abbildung (a). Die Messung erfolgte bei einer Abtastfrequenz von 999 Hz. Alle Signale über 499,5 Hz sind unterabgetastet. Zwischen den Messpunkten wurde linear interpoliert. Die angegebenen Unsicherheiten entsprechen 2σ für 10 Wiederholungsmessungen. Von den 4096 LSB des ADC werden bei 19,5 V ≈ 4000 LSB, bei 1,95 V ≈ 400 LSB und bei 0,195 V ≈ 40 LSB ausgesteuert. Die Abnahme von $|S(f)|$ bei höheren Anregungsamplituden wird vermutlich durch Nichtlinearitäten im Verstärker und ADC verursacht (siehe auch Abschnitt 7.2). Bei sehr hohen Frequenzen in Abb. (a) im Bereich ab 80 kHz kann für alle Amplituden eine Zunahme der Phasenunsicherheit beobachtet werden.

4 Nicht-äquidistantes Abtasten und seine Auswirkungen

In diesem Kapitel werden die Eigenschaften der **Drei-Parameter-Sinus-Approximation (3PSA)** im Frequenzraum erläutert. Die Verwendung sequenzieller 3PSAs zur Bestimmung der Signalfrequenz als nicht-iterative Alternative zur **Vier-Parameter-Sinus-Approximation (4PSA)** wird erklärt. Die verschiedenen Approximationsverfahren werden mit Hilfe von Simulationen verglichen, und die statistischen Eigenschaften ihrer Ergebnisse werden erläutert. Außerdem werden die Auswirkungen von Abtastzeitabweichungen beschrieben und simuliert.

4.1 Bandbreite der Drei-Parameter-Sinus-Approximation

Für die Sinus-Approximation werden immer nur Messwerte aus einem zeitlich begrenzten Beobachtungsfenster $g(t)$ mit der Dauer T_W und dem Startzeitpunkt t_0 verwendet. Das Beobachtungsfenster kann durch die Rechteckfunktion $\text{rect}(\frac{t}{a})$ der Breite a gemäß

$$\text{rect}\left(\frac{t}{a}\right) = \begin{cases} 0, & \text{wenn } |t| > \frac{a}{2} \\ \frac{1}{2}, & \text{wenn } |t| = \frac{a}{2} \\ 1, & \text{wenn } |t| < \frac{a}{2} \end{cases} \quad (4.1)$$

wie folgt beschrieben werden.

$$g(t) = \frac{1}{T_W} \text{rect}\left(\frac{t - t_0}{T_W}\right) = \begin{cases} 0, & \text{wenn } |t - t_0| > \frac{1}{2}T_W \\ \frac{1}{2T_W}, & \text{wenn } |t - t_0| = \frac{1}{2}T_W \\ \frac{1}{T_W}, & \text{wenn } |t - t_0| < \frac{1}{2}T_W \end{cases} \quad (4.2)$$

Die Fourier-Transformation dieser Fensterfunktion liefert folgende Funktion.

$$\mathcal{F}(g(t)) = G(f) = \underbrace{\text{sinc}(fT_W)}_{\text{Amplitudeneinhüllende}} \cdot \underbrace{e^{\frac{-i2\pi f(t_0 + \frac{T_W}{2})}{T_W}}}_{\text{Phasenverzögerung}} \quad (4.3)$$

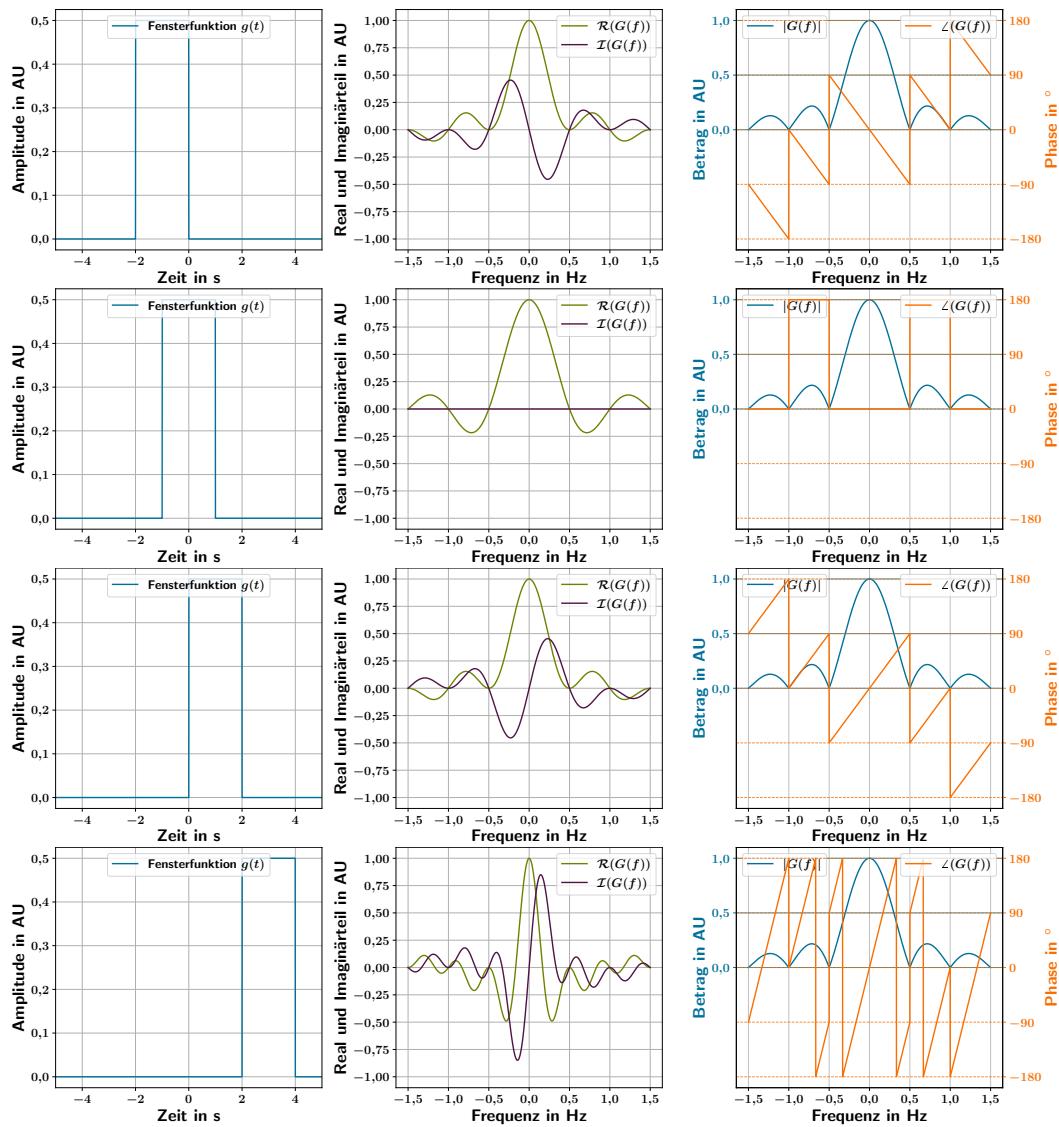


Abbildung 4.1: **Darstellung der Fourier-Transformation zeitlich verschobener rechteckiger Beobachtungsfenster.**

Die Startzeitpunkte sind $t_0 = [-2, -1, 0, 2]\text{s}$ bei einer Länge T_W von 2s. Das Rechteck-Fenster wird durch die Funktion $g(t)$ nach Gleichung 4.2 beschrieben. Die Fourier-Transformation liefert eine komplexe Sinc-Funktion (siehe Gleichung 4.4) mit den Nullstellen an den Frequenzen $n \frac{1}{T_W}$. Die Phase hängt linear von der Frequenz f und dem mittleren Fensterzeitpunkt \bar{t}_s ab (siehe Gleichung 4.4).

Die Abbildung 4.1 visualisiert die Funktionen im Zeit- und Frequenzbereich. Die Nullstellen der einhüllenden Sinc-Funktionen befinden sich an den Positionen $n \cdot \frac{1}{T_W}$ für $n \in \mathbb{Z} \neq 0$. Die 3-dB-Punkte (halbe Amplitude) des Betrags befinden sich an den Stellen $\pm \frac{0.44}{T_W}$. Für die Phase $\varphi(f)$ der Fourier-Transformation ergibt sich ein linearer Verlauf, der innerhalb der ersten Nullstellen der Sinc-Funktion $-\frac{1}{T_W} < f < \frac{1}{T_W}$ durch folgende Formel beschrieben werden kann.

$$\varphi(f) = \text{atan2}\left(e^{-i2\pi f(t_0 + \frac{T_W}{2})}\right) = \frac{f(t_0 + \frac{T_W}{2})}{T_W} = \frac{f\bar{t}_s}{T_W} \quad (4.4)$$

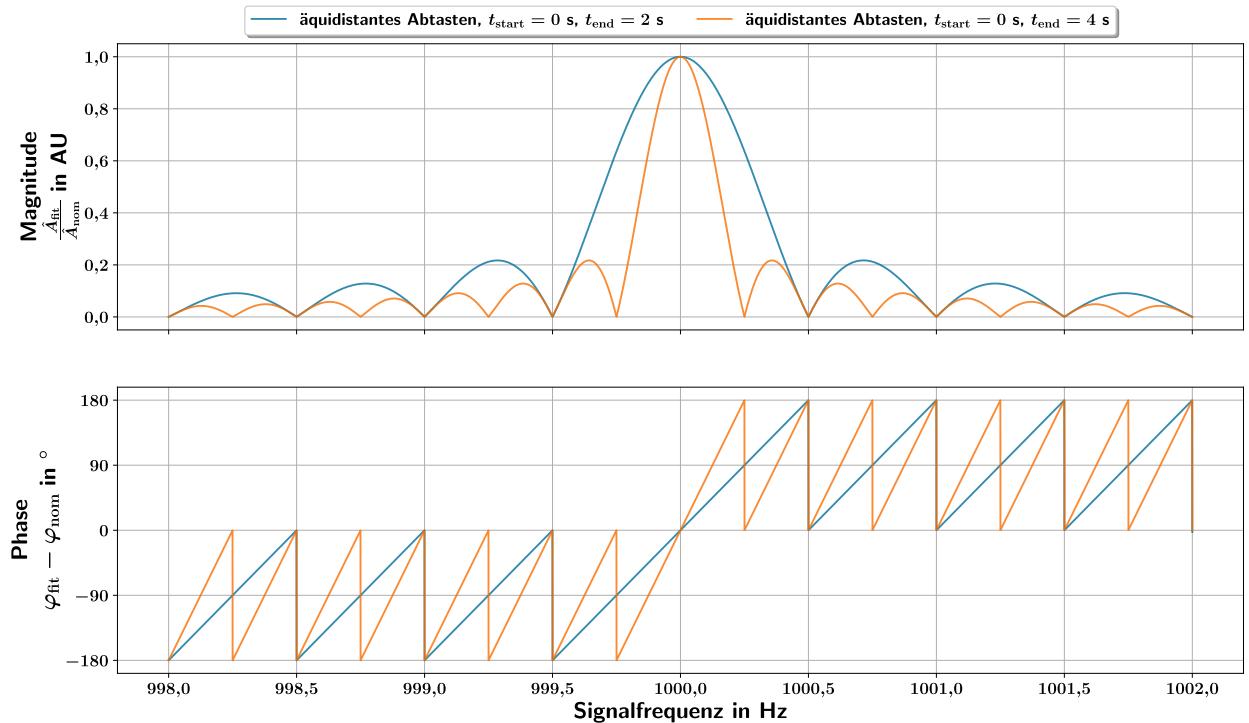


Abbildung 4.2: **Simulierte Übertragungsfunktion einer 3PSA mit verschiedenen Beobachtungslängen.** Die Übertragungsfunktion entspricht dem theoretischen Verlauf aus Gleichung 4.3.

Die Komponente $(t_0 + \frac{T_w}{2})$ im Exponenten des Phasenterms entspricht im diskreten Fall dem Mittelwert der Abtastzeitpunkte \bar{t}_s .

Nach dem Faltungstheorem entspricht die Multiplikation zweier Funktionen $g(t)$ und $h(t)$ im Zeitbereich einer Faltung (\circledast) im Frequenzbereich [105].

$$g \cdot h = \mathcal{F}^{-1}(\mathcal{F}(g) \circledast \mathcal{F}(h)) \quad (4.5)$$

Dies führt dazu, dass eine 3PSA, die auf Daten aus einem begrenzten Beobachtungszeitraum angewendet wird, eine Sensitivität und Phasenabweichung hat, welche dem in Gleichung 4.3 beschriebenen Verlauf um die Approximationsfrequenz folgt. Die Abbildung 4.2 zeigt eine Simulation zur Ver deutlichung dieses Effekts. Für Frequenzen zwischen 998 Hz bis 1002 Hz wurden numerisch Sinus Signale erzeugt. Die Amplitude \hat{A}_{nom} wurde auf 1 gesetzt, die Initialphase φ_{nom} auf 0° . Es wurden 1 s und 4 s lange Zeitabschnitte mit der Abtastrate $f_s = 1 \text{ kHz}$ simuliert, mit einer Länge von 1000 bzw. 4000 Werten. Die Zeitpunkte wurden äquidistant mit $\delta t = \frac{1}{f_s}$ gewählt. Für diese Zeitreihen wurden nun 3PSAs mit der Approximationsfrequenz $f_{\text{fit}} = 1000 \text{ Hz}$ berechnet. Die Abbildung 4.2 zeigt die Approximationsergebnisse. Der berechnete Magnitudenverlauf (oben in Abb. 4.2) entspricht exakt dem in Gleichung 4.3 analytisch bestimmten Verlauf für ein rechteckiges Beobachtungsfenster. Auch die Abweichungen der approximierten Phase φ_{fit} (unten in Abb. 4.2) vom nominalen Wert entsprechen genau dem durch die Faltung mit dem Rechteckfenster erwarteten Verlauf nach Gleichung 4.4. Werden die Messwerte nicht äquidistant, sondern gleichverteilt zufällig gewählt, ergibt sich ein abgesehen vom Quantisierungsrauschen der Numerik identischer Verlauf, siehe Abbildung 10.1 im Anhang.

4.2 Sequentielle Sinus-Approximation

In der Beschleunigungsmesstechnik werden, falls genug Messdaten zur Auswertung bereitstehen, sequentielle Sinus-Approximationen durchgeführt. Dabei wird die Zeitreihe in Segmente (engl. *Slices*) zerlegt und jeweils eine **3PSA** berechnet. Der Algorithmus der **3PSA**, nach Gleichung 2.27 wird explizit gelöst. Daher liefert das Verfahren immer eine Lösung, auch wenn keine Nutzsignalenergie in der Approximationsbandbreite (siehe Abb. 4.1) in den Messdaten enthalten ist.

Über die Dauer T_W der einzelnen *Slices* kann die Bandbreite der Approximation eingestellt werden. Dabei ist die Bandbreite

$$B(T_W) = \frac{1}{T_W}. \quad (4.6)$$

Die numerische Komplexität, d.h. der Rechenaufwand, der **3PSA** wird im Wesentlichen durch die Rechenzeit der notwendigen Matrixinversion bestimmt. Dabei liegt die Komplexität für diese Operation zwischen $O(n^3)$ für die Gauß-Jordan-Elimination und $O(n^{2.376})$ für den Coppersmith-Winograd-Algorithmus [21]. Die **FFT** kommt auf Grund der äquidistanten Werte und der dazu passenden Wahl einer Basis von orthonormierten Frequenzen ohne Matrixinversion aus, da die Inversion eine Einheitsmatrix zur Folge hätte, und durch die Wahl der Anzahl der Werte als 2^n kann sogar die Multiplikation mit der Design-Matrix **D** optimiert werden und eine Komplexität von lediglich $O(n \log(n))$ ist nötig. Bei der sequentiellen **3PSA** (abgekürzt **S3PSA**) müssen kleinere Matrizen mit insgesamt signifikant reduzierter Komplexität invertiert werden. Zusätzlich kann diese Aufgabe auch parallelisiert werden, da die einzelnen Approximationen unabhängig voneinander sind. Für äquidistantes Abtasten reicht sogar einmaliges Berechnen und Invertieren der Designmatrix aus. Aus den zirkulären Standardabweichungen (siehe Abschnitt 4.3) der approximierten Parameter (Amplitude und Phase) können Informationen über das **SNR** innerhalb der durch die Approximation abgedeckten Bandbreite gewonnen werden. Diese Information kann zur automatischen Qualitätsbestimmung einer Messung sowie als Grundlage für eine Typ-A-Unsicherheitsbetrachtung herangezogen werden. Während der Kalibrierungsmessungen kann es z. B. durch thermische Effekte zu leichten Amplitudenschwankungen des Anregungssignals kommen. Diese **Amplitudenmodulation (AM)** führt im Frequenzbereich zum Auftreten von Seitenbändern, welche Teile der Signalenergie enthalten. Bei der sequentiellen Sinusapproximation können diese Signalanteile durch geeignete Fensterlängen/Bandbreiten berücksichtigt werden. Da die **AM**-Seitenbänder symmetrisch zur Trägerfrequenz sind, haben sie aufgrund der entgegengesetzten Vorzeichen der Phasenabweichungen insgesamt keine Auswirkung auf die approximierte Phase.

4.2.1 Vier-Parameter-Sinus-Kalibrierung durch sequentielle Drei-Parameter-Sinus-Kalibrierung mit Phasenauswertung

Bei den in dieser Arbeit durchgeführten Messungen wurden relativ lange Messzeiten im Bereich einiger 10 s verwendet. Dies entspricht selbst bei der tiefsten betrachteten Frequenz von $f = 10$ Hz einigen 100 Schwingungsperioden. Sind viele Perioden an Messdaten vorhanden, bietet sich eine sequentielle **3PSA** aus den oben genannten Gründen an. Bei der Kalibrierung von digitalen Sensoren werden die Sensoren und Referenzdaten gegen unterschiedliche Zeitbasen erfasst. Diese Frequenzabweichungen führen zu kleinen Amplitudenabweichungen, aber großen Abweichungen des Phasenmittelwerts. Daher muss die vom **ACS** erzeugte Frequenz ebenfalls kalibriert werden [56].

Tabelle 4.1: **Vier-Parameter-Sinus-Approximation** durch sequentielle 3PSA

-
- a) Bestimmen der mittleren Samplefrequenz \bar{f}_s nach 4.7.
 - b) Zerteilen der Zeitreihe in S Slices mit je P Perioden und M Werten (siehe Gleichung 4.8).
 - c) Bestimmung der mittleren Slice-Zeitpunkte t_s nach Gleichung 4.9.
 - d) 3PSA für alle Slices mit der Startfrequenz ω_0 .
 - e) Phasensprungkorrektur des Initialphasen-Vektors $\vec{\varphi}$ mit `numpy.unwrap()` in Python.
 - f) Bestimmung der Frequenzabweichung $\Delta\omega$ (4.10) durch OLS
Python-Funktion (`numpy.polyfit`) von $\vec{\varphi}(\vec{t}_s)$.
 - g) Erneute S3PSA für alle Slices mit der Frequenz $\omega_0 + \Delta\omega$.
 - h) Bestimmen von Mittelwert und Kovarianz für die Parameter nach Gleichung 4.14.
-

Statt der langsamen 4PSA, denn jeder Iterationsschritt erfordert eine komplette 3PSA, kann auch die Änderung der Phase der Slices pro Zeit zur Bestimmung der Frequenzabweichung benutzt werden. Die Verwendung von Phasendetektion und Phasenakkumulation zur Erzeugung der Regelgröße einer PLL basiert auf demselben Prinzip [66]. Für die S3PSA wird die erfasste Zeitreihe y (mit je N Werten) in Slices mit mindestens einer Periode zerlegt. Da die Daten nicht notwendigerweise äquidistant abgetastet sind, wird zuerst aus den Abtastzeitpunkten $t_1, \dots, t_n, \dots, t_N$ die durchschnittliche Abtastfrequenz \bar{f}_s als arithmetisches Mittel der Zeit zwischen den Messwerten bestimmt

$$\bar{f}_s = \frac{1}{\frac{1}{N-1} \sum_{i=0}^{N-1} (t_{i+1} - t_i)} \quad (4.7)$$

Die Zahl der Messwerte M pro Slice mit P Perioden ergibt sich dann als

$$M = \lfloor \frac{\bar{f}_s}{f} \cdot P \rfloor, \quad (4.8)$$

wobei M zu einer ganzen Zahl gerundet wird. Jedem dieser S Slices wird der Mittelwert der Abtastzeitpunkte \bar{t}_s zugeordnet.

$$t_s = \frac{1}{M} \sum_{i=0}^M t_{(i+M \cdot S)} \quad (4.9)$$

Für jedes Slice wird nun eine 3PSA, wie im Abschnitt 2.4 beschrieben, durchgeführt. Weicht die Signalfrequenz von der Approximationsfrequenz ab, kommt es nach Gleichung 4.4 zu einer von der Frequenzdifferenz und dem Slice-Zeitpunkt t_s linear abhängigen Phasenverschiebung.

$$\frac{\Delta\varphi}{\Delta t} = \Delta\omega \quad (4.10)$$

Die Frequenzabweichung kann aus den Parametervektoren der S4PSA in Polarkoordinaten $\vec{\varphi} = (\varphi_{S0}, \varphi_{S1}, \dots, \varphi_{Ss})$ und $\vec{t} = (t_{S0}, t_{S1}, \dots, t_{Ss})$ direkt durch eine lineare Regression bestimmt werden. Dabei ist zu beachten, dass es in $\vec{\varphi}$ zu einem 2π -Phasensprung kommen kann, falls $\vec{\varphi}$ 180° über- oder unterschreitet. In diesem Fall muss dann sukzessive 2π addiert oder subtrahiert werden; die Funktion `numpy.unwrap()` bietet in Python diese Funktionalität.

Tabelle 4.1 fasst den Algorithmus zusammen.

4.2.2 Beispiel mit realen Messdaten

In Abbildung 4.3a sind die beim einem Experiment mittels S3PSA erhaltene Phasen φ als Funktion der Zeit aufgetragen. Während der Messdauer von 19 s wurden etwa 4750 Perioden des Signals mit der nominalen Frequenz von 250 Hz erfasst. Die Zeitreihe wurde in 475 *Slices* zerteilt; bei der Abtastrate von 1 kS/s enthält jedes *Slice* 40 Samples. Es ist sowohl im Sensorsignal (oben, grün Z-Beschleunigung) als auch in der analogen Phasenreferenz (unten, orange, Spannung CH 2) eine lineare Änderung der Phase um etwa 3° über den Beobachtungszeitraum zu erkennen. Die lineare Regression für die Daten der analogen Phasenreferenz liefert für die Steigung $\frac{\Delta\varphi}{\Delta t}$ den Wert $-0,003\,436\,67 \text{ rad/s}$. Dies entspricht einem Frequenzoffset Δf von $-0,5469 \text{ mHz}$ oder $2,1 \text{ ppm}$, was deutlich weniger als die in der ISO 16063 erlaubten 5 ppm ist. Im nächsten Schritt wurde eine S3PSA mit der korrigierten Frequenz von $f_{\text{corr}} = f_0 + \Delta f = 249,999453 \text{ Hz}$ durchgeführt. Abbildung 4.3b zeigt das Ergebnis der S4PSA mit korrigierter Frequenz. Es ist keine lineare Phasenabweichung vom Mittelwert mehr zu erkennen.

4.2.3 Limitierung des Verfahrens

Das im vorangegangenen Abschnitt 4.2.1 vorgestellte sequentielle Verfahren ist sehr schnell, naiv¹ parallelisierbar und liefert nach nur einer Iteration das optimale Ergebnis. Verglichen mit der 4PSA nach IEEE 1057 gibt es dennoch zwei zusätzliche Einschränkungen.

Erstens werden deutlich mehr Samples benötigt, da mindestens zwei halbe Perioden im Beobachtungszeitraum benötigt werden, verglichen mit nur einer halben Perioden für den 4PSA nach IEEE 1057.

Zweitens muss die Initialschätzung für f_0 deutlich weniger als $\Delta\omega_{\text{max}}$ vom tatsächlichen Wert abweichen.

$$\Delta f_{\text{max}} = \frac{1}{T_W} = \frac{1}{\Delta t_s} \quad (4.11)$$

Diese Limitierung lässt sich auf zwei verschiedene Weisen begründen.

Erstens, für größere Frequenzdifferenzen wäre die Phasenabweichung von einem *Slice* zum nächsten größer als π (siehe Abb. 4.2 und Gleichung 4.4). Eine Phasenentfaltung (*Phase-unwrap*) ist nicht mehr eindeutig möglich und die Steigung kann nicht mehr richtig bestimmt werden.

Zweitens, für größere Frequenzabweichungen nimmt die Sensitivität der 3PSA schnell ab, bis sie an der Stelle $\frac{1}{T_W}$ die erste Nullstelle erreicht. Das Signal-Rausch-Verhältnis wird dadurch verschlechtert. Bei zu niedrigem SNR kommt es zu Problemen bei der Korrektur von Phasensprüngen. Zeitlich konstante Störsignale werden im Gegensatz dazu nicht herausgemittelt. Wenn die Frequenzabweichung bekannt ist, sollte die Beobachtungsfensterlänge so gewählt werden, dass sich die gesuchte Frequenz innerhalb der 3-dB-Grenze der 3PSA befindet. Die Kovarianzen der sequentiellen 3PSA können nur bei gleichen Fensterlängen T_W und Gesamtmeessdauern sinnvoll verglichen werden, da das SNR der einzelnen 3PSA entscheidend von ihrer Bandbreite abhängt, schließlich integriert die 3PSA die Amplituden in der Beobachtungsbandsbreite.

¹Die Berechnung der einzelnen *Slices* ist völlig unabhängig und ohne gemeinsamen Speicherzugriff möglich.

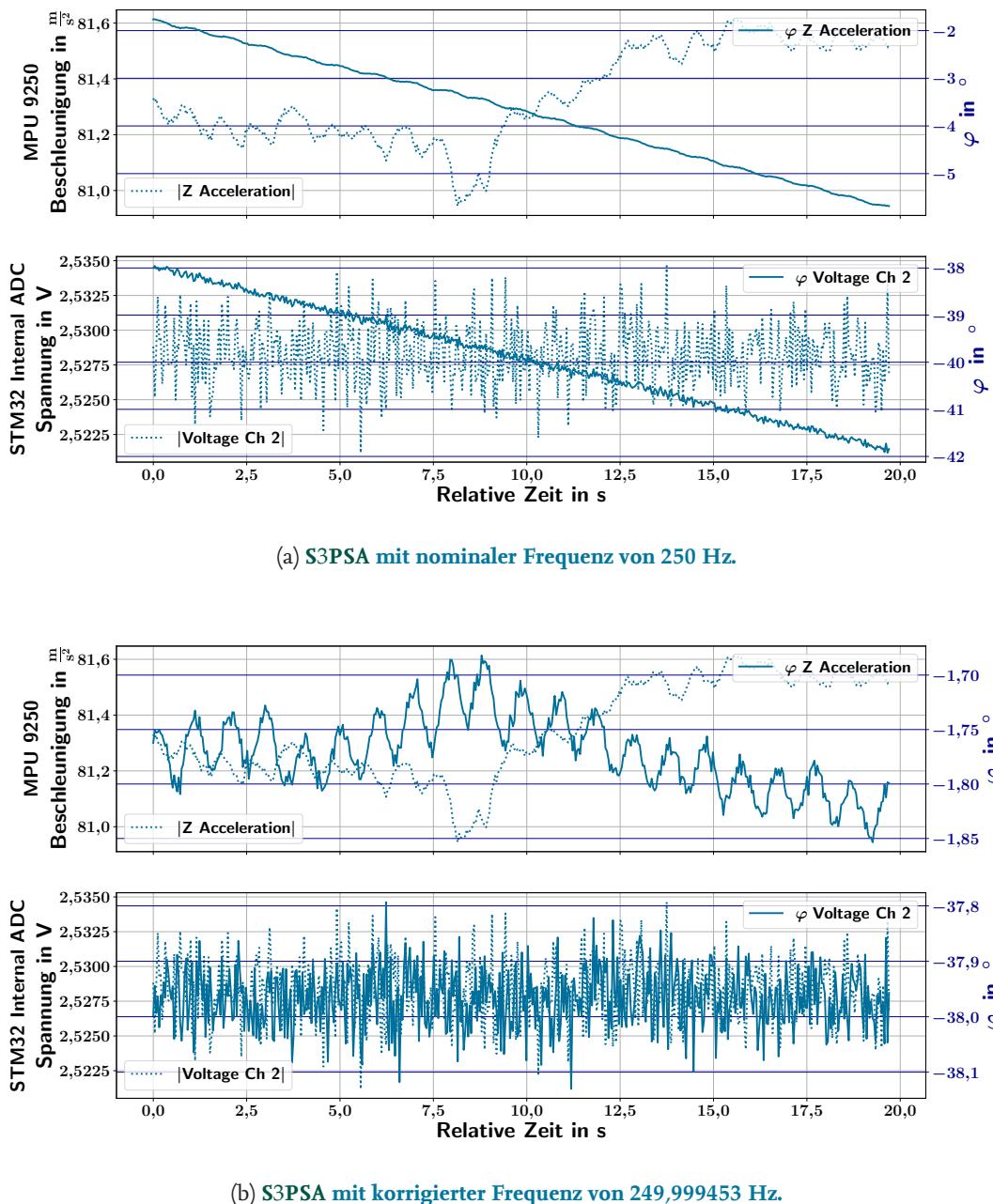


Abbildung 4.3: **Auswirkungen der Frequenzkorrektur auf die Phase der S3PSA.** Beide Grafiken zeigen die Ergebnisse einer S3PSA für jeweils 10 Perioden mit je 40 Punkten pro *Slice* einer Messung mit der MPU-9250 IMU bei einer Nominalfrequenz von 250 Hz. Wird die nominal Frequenz für die S3PSA verwendet wie in (a) zu sehen, zeigt sich sowohl für die Phase der Z-Beschleunigung als auch der Spannung des zweiten ADC-Kanal welcher das Phasenreferenzsignal erfasst eine starke zeitabhängige Phasenabweichung von 4° über die Messdauer von 20 s. Aus dieser Phasenabweichung wurde mittels linearer Regression die Steigung $\frac{\Delta\varphi}{\Delta t}$ als $-0,003\,436\,67 \text{ rad/s}$ bestimmt. Dies entspricht einer Frequenzabweichung Δf von $-546,9 \mu\text{Hz}$ oder $2,1 \text{ ppm}$. Die untere Abbildung (b) zeigt die Approximationsergebnisse mit der korrigierten Frequenz, die zeitabhängige Phasenabweichung ist nicht mehr vorhanden. Die nun erkennbaren periodischen Phasenabweichungen könnten z. B. durch Einkopplung von Obertönen des Stromnetzes stammen.

4.3 Statistische Momente der Approximationsergebnisse in polaren und kartesischen Koordinaten

Die 3PSA liefert die Koeffizienten $A + iB$ und den Gleichanteil D . In diesem Abschnitt soll an Hand von Beispielen auf die statistischen Eigenschaften der Approximationsergebnisse eingegangen werden.

4.3.1 Konversion in Polarkoordinaten

Zur Konversion von komplexen Zahlen in Polarkoordinaten werden die *Python*-Funktionen `abs(_nx.complexfloating)` und `numpy.angle(_nx.complexfloating)` für die komplexe Zahl $C = A + iB$ verwendet. Der Datentyp `_nx.complexfloating` ist eine komplexe Zahl aus zwei 64-Bit IEEE-754 Floatingpoint Werten [4, 40]. Die Amplitude ist

$$\hat{a} = \sqrt{(A^2 + B^2)} = |C|. \quad (4.12)$$

Die Initialphase φ wird mit der Funktion `numpy.angle(_nx.complexfloating)` bestimmt.

$$\varphi = \text{numpy.angle}(A + iB) = \begin{cases} \arctan\left(\frac{A}{B}\right) & \text{falls } B > 0 \\ \arctan\left(\frac{A}{B}\right) + \pi & \text{falls } B < 0 \text{ und } A \geq 0 \\ \arctan\left(\frac{A}{B}\right) - \pi & \text{falls } B < 0 \text{ und } A < 0 \\ +\frac{\pi}{2} & \text{falls } B = 0 \text{ und } A > 0 \\ -\frac{\pi}{2} & \text{falls } B = 0 \text{ und } A < 0 \\ 0 & \text{falls } B = 0 \text{ und } A = 0 \end{cases} \quad (4.13)$$

Bis auf den letzten Fall entspricht die Funktion `numpy.angle` der `atan2`-Operation. Diese liefert für den Nullvektor $A = 0$ und $B = 0$ aber einen ungültigen Wert (`NaN`) statt den Wert 0.

4.3.2 Mittelwert und Varianz

Der zirkuläre Mittelwert \bar{C} der S komplexen S3PSA-Ergebnisse $C = A + iB$ wird wie folgt berechnet.

$$\bar{C} = \underbrace{\left| \frac{1}{S} \sum_{j=1}^S |C_j| \right|}_{\text{Amplitude}} \cdot \underbrace{\frac{\sum_{j=1}^S \frac{C_j}{|C_j|}}{\left| \sum_{j=1}^S \frac{C_j}{|C_j|} \right|}}_{\text{Phase}} = \underbrace{\left| \frac{1}{S} \sum_{j=1}^S |C_j| \right|}_{\text{Amplitude}} \cdot \underbrace{\frac{1}{S} \sum_{j=1}^S e^{-i \arg(C_j)}}_{\text{Phase}} \quad (4.14)$$

Die Mittelwerte von Amplitude und Phase werden unabhängig voneinander berechnet, um final zu einer komplexen Zahl zusammengefasst zu werden. Für die Mittelwertbildung der Phase werden alle Werte normiert, um eine gleichmäßige Gewichtung zu erreichen.

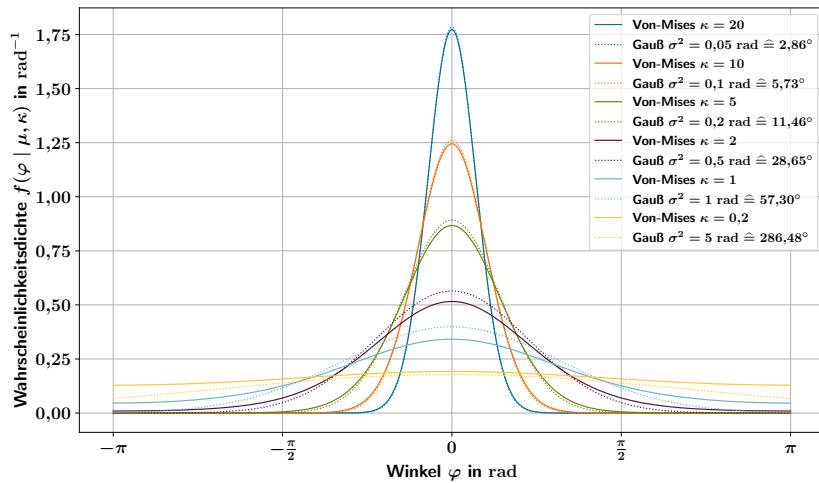


Abbildung 4.4: **Von-Mises-Verteilungsdichtefunktionen und genäherte Normalverteilungen.** Die Normalverteilungen wurden durch $\frac{1}{\kappa} \approx \sigma^2$ genähert. Die Näherung gilt nur für konzentrierte Verteilungen mit großem κ . Für kleine κ geht diese Verteilung in die Gleichverteilung mit einer Wahrscheinlichkeitsdichte von $f = \frac{1}{2\pi}$ über. Da die Von-Mises-Verteilung circulär ist befinden sich alle Werte immer im Bereich $[-\pi, \pi]$ es gilt also $\int_{-\pi}^{\pi} f(\varphi | \mu, \kappa) d\varphi = 1$.

Da der Wertebereich der Phase auf $-\pi$ bis π begrenzt ist, entstammen die approximierten Phasen keiner Normalverteilung sondern entstammen einer Von-Mises-Verteilung. Diese Verteilung ist zirkulär und nur für den Bereich $-\pi, \pi$ definiert. Die Dichtefunktion um die Lage des Maximums μ mit der Schärfe der Verteilung κ ist wie folgt definiert.

$$f(\varphi | \mu, \kappa) = \frac{e^{(\kappa \cos(\varphi - \mu))}}{2\pi I_0(\kappa)} \quad (4.15)$$

Dabei ist $I_0(\kappa)$ die modifizierte Besselfunktion erster Art. Die Abbildung 4.4 zeigt die Verteilungsdichtefunktion unterschiedlicher Von-Mises-Verteilungen und Normalverteilungen, deren Varianzen unter der Annahme $\frac{1}{\kappa} \approx \sigma^2$ genähert wurden. Für konzentrierte Verteilungen mit $\kappa \gg 5$ liefert diese Näherung gute Ergebnisse. Im Rahmen dieser Arbeit liegen die erwarteten Phasenstreuungen im Bereich um $0,5^\circ \approx 0,0087$ rad; dies entspricht einem $\kappa = \frac{1}{\sigma^2} \approx 13000$. Es kann also die Normalverteilung als Näherung für die Verteilungsfunktion der Phasenwerte verwendet werden. Bei der Berechnung der Phasenvarianzen wird nur die Differenz zum Mittelwert betrachtet, um den Sprungstellen bei $-\pi$ und π zu entgehen.

Die Python-Funktion `meanValueInPolarCoords(ABCwt)` im Listing 4.1 übernimmt die Berechnung von Mittelwerten und Varianzen/Kovarianzen entsprechend zu der Gleichung 4.14.

Die Berechnung des einfachen komplexen Mittelwerts nach

$$\bar{C} = \bar{A} + i\bar{B} \quad (4.16)$$

führt aufgrund der Unabhängigkeit von Phasen- und Amplitudenverteilungen und der daraus folgenden kreisbogenförmigen Verteilung in kartesischen Koordinaten zu einer systematisch zu kleinen Amplitude und wird in diesem Kapitel nur aus didaktischen Gründen erwähnt.

Code Listing 4.1: Mittelwertbildung in Polarkoordinaten in Python

```

1  def meanValueIn_PolarCoords(ABCwt):
2      DC = ABCwt[:, 2]
3      Complex = ABCwt[:, 1] + 1j * ABCwt[:, 0]
4      Freq = ABCwt[3]
5      lengthMean = np.mean(abs(Complex))
6      normalized = Complex / abs(Complex) # normalize vectors
7      radialCord = np.sum(normalized) / abs(np.sum(normalized)) # all vectors are ←
8          added to have one vector pointing in the direction of the mean value
9      deltaAng = np.angle(normalized) - np.angle(radialCord) # differences of the ←
10         angles, this value can be larger than -180 -- 180 deg
11      deltaAng=np.arctan2(np.sin(deltaAng), np.cos(deltaAng)) # map to -pi -->+pi ←
12          with atan2
13      meanValues = [ lengthMean,
14          np.mean(DC),
15          np.mean(Freq),
16          np.angle(radialCord),]
17      coVarData = np.cov(np.stack(
18          (
19              abs(Complex) - lengthMean, # subtracting mean, just for safty, this shouldnt ←
20                  change anything here
21              DC,
22              Freq,
23              deltaAng, # atan2 mapping is needed to deal with discontinuity at +-pi
24              ABCwt[:, 4]), # variance against time is correlated to residual phase due to ←
25                  frequency mismatch
26              axis=0),
27              bias=True) # bias=True normation With N like np.std
28      return meanValues,coVarData

```

4.3.3 Simulationen zur Mittelwertbildung und Streuung

Zur Veranschaulichung der Eigenschaften der Ergebnisse der sequentiellen Sinus-Approximationen und der Mittelwertberechnungen wurden für die vier Verfahren

- S3PSA ohne Frequenzkorrektur,
- S4PSA ohne Frequenzkorrektur,
- S3PSA mit Frequenzkorrektur durch $\frac{\Delta\varphi}{\Delta t}$,
- S3PSA mit Frequenzkorrektur durch 4PSA

entsprechende Simulationen durchgeführt. Die simulierten Werte $y(n)$ wurden mit den Parametern aus Tabelle 4.2 nach der folgenden Formel berechnet.

$$y(n) = \hat{a}_{\text{nom}} \sin(2\pi(f_0 + \Delta f) \cdot (\frac{1}{f_s}n + t_{\text{jitter}}(n)) + \varphi_0) + a_{\text{noise}}(n) \quad (4.17)$$

Tabelle 4.2: Übersicht der Simulationsparameter

Parameter	Symbol	Wert
Approximationsfrequenz	f_0	250 Hz
Frequenzabweichung	Δf	625 μ Hz (2,5 ppm)
Amplitude	\hat{a}_{nom}	1 m/s ²
Amplitudenrauschen	a_{noise}	0,01 m/s ²
Amplitudenverteilung		Normalverteilt
Phase	φ_0	45°
<i>Jitter</i> -Verteilungsbreite	a_{jitter}	2 μ s
<i>Jitter</i> -Verteilung		Gleichverteilt
Simulationsdauer		19 s
Abtastrate	f_s	1000/s
Perioden pro <i>Slice</i>	P	10
Gesamtzahl der <i>Slices</i>	S	450
Messwerte pro <i>Slice</i>	M	40

Dabei beschreiben die Funktionen $a_{\text{noise}}(n)$ und $t_{\text{jitter}}(n)$ Rauschquellen, die normales beziehungsweise gleichverteiltes mittelwertfreies Rauschen mit definierter Standardabweichung respektive Verteilungsbreite vorgeben. Die Simulationsparameter wurden entsprechend der realen Messung in Abbildung 4.3a bzw. 4.3b gewählt. Die Annahme, dass die Abtastzeitabweichungen (also der *Jitter*) gleichverteilt und zeitlich unkorreliert seien, ist in der Realität nicht erfüllt; für die Untersuchung der Eigenschaften der Approximationsmethoden ist sie dennoch zweckmäßig.

Die Ergebnisse der Sinus-Approximationen sind in den Abbildungen 4.5 bis 4.8 dargestellt. Die runden teiltransparenten Punkte in den Teilgrafiken zeigen jeweils die Approximationsergebnisse für einen *Slice* der sequentiellen Approximation. Die mittlere Zeit des *Slices* t_s ist durch eine Farbkodierung visualisiert.

Die jeweils linke Teilgrafik zeigt die Approximationsergebnisse in einem kartesischen Koordinatensystem mit Realteil **A** und Imaginärteil **B**. Die 3PSA liefert direkt diese komplexen Werte. Die jeweils rechte Grafik zeigt auf der Y-Achse den Betrag bzw. die Amplitude \hat{a} der Approximationsergebnisse und auf der X-Achse die dazugehörigen Phasenwerte φ .

In beiden Grafiken sind als oranger Stern (★) der Nominalwert sowie die Mittelwerte für die Mittelwertbildungsverfahren dargestellt, in Rot (★) der Mittelwert der komplexen Komponenten und in Grün der (★) Mittelwert von Amplitude und Phase nach Gleichung 4.14. Für das zum Koordinatensystem passende Mittelwertbildungsverfahren sind die korrespondierenden 1σ - und 2σ -Konfidenzintervalle als Ellipsen eingezeichnet. Die Ausrichtung und Größe der Ellipsen wird dabei aus den Eigenvektoren der Kovarianzmatrix bestimmt, die Halbmesser entsprechend aus den Eigenwerten. Der verwendete Programmcode findet sich im Listing 10.1.

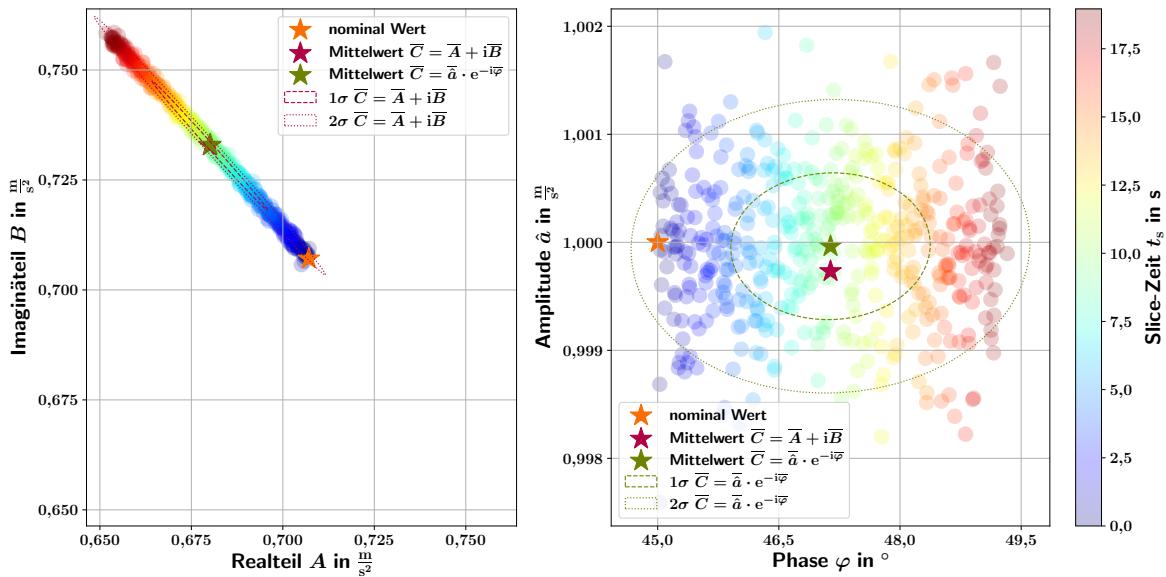


Abbildung 4.5: **Simulation einer S3PSA ohne Frequenzkorrektur.** Es ist eine starke zeitliche Korrelation erkennbar; die Phasenwerte der frühen Slices (blau) weisen nur eine geringe Phasenabweichung zum Nominalwert von 45° auf, die späten Slices (rot) weichen um fast 4° vom Nominalwert ab.

Die Abbildung 4.5 zeigt eine Simulation für die nichtfrequenzkorrigierte 3PSA. In beiden Darstellungen (links kartesisch, rechts polar) ist die starke zeitliche Korrelation der Phase wegen der Approximationsfrequenzabweichung erkennbar (siehe Gleichung 3.8). In der komplexwertigen Darstellung ist eine starke negative Korrelation zwischen Real- und Imaginärteil zu erkennen. Der Korrelationskoeffizient beträgt in diesem Fall $r_{AB} = -0,997$. Bei genauerer Betrachtung und bei Vergleich mit der Darstellung in Polarkoordinaten ist zu erkennen, dass die Approximationsergebnisse auf einer Kreisbahn mit dem Radius $\hat{a}_{\text{fit}} = \hat{a}_{\text{nom}} = 1,0$ liegen. Abbildung 10.2 im Anhang zeigt gut erkennbar die Kreisbahn bei einer Simulation mit der zehnfachen Dauer von 190 s und sonst identischen Parametern. Die bogenförmige Verteilung in kartesischen Koordinaten kann durch die Kovarianzmatrix und die daraus berechneten Ellipsen nicht korrekt wiedergegeben werden. Bei der Berechnung des Mittelwerts aus A und B (★) kommt es auf Grund der großen Phasenstreuung erwartungsgemäß zu einer systematischen Verringerung der Amplitude.

In der Polardarstellung (rechts) ist, abgesehen von der zeitlichen Korrelation, keine Korrelation zwischen Amplitude und Phase erkennbar. Der Korrelationskoeffizient $r_{\hat{a}\varphi}$ für diese Simulation ist $r_{\hat{a}\varphi} = 0,0023$. Bei wiederholten Simulationen ist der Korrelationskoeffizient im Mittel $\bar{r}_{\hat{a}\varphi} = 0,0$. Die Amplitude und Phase sind folglich vollständig unkorreliert!

Der Mittelwert der Phase weicht bei beiden Mittelungsverfahren erwartungsgemäß um

$$\Delta\varphi = \Delta f \cdot \bar{t}_s = 625 \mu\text{Hz} \cdot 9,5 \text{ s} = 0,0059 \text{ rad} = 2,1375^\circ \quad (4.18)$$

vom Nominalwert ab.

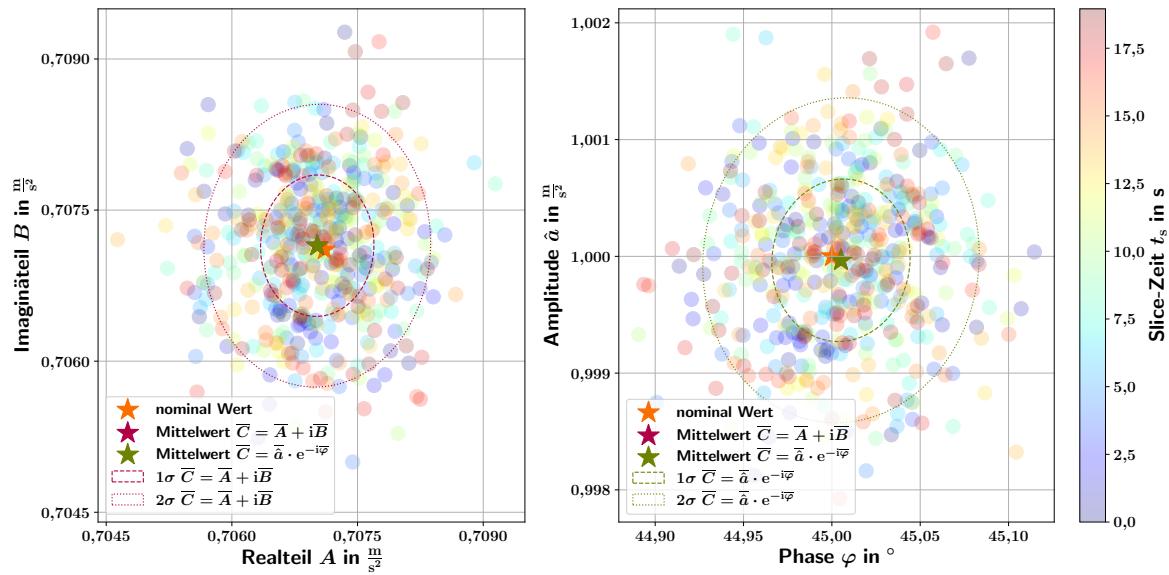
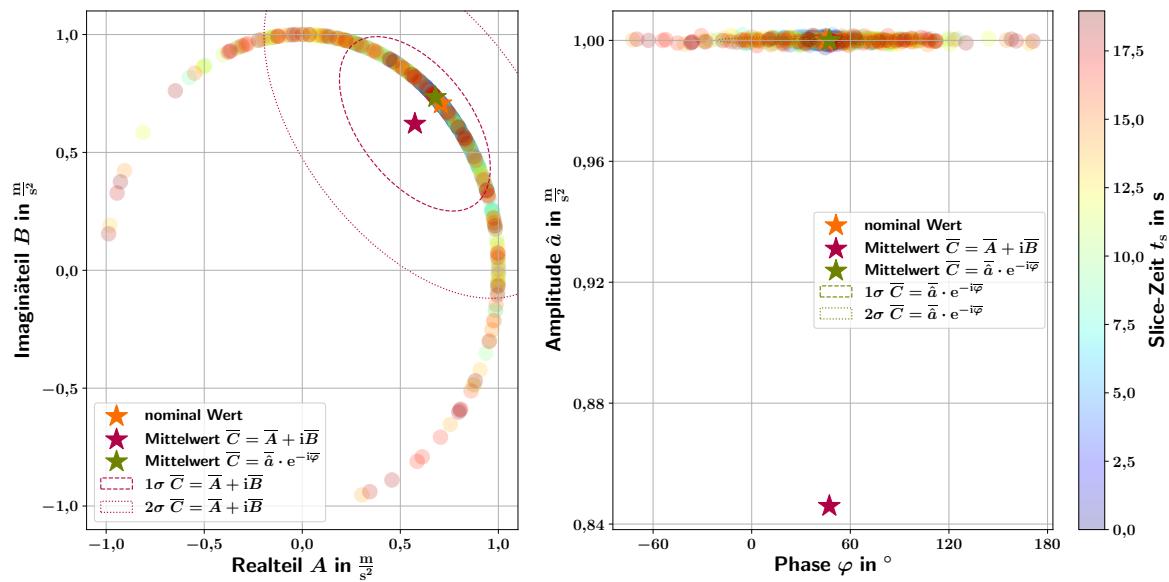

 Abbildung 4.6: Simulation einer S3PSA mit Frequenzkorrektur mit $\frac{\Delta\varphi}{\Delta t}$.


Abbildung 4.7: Simulation einer S4PSA ohne Frequenzkorrektur.

Die Abbildung 4.6 zeigt die Approximationsergebnisse für die S3PSA nach Frequenzkorrektur mit $\frac{\Delta\varphi}{\Delta t}$.

Es sind in beiden Darstellungen keine nennenswerten Korrelationen erkennbar ($r_{AB} = -0,078$, $r_{\hat{a}\varphi} = 5,04 \cdot 10^{-5}$). Für wiederholte Simulationen streuen die Approximationsergebnisse um den nominalen Wert. Die Varianzen bzw. Standardabweichungen werden durch die Länge der *Slices* bestimmt. Längere *Slices* haben eine geringere Bandbreite, dadurch ein besseres **SNR** und somit eine niedrigere Streuung. Für vollständig unkorrelierte, zufällige Rauschquellen hat die *slice*-Länge keinen Einfluss auf die Varianz des Mittelwerts aller *Slices*.

Die Abbildung 4.7 zeigt die Ergebnisse der S4PSA. Es ist eine sehr starke Streuung der Phasenwerte

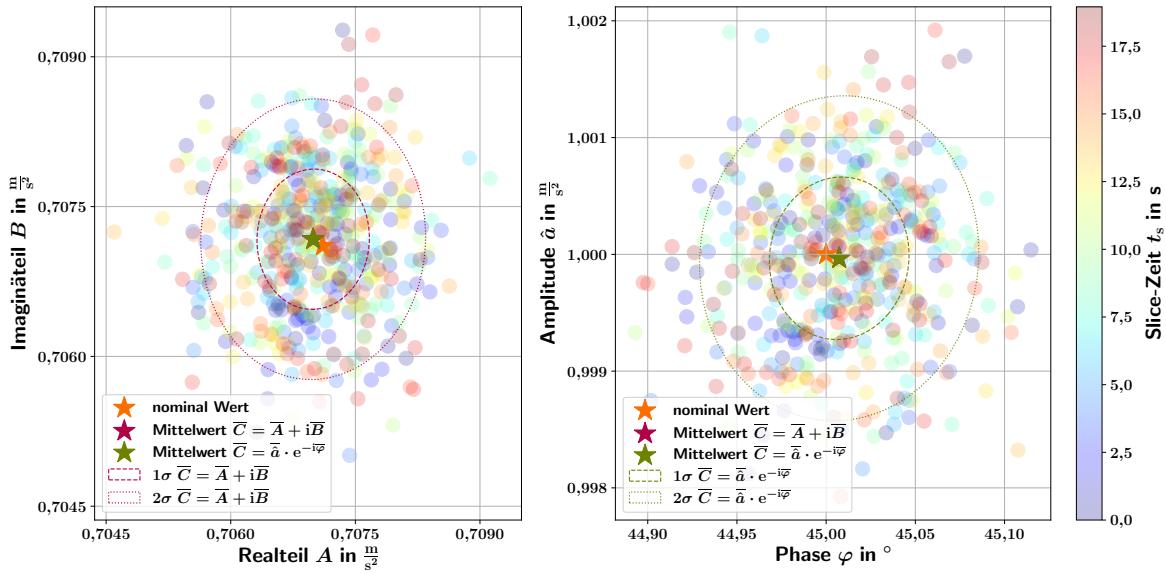


Abbildung 4.8: **Simulation einer S3PSA nach Frequenzkorrektur durch 4PSA.** Das Ergebnis entspricht einer Frequenzkorrektur mit $\frac{\Delta\varphi}{\delta t}$.

erkennbar. Die approximierten Phasen und Frequenzen (hier nicht gezeigt) sind stark korreliert. Die Laufzeit $\tau = \varphi f$ wird korrekt approximiert; die Werte für f und korrespondierend für φ streuen dabei normalverteilt.

Eine statistische Auswertung in kartesischen Koordinaten ist aufgrund der starken Phasenstreuung nicht sinnvoll möglich. Deshalb wird die Amplitude des komplexen Mittelwerts (★) um 15 % zu niedrig bestimmt.

In Polarkoordinaten hingegen ist eine Bestimmung der Amplitude möglich. Der Amplitudenmittelwert weicht um deutlich weniger als 0,05 % vom Nominalwert ab, siehe Abbildung 4.7. Auch für dieses Approximationsverfahren sind die Amplitude und Phase nahezu unkorreliert ($r_{\hat{a}\varphi} = -0,05058388$). Die Abbildung 4.8 zeigt die Ergebnisse einer S3PSA mit Frequenzkorrektur durch 4PSA (nicht sequentiell). Die Ergebnisse dieses Verfahrens sind äquivalent zu denen der S3PSA nach Frequenzkorrektur durch $\frac{\Delta\varphi}{\Delta t}$. **Fazit:** Die Mittlung in Polarkoordinaten (★) nach Gleichung 4.14 zeigt im Gegensatz zu der Mittelung in komplexen Koordinaten (★) keine systematischen Abweichungen. Außerdem sind in Polarkoordinaten Amplitude und Phase unkorreliert, und auf eine Betrachtung der Kovarianz kann verzichtet werden. Als nichtiteratives, naiv parallelisierbares Verfahren wird in dieser Arbeit die S3PSA nach Frequenzkorrektur mit $\frac{\Delta\varphi}{\Delta t}$ und Mittelwertbildung in Polarkoordinaten nach Gleichung 4.14 verwendet.

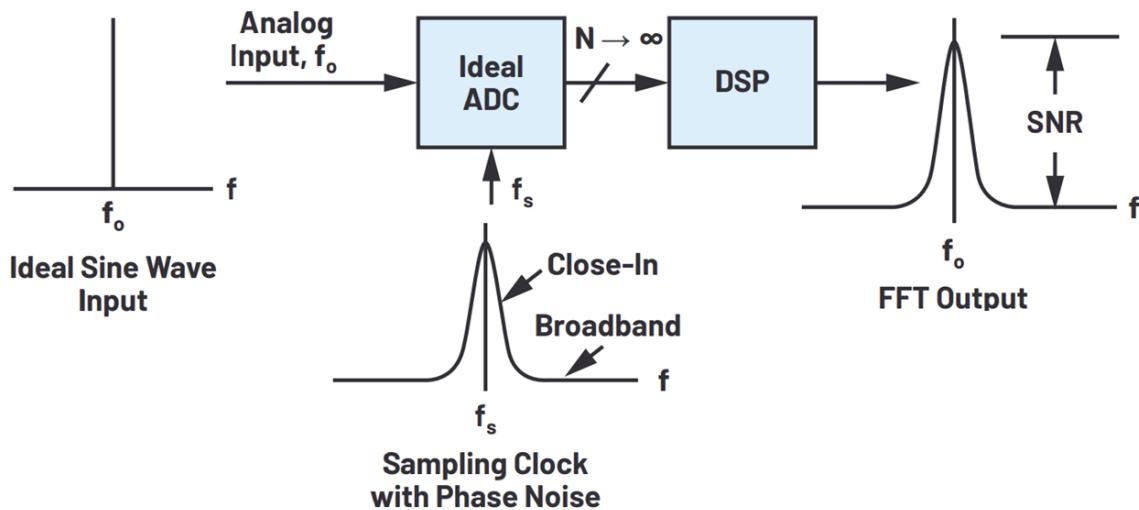


Abbildung 4.9: **Auswirkung von Abtastphasenrauschen.** Ein ideales monofrequentes Sinussignal wird mit einem idealen ADC abgetastet. Durch das Phasenrauschen des Abtasttaktes kommt es zu einer Verbreiterung des Sinussignals im Spektralbereich. Das breitbandige Phasenrauschen begrenzt das maximal mögliche SNR. Verändert aus [53].

4.4 Jitter und Phasenrauschen

In vielen Bereichen der digitalen Signalverarbeitung werden äquidistant abgetastete Messwerte verarbeitet. In der Realität kommt es immer zu Abweichungen der Abtastzeitpunkte. Diese Abweichungen werden als *Sampling Jitter* bezeichnet. Eine einfache Beschreibung des *Jitters* ermöglicht die Angabe der Standardabweichung der Abtastzeitpunkte. Dies wird als *RMS-Jitter-Amplitude* bezeichnet. Da die Zeitabweichungen bzw. die Phasenlage untereinander zeitlich korreliert sind, ist die Betrachtung der *RMS-Jitter-Amplitude* nicht mehr ausreichend, und eine Betrachtung der Phasenrauschleistungsdichte $\mathcal{L}(f)$ im Frequenzraum ist nötig [1, 22, 54, 38, 25, 90]. Bei der Abtastung eines Signals mit periodischen Abtastpulsen wird das Signal im Frequenzbereich mit der Fourier-Transformierten der Abtastpulsequenz gefaltet. Für den Fall von perfekt äquidistanten Dirac-Pulsen als Abtastfunktion ergeben sich im Frequenzbereich wieder Dirac-Pulse im Abstand der Abtastfrequenz. Eine Faltung mit diesen Dirac-Pulsen beeinflusst nach dem Nyquist-Shannon-Abtasttheorem ein Nutzsignal mit einer Bandbreite kleiner als die halbe Abtastfrequenz nicht. Sind die Abtastpulse durch *Jitter* bzw. Abtastphasenrauschen nicht äquidistant, ist die Fourier-Transformierte dieser Abtastpulse kein Dirac-Puls mehr, sondern eine kontinuierliche Funktion. Die Spektralrauschleistungsdichte um den verbreiterten Dirac-Puls der mittleren Abtastrate \bar{f}_s entspricht dabei dem Phasenrauschleistungsdichtespektrum $\mathcal{L}(f)$, dabei bezeichnet f den Abstand zu \bar{f}_s . Abbildung 4.9 zeigt die Auswirkung von Abtastphasenrauschen auf die Erfassung eines idealen Sinussignals, welches anschließend Fourier-transformiert wird. In erster Näherung kann die Phasenrauschleistungsdichte durch zwei Funktionsteile beschrieben werden.

In niedrigen Abstand zur Abtastfrequenz (Träger) befindet sich das sogenannte *Close-In*-Phasenrauschen oder auch *Narrow-Band*-Phasenrauschen, welches durch korrelierte Zeitabweichungen verursacht wird. Das einfache *Leeson*-Modell modelliert die *Close-In*-Komponente des Phasenrauschens durch zu $\frac{1}{f^3}$, $\frac{1}{f^2}$ und $\frac{1}{f}$ proportionale Terme mit unterschiedlichen Grenzfrequenzen. Für größere Frequenzdifferenzen konvergiert das Phasenrauschen gegen das *Broad-Band*-Phasenrauschen, welches durch die zeitlich unkorrelierte *Jitter*-Komponente verursacht wird. Im *Leeson*-Modell wird diese Komponente als thermisches Rauschen angenommen und durch den Term $\frac{2k_B T F}{P_{\text{Sig}}}$ beschrieben. Hierbei ist k_B die Boltzmann-Konstante, T die thermodynamische Temperatur, P_{Sig} die Signalleistung und F die empirisch bestimmte *Excess-Noise-Number*. Das *Broad-Band*-Phasenrauschen begrenzt zusätzlich zur Amplitudenquantisierung den möglichen Dynamikumfang des Messsystems, (englische Bezeichnung *Spurious-free Dynamic Range, SFDR*). In der Literatur werden viele verschiedene Modelle für die Phasenrauschleistungsdichte beschrieben. Allen gemeinsam ist, dass prinzipiell nur die Spektralleistungsdichte des Phasenrauschens, nie jedoch die tatsächliche Phasenabweichung zu einem Zeitpunkt modelliert wird. Dies ist auf Grund der statistischen Natur des Phasenrauschens als zeitvarianter, cyclostatistischer Prozess nicht möglich.

4.4.1 Grafische Darstellung der Auswirkungen von Phasenrauschen

Die Auswirkungen von Abtastphasenrauschen werden in diesem Kapitel anhand von Grafiken, welche auch als animierte Videos verfügbar sind [87], visualisiert. Abbildung 4.10 zeigt überdramatisiert die Auswirkungen von zeitlich korrelierten (zeitvarianten) Abtastzeitpunkt-Abweichungen. Oben in der Grafik ist die Zeitabweichung der Abtastzeitpunkte in Vielfachen der Zeit zwischen zwei Abtastzeitpunkten $\frac{1}{f_s} = \Delta t$ dargestellt. Der Mittelwert der Zeitabweichung ist Null. Zu Beginn der Simulation beträgt die Zeitabweichung $-1,25 \Delta t$ und steigt bis zum Zeitpunkt $t = 0,5 \text{ s}$ linear bis auf eine Abweichung von $1,25 \Delta t$, danach fällt die Zeitabweichung wieder linear ab. Wie auch zuvor in den Grafiken 4.5 bis 4.8, ist die Zeit innerhalb der Simulation regenbogenfarbig kodiert: **violett** zu Beginn, **grün** in der Mitte und **rot** am Ende.

Die mit \star markierten Werte wurden an den nicht-äquidistanten Zeitpunkten $t = t_n + \Delta t_n$ erfasst und werden auch mit diesen Zeitpunkten dargestellt. Deshalb liegen diese Werte immer auf dem **blau** gestrichelt dargestellten nominellen Sinusverlauf. Wird äquidistantes Abtasten angenommen, werden die Werte betrachtet, als ob sie an den nominellen Zeitpunkten t_n erfasst worden wären. Dadurch kommt es zu Amplitudenabweichungen; es kann jedoch prinzipiell nicht zu einer Zunahme der Amplitude durch das Abtastphasenrauschen kommen, da die tatsächliche Signalamplitude durch die zeitliche Verschiebung nicht geändert wird. Die Pfeile visualisieren zusätzlich die zeitliche Verschiebung vom tatsächlichen (\star) zum angenommenen Zeitpunkt (\bullet). Im unteren Teil der Abbildung 4.10 ist die zeitliche Überlagerung der vier Sinusperioden dieses Beispiels dargestellt. Da die Abtastfrequenz in der Simulation ($f_s = 40 \text{ Hz} = 10 \cdot f_{\text{Signal}}$) ein ganzzahliges Vielfaches der Signalfrequenz ist, befinden sich die äquidistant angenommenen Werte (Punkte) an denselben relativen Zeitpunkten innerhalb einer Periode.

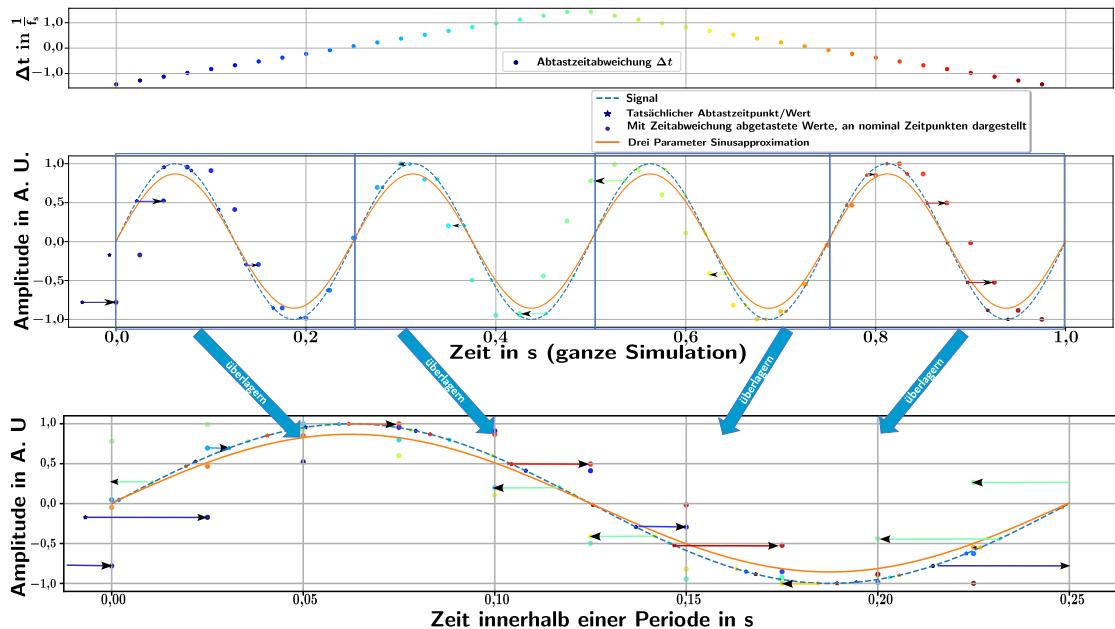


Abbildung 4.10: **Auswirkungen von Abtastphasenrauschen auf ein Sinussignal.** Oben: Zeitabweichungen der Abtastpositionen in Vielfachen der Abtastperioden. Farblich kodiert ist die Position der Werte innerhalb des 1 s langen Simulationsfensters. Mitte: Die Werte an ihren tatsächlichen Abtastzeitpunkten sind mit \star gekennzeichnet; diese liegen immer auf dem blau gestrichelt markierten nominalen Signal. Da äquidistantes Abtasten angenommen wird, werden diese Werte an den durch \bullet gekennzeichneten Zeitpunkten erwartet und auch so verwendet. Dabei zeigen die Pfeile die zeitliche Verschiebung eines Messpunktes vom tatsächlichen Zeitpunkt zum angenommenen Zeitpunkt. Unten: Überlagerung der vier Perioden auf eine relative Zeitbasis mit $\frac{1}{4}$ der Beobachtungszeit. Die Abtastzeitpunktverschiebungen führen zu einer deutlichen Reduzierung der von 3PSA oder FT bestimmten Amplitude (orange).

Simulationen der Auswirkung von Phasenrauschen bei verschiedenen Signalfrequenzen

Zur Untersuchung und Visualisierung der Auswirkungen von Abtastzeitabweichungen auf die Auswertung dynamischer Signale unter der Annahme, es sei äquidistant abgetastet worden, wurden Simulationen durchgeführt. Die in diesem Abschnitt gezeigten Grafiken sind Standbilder aus Videos, die die Simulationsergebnisse für alle Fourier-Frequenzen (ganze Periodenzahl im Simulationszeitraum \rightarrow kein Leck-Effekt) unterhalb der Shannon-Grenze der Simulation zeigen. Die Videos können entweder durch den QR-Code am Bild auf Zenodo [87] oder unter der URL <https://beseetek.github.io/jitterOnSineSignalPages/phdSeeger.html> aufgerufen werden.

Für alle Simulationen wurde eine Dauer von 1 s bei einer Abtastrate von 1000/s verwendet. Die Fourier-Frequenzen sind alle ganzen Zahlen bis 499 Hz. Die Amplitude des nominalen Signals beträgt 1, die Initialphase ist 0° . Alle Zeitabweichungen haben den Mittelwert 0. Für reale Messungen setzt dies die Kenntnis der mittleren Abtastrate im Beobachtungszeitraum voraus. Durch eine 4PSA kann die mittlere Abtastfrequenz aus Messdaten mit bekannter Anregungsfrequenz bestimmt werden. Bei statisch verteiltem Jitter wurden pro Frequenz 1000 Simulationen durchgeführt. Bei den Approximationsergebnissen (vorletzter Plot) werden die 1000 simulierten Amplituden (orange) und Phasen (violett) überlagert dargestellt. Bei der FFT werden Median und Perzentilgrenzen gezeigt.

Zeitlich unkorrelierter Jitter

Zum Einstieg in die Thematik soll die Auswirkung von normalverteiltem, zeitlich unkorreliertem Jitter mit einer Standardabweichung (RMS-Amplitude) von $0,2 \Delta t$ betrachtet werden. Die Abbildung 4.11 zeigt diese Simulation für eine Frequenz von 304 Hz. Die Überlagerung der Sinuskurven und der Abtastwerte entsprechend zu Abbildung 4.10 (unten) ist als erstes gezeigt. Da die Werte zu verschobenen Zeitpunkten erfasst werden, können keine Amplituden größer als 1 vorkommen, da das nominale Signal nie eine größere Amplitude hat. Da die Zeitabweichungen, wie in der zweiten Grafikzeile erkennbar, zeitlich vollkommen unkorreliert sind, scheinen die Messwerte in einem in X-Richtung (zeitlich) normalverteilten Band um den nominalen Verlauf zu liegen.

Die dritte Spalte zeigt die Residuen zwischen Nominalwert (blau) bzw. der approximierten Funktion (orange) und den simulierten Werten. Es ist keine Systematik in den Residuen erkennbar.

Die vorletzte Spalte zeigt ähnlich eines *Bode*-Diagramms die von der 3PSA bestimmte Amplitude (orange) und Phase (violett) für die simulierten Fourier-Frequenzen. Da die Fourier-Frequenzen betrachtet werden, sind 3PSA und FT für diese Frequenzen identisch [45]. Die von 3PSA oder FT bestimmte Amplitude nimmt mit steigender Frequenz kontinuierlich ab. Die Streuung der Amplitudenwerte nimmt dabei kontinuierlich zu. Der Verlauf der Amplitudenabnahme folgt für normalverteilten Jitter einer *Gauß*-Funktion. Mittelwertsfreier Jitter hat vernachlässigbaren Einfluss auf die Bestimmung der Phase.

In der vorletzten Grafik sind zusätzlich noch die RMS-Amplituden bzw. die Standardabweichungen der Signale in Blau gezeigt. Diese liegt konstant bei 1. Denn nach dem Satz von Plancherel

$$\int_{\mathbb{R}} |f(t)|^2 dt = \int_{\mathbb{R}} |f(\omega)|^2 d\omega \quad (4.19)$$

gilt, dass die RMS-Amplitude eines Signals dem Integral des Leistungsdichtspektrums PSD entspricht. Durch das Abtasten mit Jitter ist die RMS-Amplitude unverändert geblieben, die Amplitude der Simulationsfrequenz in der FT bzw. 3PSA hat aber drastisch abgenommen. Da Gleichung 4.19 unverändert gilt, muss die verbleibende Signalenergie auf andere spektrale Komponenten verteilt worden sein [93, 45]!

Dieser Effekt im letzten Teil (PSD des Signals unter Annahme von äquidistantem Abtasten) der Simulationsgrafik 4.11 klar erkennbar. Die verbleibende Signalenergie wird im Mittel gleichmäßig über alle Frequenzen verteilt. Die Y-Achse dieser Grafik zeigt den Logarithmus der Leistungsdichte bezogen auf die Simulationsamplitude (daher auch die Einheit dBc für *Decibel Carrier*). Die Streuung der Leistungsdichten ist dabei erheblich, wie an den 30 dBc Abstand zwischen den Perzentilgrenzen erkennbar ist. Die hier nicht gezeigte Phase der Spektren ist abgesehen von der Phase der Signalfrequenz vollkommen zufällig.

Für zeitlich unkorrelierten Jitter kann das Verhältnis $\frac{\hat{A}_{\text{Fit}}}{A_{\text{nominell}}}$ zwischen der approximierten Amplitude \hat{A}_{Fit} und der nominellen Signalamplitude A_{nominell} direkt aus der Fourier-Transformation der zeitlichen Wahrscheinlichkeitsdichtefunktion (PDF) $\Pr_{\text{Jitter}}(t)$ des Jitters bestimmt werden [1, 93].

$$\frac{\hat{A}_{\text{Fit}}(f)}{A_{\text{nominell}}(f)} = \mathcal{F}(\Pr_{\text{Jitter}}(t)) \quad (4.20)$$

Dies kann mit dem Faltungstheorem begründet werden. Das Abtasten mit einem zeitlich nicht korrelierten Abtast-*Jitter* kann auch wie eine Faltung der idealen *Delta-Dirac*-Abtastpulse mit der PDF des *Jitters* verstanden werden. Da eine Faltung im Zeitbereich einer Multiplikation im Frequenzbereich entspricht, wird die Amplitudenänderung durch die **Fourier-Transformation** der *Jitter*-PDF beschrieben.

Zeitlich korrelierte Abtastzeitabweichungen

Die Messungen der Abtastzeitpunkte (siehe z. B. Abb. 4.13 bis 4.14b) von realen Sensoren zeigen, wie auch in der Literatur [38] beschrieben, dass die Annahme zeitlich unkorreliertem *Jitter* nicht der Realität entspricht. Abbildung 4.12 zeigt beispielhaft die Auswirkungen von einer Abtastzeitpunktsdrift, die durch eine quadratische Funktion beschrieben wird. Die quadratische Funktion wurde als Näherung für den Verlauf der Sensoren *MPU-9250* und *LSM6DSRX* gewählt; online [87] sind auch Videos für andere Abweichungsverläufe (Cosinus, Gauß-Funktion, Sägezahn oder Dreieck) verfügbar. In der oberen Darstellung der simulierten Werte sind die Amplitudenabweichungen durch die korrelierte Abtastzeitabweichung erkennbar². Die einzelnen Signalperioden werden durch die kumulierte Zeitabweichung phasenverschoben. Die Perioden, die am Anfang und Ende (blau und rot) der Simulation liegen, werden aufgrund der negativen Abtastzeitabweichung zu positiven Phasen verschoben; die in der Mitte der Simulation liegenden Perioden (grün) werden entsprechend zu negativen Phasen verschoben.

Je größer die kumulierte Zeitabweichung eines Abtastwertes ist, desto größer ist auch die Einhüllende der Residuen. Da die Phase der Approximation mit der Signalphase übereinstimmt, gleichen sich die Verläufe der Residuen von Approximation oder Nominalwerten zu den simulierten Werten. Aufgrund des Amplitudenfehlers bei der Approximation sind die Residuen der Approximation jedoch insgesamt etwas größer.

Auch die kumulierte, quadratische Abtastzeitabweichung für höhere Frequenzen führt zu einer signifikanten Abnahme der approximierten Amplitude auf nur noch 60 % des Nominalwerts bei 490 Hz Signalfrequenz. Die **RMS**-Amplitude des Signals bleibt, wie auch beim zeitlich unkorrelierten *Jitter*, für alle Frequenzen bei 1. Die Signal-Energie bleibt also erhalten.

In der Darstellung (Abb. 4.12 unterster Plot) der aus den Daten mittels **FT** berechneten **PSD** ist die in Abbildung 4.9 gezeigte Verbreiterung des Signals durch Faltung mit dem Abtastphasenrauschen klar erkennbar. Im Video ist zu erkennen, dass die Form der Verbreiterung, jedoch nicht die Breite der Verbreiterung durch das Phasenrauschen, frequenzunabhängig ist. Die Breite der Verbreiterung des Signals durch das Phasenrauschen des Abtasttaktes ist linear von der Frequenz abhängig. Wie in Abbildung 4.2 und in Gleichung 4.3 zu erkennen, hängt die Bandbreite der **3PSA**, oder analog dazu die Bandbreite eines **FFT**-Bins, von der Beobachtungsfensterlänge ab. Durch die mit zunehmender Signalfrequenz ebenfalls zunehmende spektrale Verbreiterung durch das Abtastphasenrauschen wird ein immer kleiner werdender Teil der Signalenergie von der Bandbreite der **3PSA** oder **FFT** erfasst, und die approximierte Amplitude sinkt folglich.

²Dieser Effekt ist im Video deutlicher sichtbar.

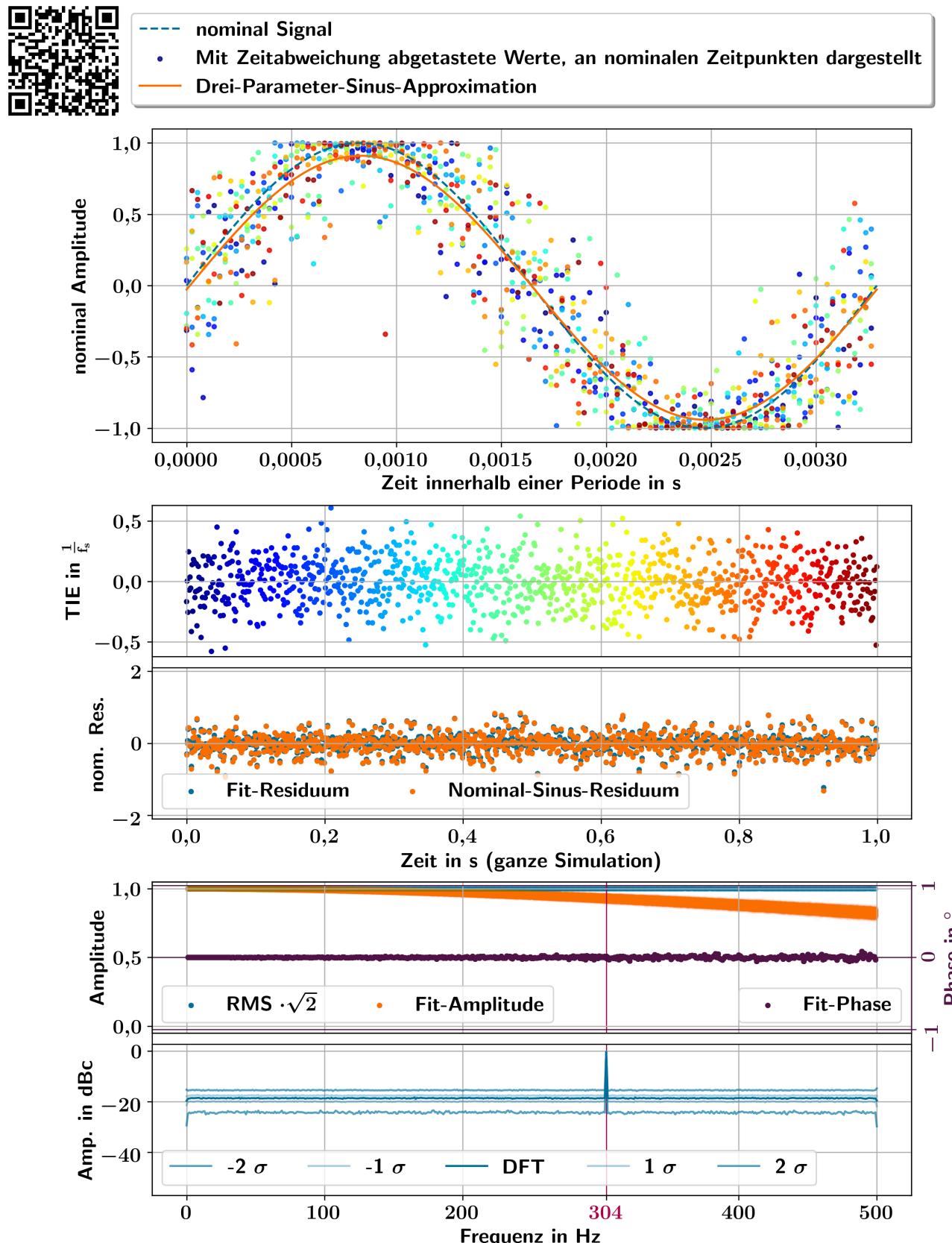


Abbildung 4.11: **Simulation der Auswirkung von normalverteiltem Abtast-Jitter mit $\sigma = 0,2 \Delta t$ auf die 3PSA bzw. FT.** Diese Grafik ist ein Standbild aus dem Video, welches auch durch den QR-Code im Bild aufgerufen werden kann. Die Video-Dateien sind zusätzlich auf Zenodo unter DOI:10.5281/zenodo.10111677 [87] abrufbar.

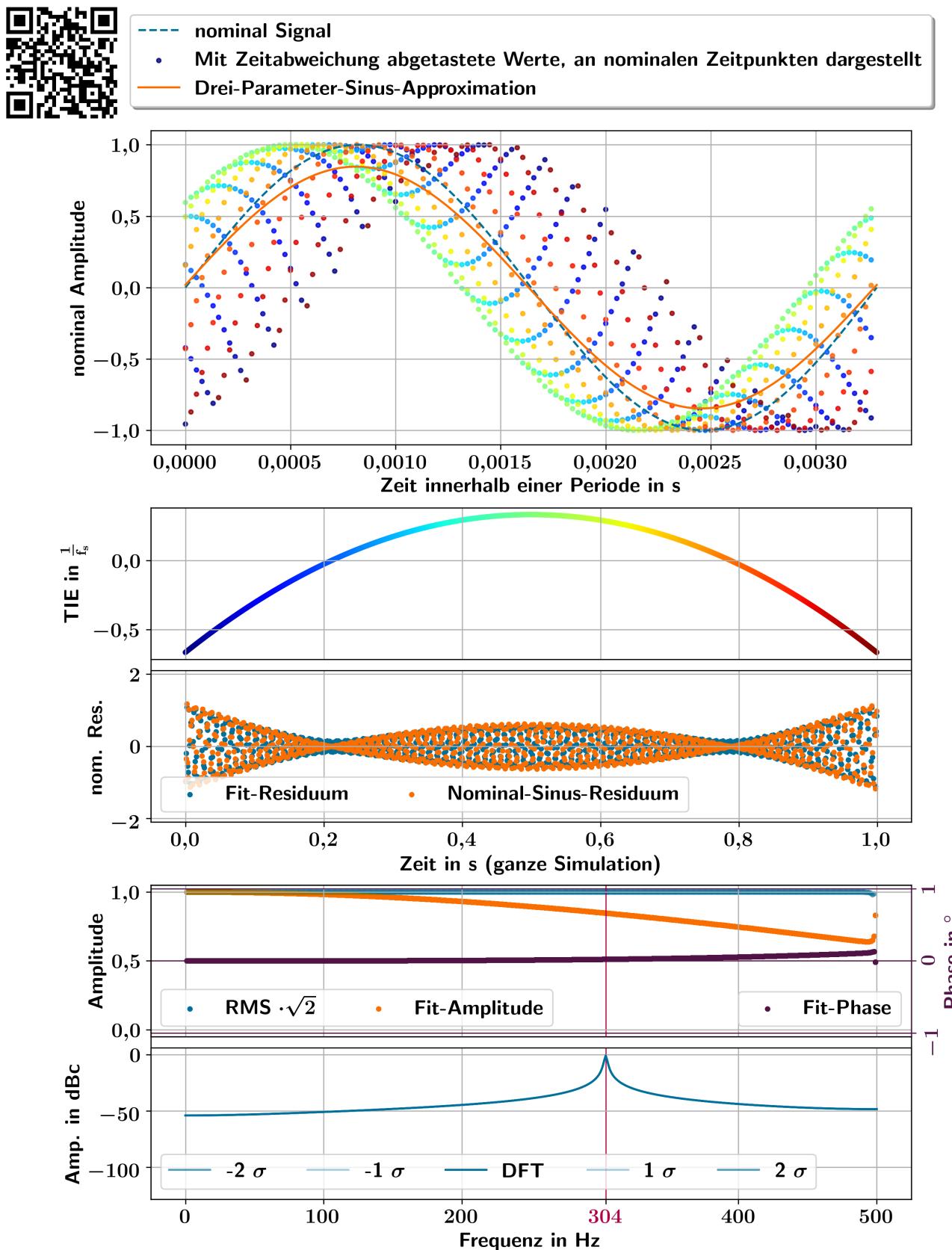


Abbildung 4.12: **Simulation der Auswirkung von quadratischer Abtastzeitpunktsabweichung mit Amplitude $1\Delta t$ auf die 3PSA bzw. FT.** Diese Grafik ist ein Standbild aus dem Video, welches auch über den QR-Code im Bild aufgerufen werden kann. Die Video-Dateien sind zusätzlich auf Zeno-
do unter DOI:10.5281/zenodo.10111677 [87] abrufbar.

4.5 Bestimmung des Phasenrauschens aus Abtastzeitdifferenzen

Wie an den vorangegangenen Beispielen gezeigt, ist das Phasenrauschen $\mathcal{L}(f)$ des Abtasttaktes eine entscheidende Einflussgröße für das Auswerten von dynamischen Signalen im Frequenzraum. Die in der IEEE 1057 2017[45] vorgeschlagenen Messmethoden unter der Verwendung von Spektrumanalysatoren für das Oszillator-Phasenrauschen können aufgrund der niedrigen Abtastfrequenz der Sensoren nicht sinnvoll angewendet werden. Daher wird in dieser Arbeit das Abtastphasenrauschen aus den Absolutzeitstempeln der Abtastzeitpunkte berechnet. Zur Bestimmung der Abtastzeitpunktsabweichungen der Sensoren wird eine an die Validierung der GNSS-Zeitbasis (Abschnitt 3.3.1) angelehnte Methodik verwendet. Mit der DAU werden die $n + 1$ Zeitstempel der DRIs der Sensoren für eine bestimmte Zeitspanne $\Delta t_{\text{FreqMeas}} = t(n) - t(0)$ erfasst. Der Mittelwert der Abtastfrequenz \bar{f}_s in der Zeitspanne $\Delta t_{\text{FreqMeas}}$ ist dann

$$\bar{f}_s = \frac{n}{t(n) - t(0)} = \frac{n}{\Delta t_{\text{FreqMeas}}}. \quad (4.21)$$

Mit der so bestimmten Abtastfrequenz können nun die erwarteten, ideal äquidistanten Zeitstempel (Ideal Time Course, ITC) berechnet werden. Analog zu Gleichung 3.6 gilt für die kumulierte Zeitabweichung, den Time Interval Error (TIE) [76], folgender Ausdruck.

$$\text{TIE}(n) = \Delta t(n) = t(n) - t(0) - t_{\text{ITC}}(n) = t(n) - t(0) - n \cdot \frac{1}{\bar{f}_s} \quad (4.22)$$

Abbildung 4.13 zeigt diese Zeitabweichung für verschiedene Sensoren sowohl in Millisekunden als auch in Vielfachen der Zeit zwischen zwei Abtastpunkten.

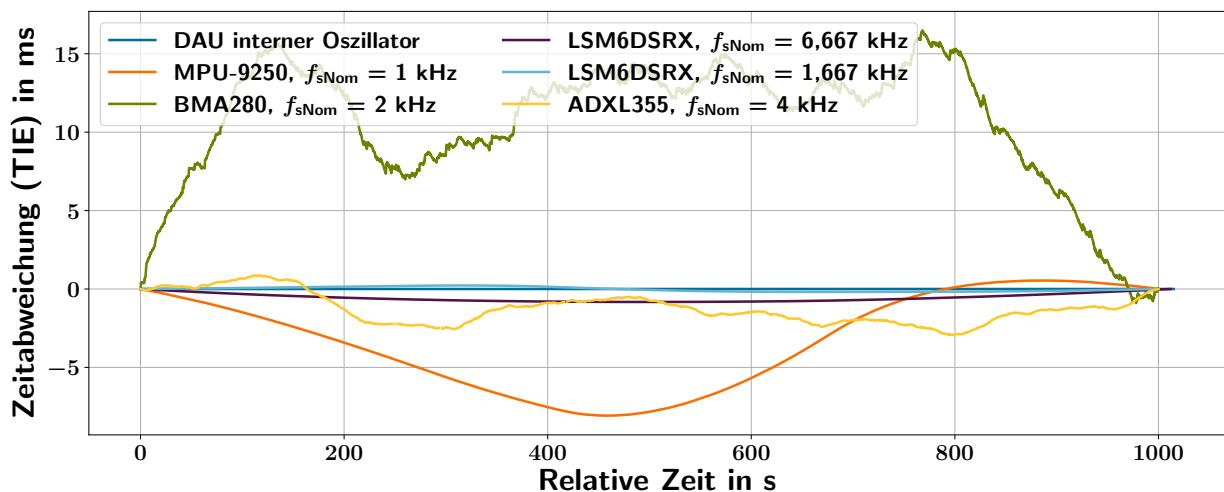


Abbildung 4.13: **Zeitabweichung der Abtastzeitpunkte verschiedener Sensoren vom äquidistanten Verlauf über 1000 s.** Für die Berechnung der ideal äquidistanten Zeitpunkte wurde die aus tatsächlichen Zeitpunkten bestimmte mittlere Abtastfrequenz verwendet, um lineare Zeitabweichungen zu kompensieren.

Über die relativ lange Beobachtungszeit von 1000 s zeigen sich bei einigen Sensoren, trotz korrekt bestimmter Abtastfrequenz f_s , erhebliche Zeitabweichungen von vielen Abtastpunkten. Die Zeitabweichungen sind zeitlich stark korreliert. Die Sensoren *BMA280* und *ADXL355* zeigen einen an *Random-Walk* erinnernden Verlauf. Der *MPU-9250* zeigt einen wesentlich glatteren Verlauf, der möglicherweise durch eine Temperaturkompensation in einer **PLL**-Regelung verursacht wird. Die Sensoren *MPU-9250* und *LSM6DSRX* sind **IMUs** und enthalten somit auch Gyroskope. **MEMS**-Gyroskope basieren oft auf vibrierenden Strukturen, in denen die Corioliskraft zu einer Energieübertragung zwischen Schwingungsmoden bzw. Richtungen führt. Die daraus resultierende Änderung der Phasenlage der Schwingung des Gyroskops wird anschließend ausgewertet [108, 37]. Aufgrund ihres Arbeitsprinzips benötigen/erzeugen diese Gyroskope eine sehr stabile Anregungsfrequenz, die mittels einer **PLL** zur Erzeugung des Abtasttaktes verwendet werden kann, was bei diesen Sensoren der Fall ist. Daher zeigen sich bei kurzen Beobachtungszeiten nur geringfügige Abtastzeitabweichungen (siehe Abbildung 4.14). Weder die Zeitabweichungen des *BMA280* noch die der anderen Sensoren sind im kurzen Beobachtungszeitraum (siehe Abb. 4.14a) unkorreliert zufällig. Die **IMUs** zeigen für kürzere Beobachtungszeiten (siehe Abb. 4.14b) einen relativ konstanten Verlauf der Zeitabweichungen, der einer quadratischen Funktion ähnelt. Dieser glatte Verlauf wird durch die geringe Regelbandbreite der **PLLs** in den **IMUs** verursacht, da die Zeitkonstanten dieser Regler groß gegen die 10 s Beobachtungszeit sind. Die Korrektur der mittleren Abtastrate führt dazu, dass alle **TIE**-Plots immer mit $\Delta t = 0$ starten und enden. Die im Sekudentakt auftretenden Zeitsprünge mit 10 ns oder 20 ns Amplitude stammen vermutlich aus der **GNSS-PPS**-Synchronisation in der **DAU** und nicht aus den Sensoren selbst.

4.5.1 Berechnung der Phasenrauschleistungsdichte aus den Abtastzeitpunkten

In den vorangegangenen Abschnitten, und insbesondere in Abbildung 4.12, ist die Bedeutung des Phasenrauschen als wichtiger Einfluss auf dynamische Messungen direkt erkennbar.

Daher finden sich auch in der Literatur aus der Hochfrequenztechnik viele Veröffentlichungen zu diesem Themenkomplex und das Abtastphasenrauschen von **ADC**-Systemen wird generell als wichtiger Einflussfaktor behandelt und dabei auch charakterisiert und quantifiziert [1, 22, 54, 38, 25, 90]. Die in der Literatur beschriebenen Messverfahren zur Bestimmung des Phasenrauschen lassen sich wegen der sehr niedrigen Taktfrequenzen nicht direkt auf die Abtasttakte (bzw. die **DRI**-Zeitpunkte) der **MEMS**-Sensoren anwenden. Typischerweise wird die Phasenrauschleistungsdichte $\mathcal{L}(f)$ eines Oszillators direkt mit einem Spektrumanalysator bestimmt.

Der Spektrumanalysator erfasst direkt die Leistung innerhalb eines Frequenzbandes, welches durch ein analoges Filter selektiert wird. Dabei bestimmt die **Resolution Bandwidth (RBW)** des Filters durch seine Einschwingzeit die Scanrate. Bei Signalfrequenzen von 1 kHz ist zur sinnvollen Untersuchung des Phasenrauschen mindestens eine Auflösung im mHz-Bereich nötig. Durch die Einschwingzeit von 1000 s pro Frequenzpunkt wären diese Messungen, selbst wenn sie technisch möglich wären, nicht praktikabel durchführbar. Bei kommerziellen Spektrumanalysatoren ist die untere Grenze der Signalfrequenz in der Regel 10 kHz und die kleinste **RBW** im Bereich von 100 mHz. Zur Untersuchung von tieffrequenten Signalen können **FFT**-Analyzer verwendet werden. Diese Geräte digitalisieren das Eingangssignal mit einem **ADC** mit hoher Auflösung und stabiler Zeitbasis. Die **FFT** der **ADC**-Signale liefert dann direkt das Spektrum bis zur halben Abtastfrequenz mit einer

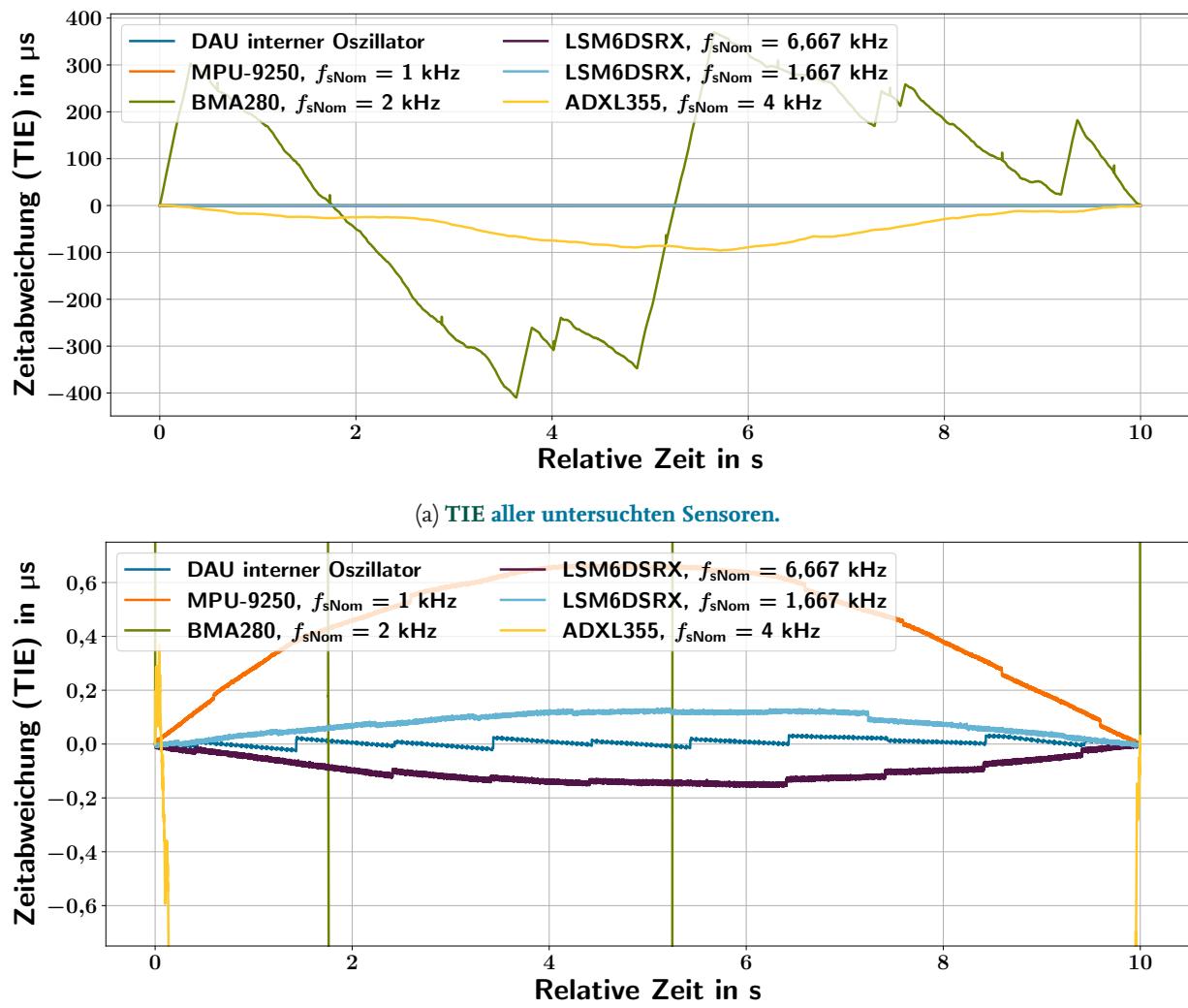


Abbildung 4.14: **Zeitabweichungen der Abtastzeitpunkte (TIE) verschiedener Sensoren vom äquidistanten Verlauf über 10 s.** (a) Alle Sensoren außer BMA280 und ADXL355 weisen eine gute Kurzzeitstabilität auf. Der Sensor BMA280 zeigt Random-Walk-artige Änderungen der Abtastrate. (b) Für den hier gezeigten Zeitraum von 10 s kann für die Sensoren MPU-9250 und LSM6DSRX aufgrund der geringen Abweichungen ein äquidistantes Abtasten angenommen werden. Die Zeitsprünge um Vielfache von 10 ns im Sekudentakt in den Vergrößerungen werden durch die simple DAU-GNSS-Synchronisierung und nicht durch die Sensoren selbst verursacht.

durch die Messdauer definierten Auflösung. Eine Messung des Abtastphasenrauschens der MEMS-Sensoren ist mit diesen Geräten also prinzipiell möglich. Wird die vertikale Auflösung auf 1 Bit reduziert, wird auch das Amplitudenrauschen durch Quantisierung unterdrückt. Da die DAU die Absolutzeitstempel der DRI-Flanken liefert, ist eine Bestimmung des Phasenrauschens aus diesen Zeitpunkten sinnvoll. Um dies zu ermöglichen, wird ausgenutzt, dass die mathematische Faltung ein kommutativer Prozess ist. Das Abtasten eines durch Phasenrauschen verbreiterten Signals mit einem perfekten ADC und einer perfekten Zeitbasis ergibt dasselbe Signal, wie beim Abtasten eines perfekten monofrequenten Signals an den Nulldurchgängen des durch Phasenrauschen beeinträchtigten Signals.

Diese Kommutativität macht sich das in dieser Arbeit verwendete Verfahren zu Nutze. Durch die im vorherigen Abschnitt beschriebene Methodik sind die tatsächlichen Abtastzeitpunkte bzw. die Abweichung vom erwarteten Zeitpunkt (TIE bzw. $\Delta t(n)$) sowie die mittlere Abtastrate \bar{f}_s bekannt. Daher ist es direkt möglich, ein synthetisches Signal $S(t_n)$ zu erzeugen, das dem mit der mittleren Abtastfrequenz \bar{f}_s auf $f = 0$ herunter gemischten, äquidistant abgetasteten Oszillatorsignal entspricht.

$$S(n) = \cos(\Delta t(n) \cdot 2\pi\bar{f}_s) + i \sin(\Delta t(n) \cdot 2\pi\bar{f}_s) = e^{i(\Delta t(n) \cdot 2\pi\bar{f}_s)} \quad (4.23)$$

Ohne Abtastphasenrauschen ist $\Delta t(n) = 0$ und es wird nur $1 + i0$ abgetastet; dies entspricht im Frequenzraum einem Dirac-Plus an der Stelle $f = 0$. Da das Signal aber an der mittleren Abtastfrequenz \bar{f}_s herunter gemischt wurde, entspricht dieses Spektrum dem eigentlichen Oszillatorsignal. Für reale Messungen liefert die **FT** dieses Signals das Phasenrauschspektrum. Die Phasenrauschleistungsdichte ist der dekadische Logarithmus des Verhältnisses der Signalleistung zur Leistung in den anderen Frequenzbändern, angegeben in dB. Der Bezug auf die Trägerleistung wird durch den Index c (engl. *carrier signal*) kenntlich gemacht. Da der simulierte Träger eine Amplitude von 1 hat ist eine Normierung nicht nötig. Folgende Gleichung liefert daher direkt die Phasenrauschleistungsdichte aus den Abtastzeitdifferenzen bei bekannter mittlerer Signalfrequenz.

$$\mathcal{L}(f) = 10 \cdot \log_{10} \left[\frac{2}{n \cdot \bar{f}_s} |\mathcal{F}\{S(n)\}|^2 \right] = 10 \cdot \log_{10} \left[\frac{2}{n \cdot \bar{f}_s} \left| \mathcal{F}\{e^{i(\Delta t(n) \cdot 2\pi\bar{f}_s)}\} \right|^2 \right] \quad (4.24)$$

Die Signalverbeiterung durch Phasenrauschen ist linear frequenzabhängig (siehe auch Video von Abb. 4.12). Um die Qualität verschiedener Oszillatoren vergleichen zu können, werden in dieser Arbeit die Phasenrauschleistungsdichtespektren gegen den normierten Frequenzabstand zur Oszillatorkreuzfrequenz aufgetragen. Die Abbildungen 4.15 zeigen die durch *Bartletts Method* [8] gemittelten Phasenrauschleistungsdichten aus Spektren von *Slices* mit jeweils $2^{21} = 2097152$ Abtastpunkten. Die Teilabbildung 4.15b zeigt eine Vergrößerung um kleine Frequenzabstände. Für eine glattere Darstellung wurde das gemittelte Phasenrauschen mit einem Gauß-Tiefpass mit der Breite $\sigma = 1$ FFT-Bin geglättet.

Die **DAU**-Zeitbasis zeigt durch die sekündliche **GNSS**-Synchronisierung, auf lange Sicht, eine hinreichend gute Stabilität. Der einfache Synchronisierungsalgorithmus verursacht die im Abstand von $0,001 \frac{\Delta f}{f_s}$ auftretenden Peaks in der Phasenrauschleistungsdichte; die Testsignalfrequenz für die **DAU** war in diesem Aufbau 1 kHz. Die sich neben den Peaks bei $n \cdot 0.001 \frac{\Delta f}{f_s}$ zeigenden symmetrischen Peaks stammen vermutlich aus der sägezahnförmigen Frequenzdrift der **PLL-PLL**-Kombination der **DAU** (vgl. 3.13a). In allen anderen Phasenrauschspektren findet sich auch immer das Phasenrauschen der **DAU**-Zeitstemplung, dies ist aber für die hier untersuchten Sensoren unkritisch. Der Beschleunigungssensor **BMA280** zeigt aufgrund der *Random-Walk*-artigen starken Abtastzeitabweichungen des integrierten RC-Oszillators starkes Phasenrauschen mit einer *Broad-Band*-Phasenrauschleistungsdichte von $-70 \frac{\text{dBc}}{\text{Hz}}$.

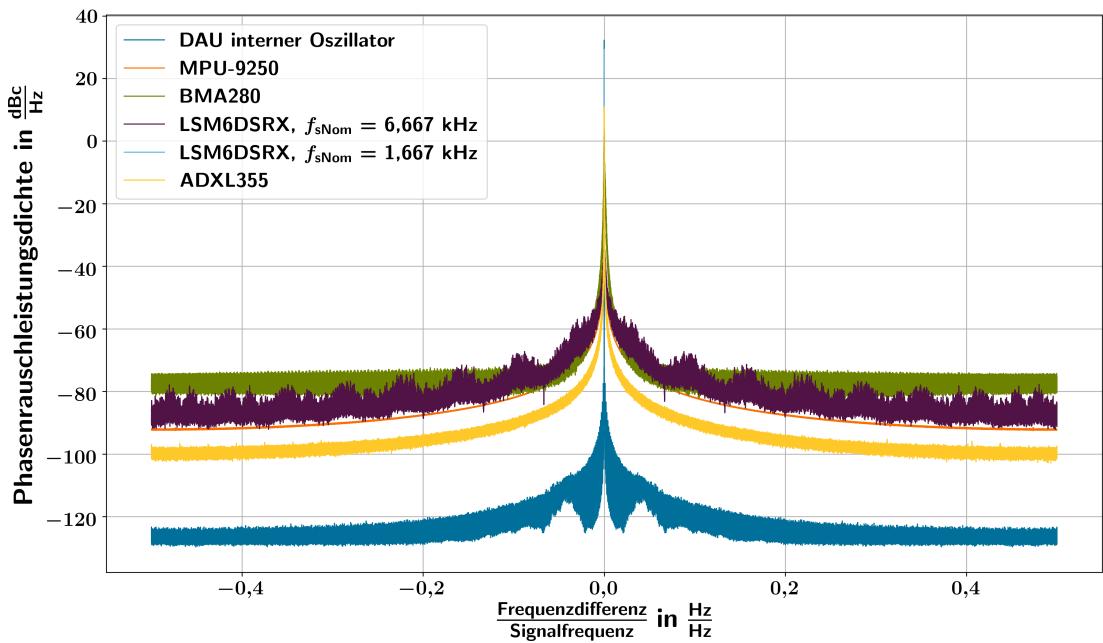
Die **IMUs** **MPU-9250** und **LSM6DSRX** ($f_{s\text{Nom}} = 1,667\text{kHz}$) zeigen den typischen Verlauf des Phasenrauschen für Oszillatoren mit **PLL**-Regelung. Bis zur Regelbandbreite ist die *Narrow-Band*-Phasenrauschleistungsdichte etwa $20 \frac{\text{dBc}}{\text{Hz}}$, für größere Frequenzabstände konvergiert die Rauschleistungsdichte kontinuierlich in einem glatten Verlauf gegen $-90 \frac{\text{dBc}}{\text{Hz}}$. Für die **IMU** **LSM6DSRX** mit reduzierter Datenrate von $f_{s\text{Nom}} = 1,667\text{kHz}$ reicht die Beobachtungsdauer von 1272 s nicht

aus, um das *Narrow-Band*-Phasenrauschen aufzulösen. Für höhere Frequenzabstände konvergiert die Rauschleistungsdichte gegen $-100 \frac{\text{dBc}}{\text{Hz}}$. Wird bei der **IMU LSM6DSRX** die maximale Datenrate von $f_{\text{sNom}} = 6,667 \text{kHz}$ ausgewählt, ändert sich der Verlauf des Phasenrauschens und die Glättung durch die **PLL** ist nicht mehr erkennbar. Die Rauschleistungsdichte steigt um etwa 10 dB an, was auf die Deaktivierung der **PLL** hindeutet.

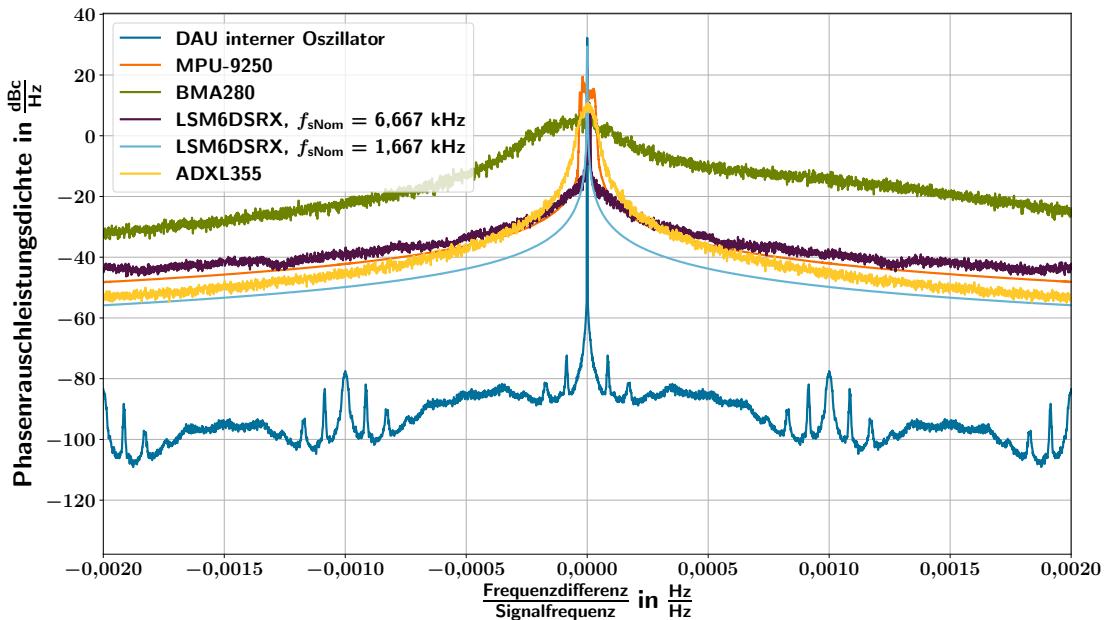
Die Auswirkungen des Phasenrauschens auf ein monofrequentes Signal mit der Frequenz 80 Hz visualisiert Abbildung 4.16. Im Unterschied zur auf die Abtastfrequenz normierten Darstellung des Frequenzabstands in Abb. 4.15 sind in Abb. 4.16 die Rauschleistungsdichten im Bereich $-1,5 \text{Hz}$ bis $1,5 \text{Hz}$ für absolute Frequenzabweichungen gezeigt. Die Abbildungen 4.19a und 4.19b zeigen die Amplitudenspektren reeller Kalibermessungen mit und ohne kompensiertem Abtastphasenrauschen.

Die nach Gleichung 4.3 berechneten spektralen Empfindlichkeiten für die Beobachtungsfensterlängen 1 s, 10 s und 30 s sind gestrichelt dargestellt.

Die durch eine Sinus-Approximation bestimmte Amplitude ist das unendliche Integral des Produktes aus spektraler Empfindlichkeit (*Window Response Function*) und durch Phasenrauschen verbreitertem Signal. Für Beobachtungsfensterlänge von 1 s wird für alle Sensoren nahezu die komplette Signalenergie von einer **3PSA** oder **FT** erfasst. Bei einer Beobachtungsfensterlänge von 10 s ist die 3-dB-Bandbreite nur noch $\frac{0,443}{10\text{s}} = 0,0443 \text{Hz}$, das *Narrow-Band*-Phasenrauschen des **BMA280** hat aber über einen wesentlich größeren Bereich noch signifikante Signalenergie. Daher ist bei einer Beobachtungsfensterlänge von 10 s oder mehr und einer Signalfrequenz von 80 Hz bei diesem Sensor selbst mit einer korrekt bestimmten mittleren Abtastfrequenz und unter der Annahme von äquidistantem Abtasten mit einer deutlichen Amplitudenabweichung zu rechnen. Bei einer Beobachtungsfensterlänge von 30 s ist auch bei der **IMU MPU-9250** mit einem kleinen Amplitudenfehler bei Messungen ohne Absolutzeitstemplung zu rechnen, da die Bandbreite der durch das *Narrow-Band*-Phasenrauschen verursachten Signalverbreiterung in etwa in der Bandbreite der Sinus-Approximation liegt. Keines der gezeigten Signale konvergiert innerhalb der Approximationsbandbreite der Beobachtungsfensterlängen 10 s und 30 s gegen die *Broad-Band*-Phasenrauschleistung. Daher ist bei diesen Messdauern für alle Sensoren mit Störungen durch Abtastphasenrauschen zu rechnen.



(a) Phasenrauschleistungsdichtespektrum.



(b) Ausschnitt des Phasenrauschleistungsdichtespektrums (a) für kleine Frequenzabstände.

Abbildung 4.15: **Phasenrauschleistungsdichtespektrum verschiedener Sensoren und der DAU.** Für *Slices* mit jeweils $n = 2^{21}$ Abtastwerten wurde die mittlere Abtastfrequenz \bar{f}_s bestimmt und nach Gleichung 4.24 die PSD des Phasenrauschen berechnet. Die gezeigte Leistungsdichte ist der Mittelwert der einzelnen *Slices*. Die Spektren wurden mit einem Gauß-Tiefpass mit $\sigma = 1$ FFT-Bin geglättet. Zur besseren Vergleichbarkeit ist der Frequenzabstand auf die Abtastfrequenz der Sensoren normiert. In der Gesamtansicht (a) ist die Konvergenz gegen das *Broad-Band*-Phasenrauschen für alle Sensoren klar erkennbar. Das Spektrum des **LSM6DSRX** mit **PLL** wird durch das Spektrum desselben Sensors ohne aktive **PLL** verdeckt. In der Vergrößerung (b) sind die *Narrow-Band*-Phasenrauschleistungskomponenten der Sensoren erkennbar. Die Regelbandbreite der **PLL** des **MPU-9250** führt zu einem Plateau der Phasenrauschleistungsdichte. Die bei der **DAU** auftretenden Spitzen bei Vielfachen von $0,001 \frac{\Delta f}{f_s}$ stammen aus der sekündlichen **PPS**-Synchronisation der **DAU**-Zeitbasis.

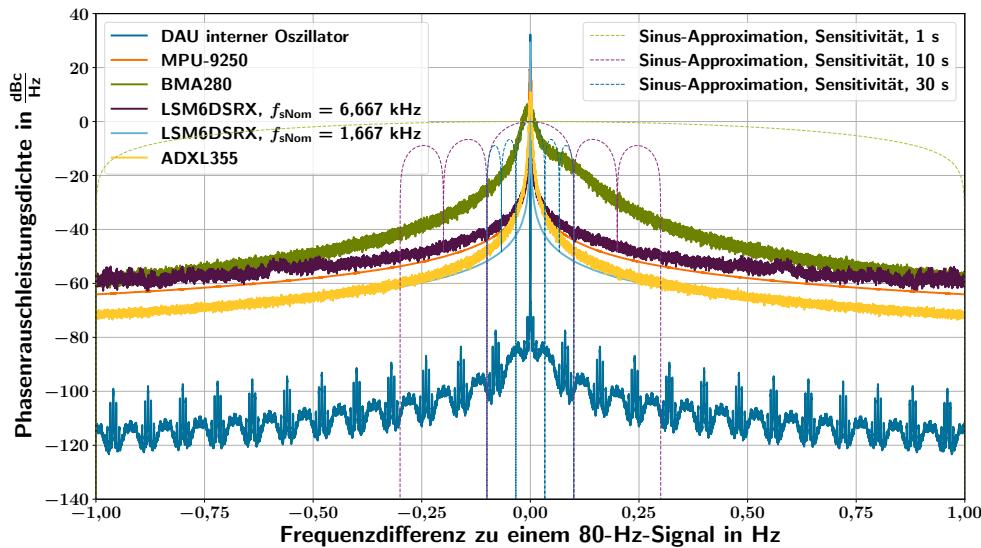


Abbildung 4.16: **Auswirkungen des Abtastphasenrauschens verschiedener Sensoren auf ein 80 Hz-Signal für verschiedene Beobachtungsfensterlängen.** Dargestellt ist die PSD eines simulierten idealen 80 Hz-Signals mit der Verbreiterung durch Phasenrauschen der Sensoren. Die Sinc-förmigen Sensitivitäten von FT oder 3PSA mit 1, 10 und 30 Sekunden Beobachtungsfensterlänge sind gestrichelt dargestellt. Die approximierte Amplitude ist das indefinite Integral des Produkts aus Sensitivität und dem durch Phasenrauschen verbreiterten Signal. Die Abb. 10.3 zeigt diese Grafik mit linearer Y-Achse.

4.5.2 Monte-Carlo-Simulation der Amplitudenabweichungen durch Phasenrauschen

Um die Auswirkungen des Abtastphasenrauschens auf Messungen mit korrekt bestimmter mittlerer Abtastfrequenz \bar{f}_s und als äquidistant angenommenem Abtasten zu untersuchen, wurden entsprechende Monte-Carlo-Simulationen durchgeführt. Dabei wurde für die einzelnen Simulations schritte wie folgt vorgegangen:

- Zeitreihe mit 10 s (oder 100 s) Länge aus den aufgezeichneten Daten auswählen.
- Mittlere Abtastfrequenz \bar{f}_s für die Zeitreihe berechnen.
- Sinussignal Y_{Simu} mit Amplitude $A_{\text{nom}} = 1$ und Phase $\varphi_{\text{nom}} = 0$ an den realen Zeitpunkten berechnen.
- Ideal äquidistante Abtastzeitpunkte t_{ITC} aus der mittleren Abtastfrequenz \bar{f}_s berechnen.
- 3PSA für Y_{Simu} und t_{ITC} durchführen.

Diese Simulation wurde für jede Frequenz 1000 mal durchgeführt. Durch die starke zeitliche Korrelation der Zeitabweichungen sind 1000 Simulationen pro Punkt ausreichend, obwohl zum Teil mehr als 1000 Zeitpunkte pro Durchgang simuliert werden. Abbildung 4.17 zeigt die Ergebnisse der Monte-Carlo-Simulation für eine Beobachtungsdauer von 10 s. Für den Beschleunigungssensor **BMA280** kommt es, wie schon bei der Betrachtung des Phasenrauschens und den Sinc-förmigen

Sensitivitäten der **FT** bzw. **3PSA** in Abbildung 4.17a erwartet, zu einer deutlichen Amplitudenabnahme für höhere Signalfrequenzen, z. B. 5 % Abweichung bei 200 Hz und bis zu 30 % Abweichung bei 500 Hz. Die Streuung der simulierten Amplituden ist nicht gleich-verteilt, sondern deutlich breiter bei niedrigeren Amplituden, da eine Amplitudenzunahme durch Abtastphasenrauschen de facto nicht möglich ist. Das Konfidenzintervall von 5 % bis 95 % erstreckt sich für den Sensor **BMA280** bei 10 s Beobachtungsfensterlänge und einer Frequenz von 990 Hz³ von 0,4 bis 0,98. Für den Sensor **BMA280** kann auch für Zeitreihen von 10 s Beobachtungsfensterlänge kein äquidistantes Abtasten angenommen werden. Für die **IMUs** **MPU-9250** und **LSM6DSRX**, die auch einen internen Oszillatator mit **PLL** besitzen, kann diese Annahme bedenkenlos getroffen werden, denn die relative Amplitudenabnahme durch das Phasenrauschen ist für 990 Hz Signalfrequenz für beide Sensoren kleiner als 10^{-5} (siehe Abb. 4.17b).

Bei der Sensorkalibrierung werden die Zeitstempel der **DAU** verwendet. Das Phasenrauschen der **DAU** führt dadurch direkt zu Amplitudenabweichungen bei der **3PSA**. Für die in dieser Arbeit verwendete Beobachtungsfensterlänge von etwa 10 s ist dadurch ein vernachlässigbarer Amplitudenfehler (5 %-95 % Konfidenzbreite) von maximal $1 \cdot 10^{-7}$ bei 990 Hz zu erwarten (siehe Abbildung 4.17c).

Wegen der **GNSS**-Disziplinierung der **DAU**-Zeitbasis führt eine Verlängerung der Beobachtungsfensterlänge bei der **DAU** nicht zu einer Vergrößerung der durch die Zeitstempelung verursachten Unsicherheitseinflüsse.

Wird die simulierte Beobachtungsfensterlänge auf 100 s verlängert (siehe Abbildung 4.18), zeigt sich auch für den **MPU-9250** eine Amplitudenabnahme auf 95 % bei 500 Hz mit einem 2σ -Konfidenzintervall von 90 % bis 99,5 %.

4.6 Vergleich von Multisinus-Approximation und DFT

Die Abbildungen 4.19a und 4.19b zeigen die **DFT** und Multisinus-Approximationen (siehe Abschnitt 2.4.2) beispielhaft für zwei Messungen mit 80 Hz Anregungsfrequenz und 30 s Beobachtungsdauer. Die Multisinus-Approximationsergebnisse sind in Grün dargestellt. Die Frequenzen für die Multisinus-Approximation wurden so gewählt, dass der 80-Hz-Grundton sowie die ersten vier Oberstöne erfasst werden. Die tatsächliche Signalfrequenz wurde durch **S3PSA** aus der Phasenabweichung $\frac{\Delta\varphi}{\Delta t}$ bestimmt. Neben jeder Harmonischen wurden 100 weitere Linien approximiert. Der Frequenzabstand wurde entsprechend der Gleichung 2.33 zu $\Delta f = \frac{1}{T_W}$ gewählt. In den Multisinus-Approximationen wird das Beschleunigungssignal als eine scharfe Linie aufgelöst. Je nach Frequenzabstand ist das **SNR** 1000 bis 10000. Die Rauschamplitude in den durch Multisinus-Approximation ausgewerteten Daten beträgt für beide Sensoren etwa $5 \cdot 10^{-4} \text{ m/s}^2$. Um den Grundton und die erste Harmonische kann bei der **IMU** **MPU-9250** eine vermutlich aus einer **Amplitudenmodulation (AM)** des Anregungssignals stammende Signalkomponente mit einer Bandbreite von 0,4 Hz und einer Amplitude von 10^{-1} m/s^2 bzw. $5 \cdot 10^{-3} \text{ m/s}^2$ für die erste Harmonische erkannt werden.

³1000 Hz verletzen bei einer näherungsweise äquidistanten Abtastung mit 2 kHz das *Nyquist-Shannon-Theorem*; daher werden 990 Hz simuliert.

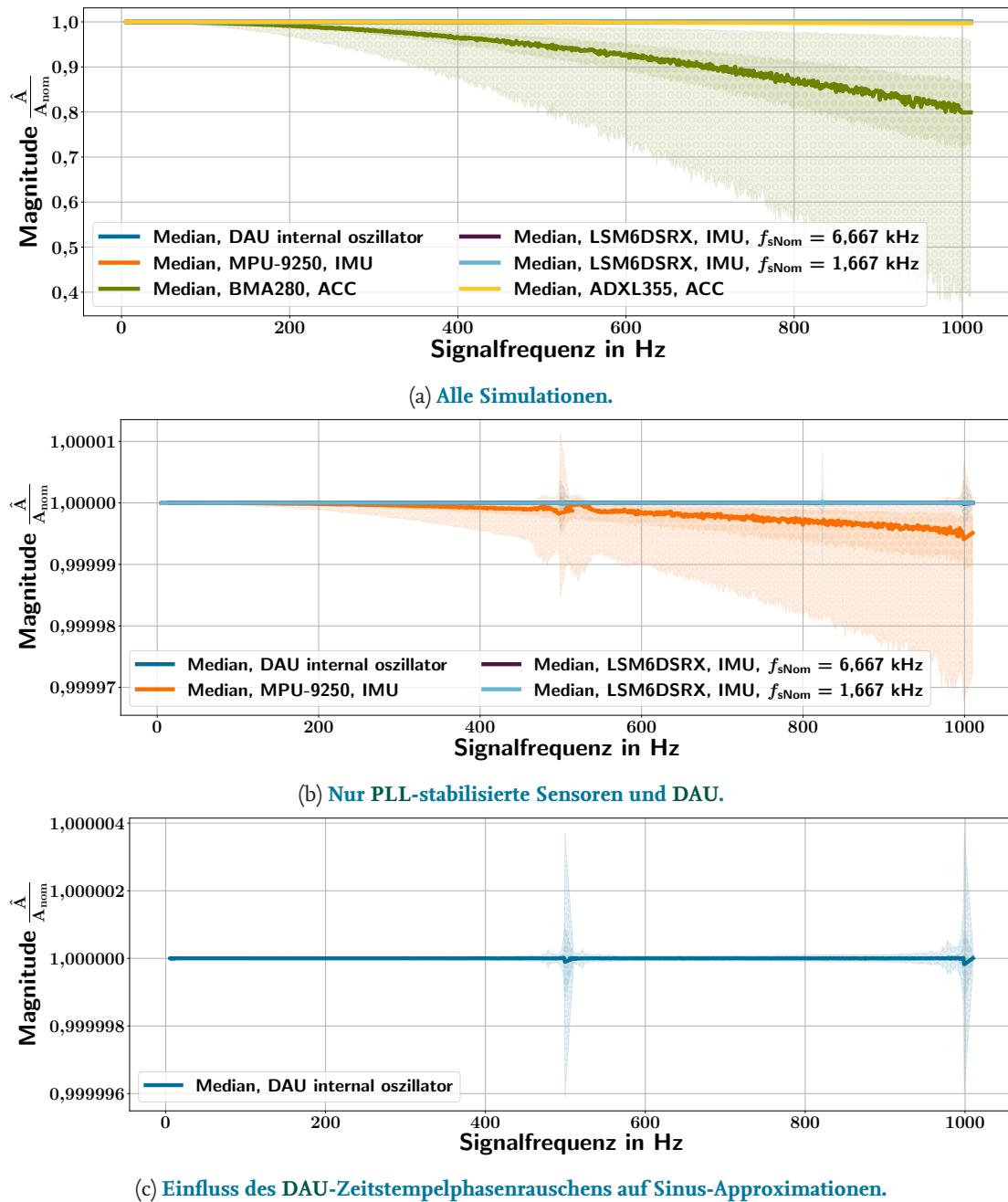


Abbildung 4.17: **Monte-Carlo-Simulation der Amplitudenabnahme durch Phasenrauschen bei 10 s Beobachtungsfensterlänge.** Die 1σ (dunkler) und 2σ breiten Konfidenzintervalle der 1000 Simulationen sind um die Medianlinie transparent dargestellt. Durch das relativ starke Phasenrauschen kommt es beim Sensor **BMA280** bei einer Beobachtungsfensterlänge von 10 s zu deutlichen Amplitudenfehlern, wenn äquidistantes Abtasten angenommen wird, trotz Verwendung der korrekten mittleren Abtastrate \bar{f}_s . Die durch das Phasenrauschen der DAU-Zeitstempelung verursachte Amplitudenabweichung ist mit einem relativen Einfluss von $< 10^{-6}$ vernachlässigbar.

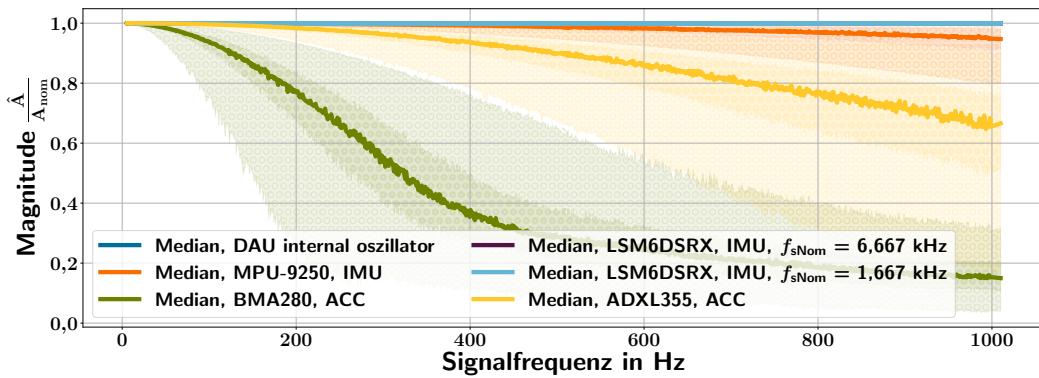


Abbildung 4.18: **Monte-Carlo-Simulation der Amplitudenabnahme durch Phasenrauschen für 100 s Beobachtungsfensterlänge.** Hierbei kommt es auch bei der **IMU MPU-9250** bei höheren Signalfrequenzen zu einer Amplitudenabnahme auf bis zu im Median 95 % bei 500 Hz. Die 1σ - und 2σ -Intervalle der 1000 Simulationen sind transparent dargestellt.

Neben einer **AM** des Anregungssignales kommen auch Störungen in der digitalen Filterung innerhalb der Sensoren durch das Abtastphasenrauschen als Quelle für diese breitbandige Signalkomponenten in Betracht. Die Harmonischen können in der Multisinus-Approximation mit einem **SNR** von 10 bis 100 immer aufgelöst werden. Auf Grund der deutlich geringeren Amplitude ist um die höheren Harmonischen weder beim *BMA280* noch beim *MPU-9250* die **AM**-/Filter-Komponente erkennbar. Bei der in **Blau** dargestellten **DFT** kommt es durch das Abtastphasenrauschen und den Leck-Effekt (*Spectral leakage*) zu signifikanten Störungen. Die durch das Abtastphasenrauschen verursachte Signalkomponente dominiert zwischen 0 Hz und der ersten Harmonischen bei beiden Sensoren das gesamte Spektrum. Um die Auswirkungen des Leck-Effekts zu minimieren, wurde die verwendete Zahl der Datenpunkte N_{LowLeak} aus allen Datenpunkten N so gewählt, dass die Abweichung Δf zwischen der tatsächlichen Signalfrequenz f_{Sig} und der nächsten **FFT**-Line f_{DFT} möglichst klein wird. Dafür wurde folgende Gleichung minimiert.

$$\Delta f = |f_{\text{Sig}} - f_{\text{DFT}}| = \min \underbrace{\left(f_{\text{Sig}} \bmod \frac{\bar{f}_s}{N} \right)}_{\text{positive Abweichungen}}, \underbrace{\left| f_{\text{Sig}} \bmod \left(\frac{\bar{f}_s}{N} \right) - \frac{\bar{f}_s}{N} \right|}_{\text{negative Abweichungen}} \quad (4.25)$$

Hierbei wurde die durchschnittliche Abtastfrequenz \bar{f}_s der gesamten Messung verwendet. Die aus diesen Daten berechnete **DFT** ist in **Orange** dargestellt. Für beide Sensoren konnte so eine deutliche Reduktion des Leck-Effekts erreicht werden. Bei der **IMU MPU-9250** zeigt sich dies in einer deutlichen Abnahme der hochfrequenten Komponenten. Beim Spektrum des *BMA280* zeigt sich durch die Vermeidung des Leck-Effekts eine drastische Reduktion der Störungen. Da der Einfluss des *Broad-Band*-Abtastphasenrauschen kleiner als das Eigenrauschen des Sensors ist, kommt es für große Frequenzabstände nicht mehr zu einer Erhöhung der Rauschleistung.

Fazit Die Verwendung von Multisinus-Approximation mit Absolutzeitstempeln der Messwerte be seitigt den Einfluss des Abtastphasenrauschen nahezu vollständig. Es kommt nicht mehr zu einer Signalverbreiterung und auch zu keiner Amplitudenabnahme. Dadurch können auch die Obertöne ausgewertet werden. Nachteilig an diesem Verfahren ist alleine die signifikant höhere Rechenzeit der Multisinus-Approximation im Vergleich zur **DFT** [80].

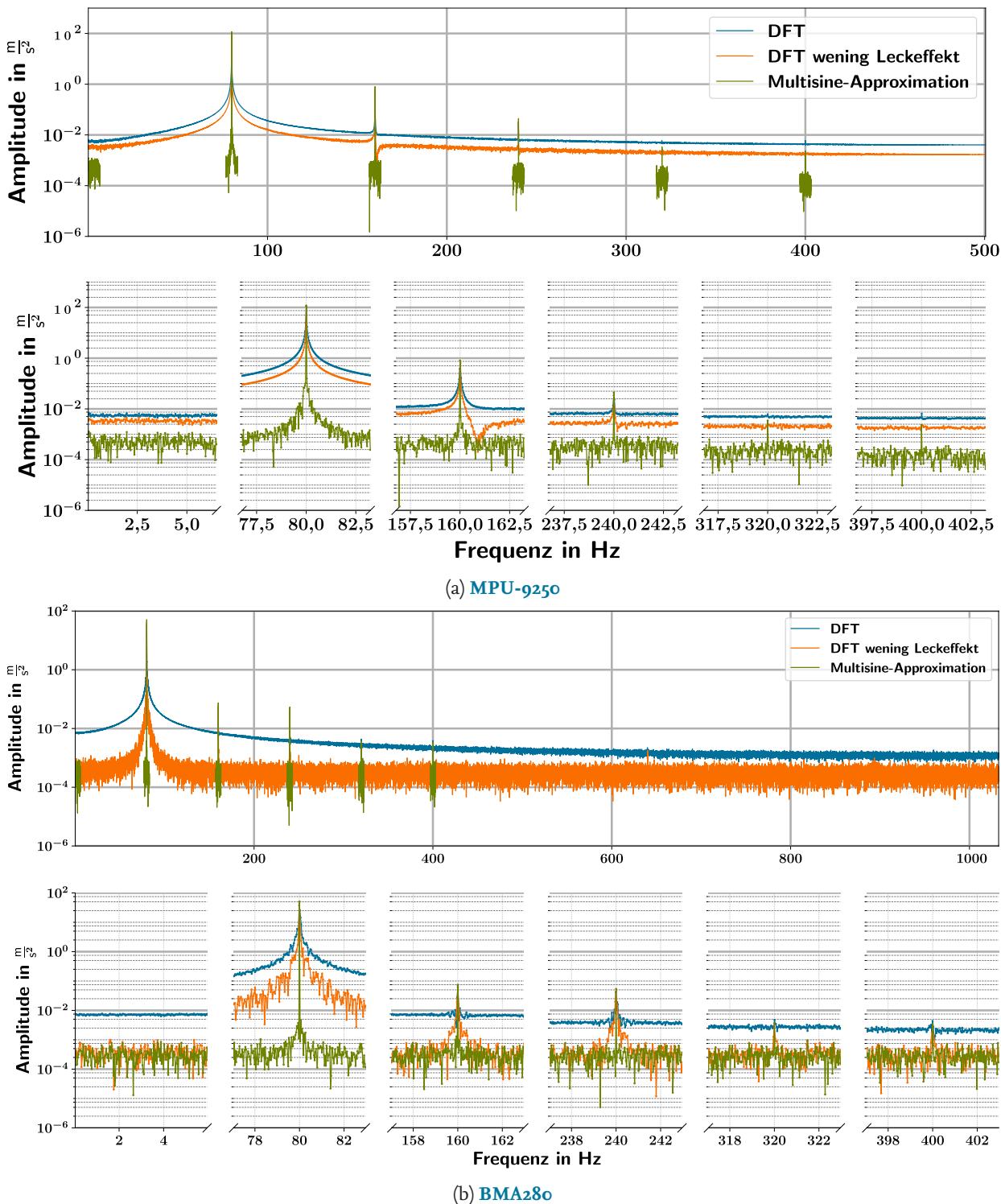


Abbildung 4.19: **Vergleich von Multisinus-Approximation und DFT für MPU-9250 und BMA280.** Die Anregungsfrequenz betrug 80 Hz bei einer Anregungsamplitude von 128 m/s^2 . Die nominale Abtastrate war 1000 Hz für MPU-9250 und 2000 Hz für BMA280. Die Messdauer betrug 30 s. Beide Sensoren zeigen deutlich die Effekte von *Spectral leakage* und Abtastphasenrauschen in den DFT-Spektren, während die Multisinus-Approximation korrekte Ergebnisse liefert. Um den Grundton übersteigt die Signalverbreiterung das von der Multisinus-Approximation bestimmte Rauschen um bis zu drei Größenordnungen. Die 3. und 4. Obertöne können nur in der Multisinus-Approximation aufgelöst werden.

5 Kalibrierung digitaler Sensoren

In diesem Kapitel wird die Erweiterung der in Abschnitt 2.2.1 beschriebenen, bestehenden analogen Kalibriersysteme (Analogue Calibration System, ACS) zur Kalibrierung digitaler Sensoren unter der Verwendung einer analogen Phasenreferenz vorgestellt.

Die Abbildung 5.1 zeigt, wie Abbildung 2.2 noch einmal schematisch den Aufbau eines analogen Kalibriersystems nach ISO 16063 [3].

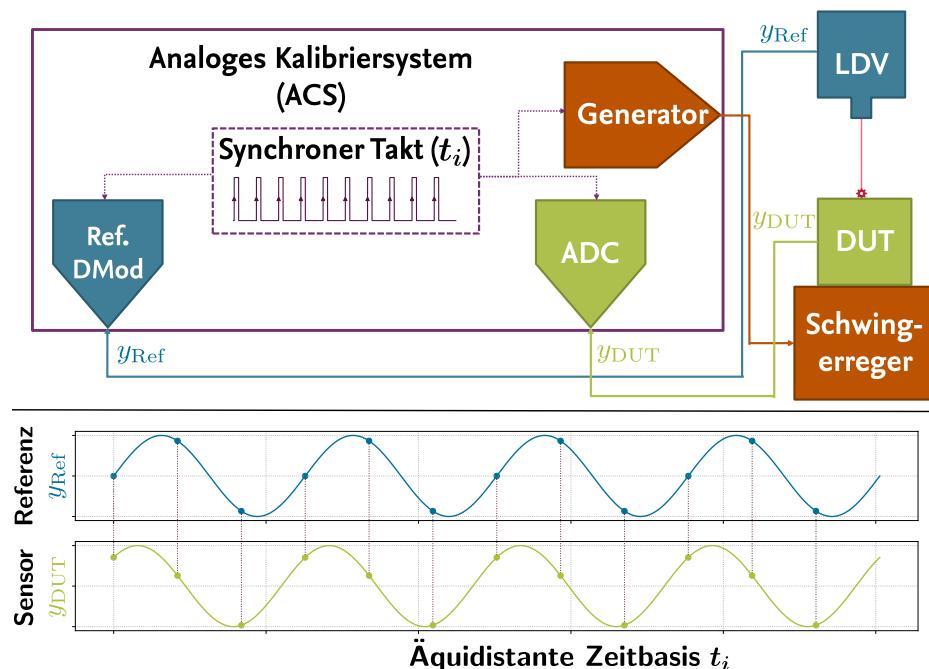


Abbildung 5.1: **Primäres einachsiges Beschleunigungskalibrierungssystem nach ISO 16063-11 [3] Methode 3: Sinus-Approximation.** Das DUT wird von einem Schwingerreger mit einer sinusförmigen Bewegung mit der Frequenz f beaufschlagt. Mittels LDV wird die momentane Beschleunigung $y_{Ref,f}(t_i)$ gegen eine Relativzeitbasis t_i abgetastet. Die vom DUT erfasste Beschleunigung $y_{DUT,f}(t_i)$ wird dazu synchron abgetastet.

Das Kalibriersystem erfasst die DUT- und LDV-Daten synchron mit Bezug auf eine interne Zeitbasis. Abweichungen der Oszillatorkreisfrequenz und das Phasenrauschen sind auf beiden Kanälen vollständig korriktiert und kürzen sich bei der Bestimmung der Übertragungsfunktion heraus. Die Systeme bieten üblicherweise keine genaue Absolutzeitstempelung für die Messwerte. Eine Integration der digitalen Schnittstellen ist bei kommerziellen Systemen nicht ohne Weiteres möglich.

5.1 Kalibrierung digitaler Sensoren

Zur Kalibrierung der digitalen Sensoren wird die in Abschnitt 3 vorgestellte **Data Acquisition Unit (DAU)** in Kombination mit einem bestehenden **ACS** verwendet. Abbildung 5.2 zeigt schematisch das für digitale Sensoren erweiterte Kalibriersystem. Der Phasenbezug zwischen beiden Systemteilen wird durch ein analoges Synchronisierungssignal hergestellt, welches von beiden Systemen erfasst wird. Dieses Synchronisierungssignal kann z. B. das Anregungssignal des Kalibriersystems sein oder, wie manche Systeme es bieten, ein **Constant Output Level Amplitude (COLA)**-Signal. Das Synchronisierungssignal muss die gleiche Frequenz und einen starren Phasenbezug zur mechanischen Anregung haben. Die Amplitude sollte möglichst groß sein, um die **ADCs** von **DAU** und **ACS** gut auszusteuern. Die Amplitude des Synchronisierungssignals wird nicht ausgewertet, da nur der Phasenbezug synchronisiert wird.

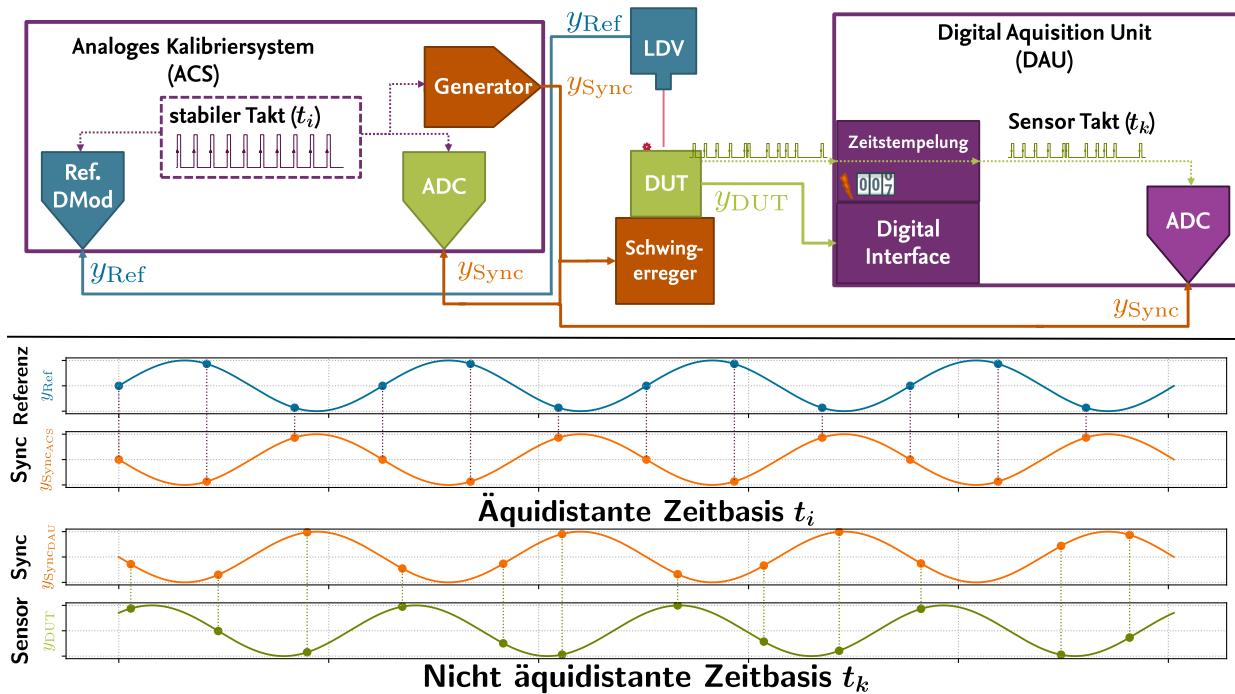


Abbildung 5.2: **Erweiterung eines analogen Kalibriersystems mit einer DAU**. Das **ACS** bestimmt die Anregungsamplitude \hat{a}_{ACS} sowie die Phasendifferenz $\varphi_{ACS}(f) - \varphi_{Sync_{ACS}}(f)$ zu einem Synchronisierungssignal y_{Sync} . Die **DAU** erfasst und zeitstempelt die Sensordaten y_{DUT} sowie das Synchronisierungssignal y_{Sync} .

Die **DAU** erfasst die Beschleunigungsmesswerte des Prüflings $y_{DUT,f}(t_k)$ an den nicht notwendigerweise äquidistanten Zeitpunkten t_k , in Abb. 5.2 **grün** dargestellt, außerdem synchron dazu das Synchronisierungssignal $y_{Sync_{DAU,f}}(t_k)$, in der Abbildung **orange**. Durch die **GNSS**-stabilisierte Zeitbasis der **DAU** werden die Zeitpunkte t_k mit Unsicherheiten im Bereich einiger 10 Nanosekunden bestimmt. Das Synchronisierungssignal wird auch in den **DUT-ADC** des **ACS** eingespeist und von diesem als $y_{Sync_{ACS,f}}(t_i)$ synchron zum Referenzsignal des **Laser-Doppler-Vibrometer** $y_{REF,f}(t_i)$ erfasst. Das **ACS** wird also so konfiguriert, dass es die Übertragungsfunktion zwischen Synchronisie-

rungssignal $y_{\text{Sync}_{\text{ACS}}}$ und Beschleunigung der Schwingarmatur y_{REF} bestimmt. In der Regel gibt das **ACS** nicht nur den Amplitudenübertragungskoeffizienten aus, sondern zusätzlich noch Mittelwert $\hat{a}_{\text{REF}}(f)$ und Standardabweichung $\sigma(\hat{a}_{\text{REF}}(f))$ der Anregungsamplitude.

Bei der Kalibrierung werden sukzessive monofrequente Sinusschwingungen angeregt. Mittels **S3PSA** werden aus den Zeitreihen die Amplituden und Initialphasen bestimmt. Für die Bestimmung des Betragsübertragungskoeffizienten $|S(f)|$ werden nur die Amplituden der Zeitreihen $y_{\text{DUT},f}(t_k)$ und $y_{\text{REF},f}(t_f)$ benötigt.

$$|S(f)| = \frac{\overline{\hat{a}_{\text{DAU}}(f)}}{\overline{\hat{a}_{\text{ACS}}(f)}} \quad (5.1)$$

Dabei werden die Mittelwerte der Amplituden unter Verwendung der **S3PSA** in der **DAU** und im **ACS** nach Gleichung 4.14 gebildet. Da die *Slices* im **ACS** und in den **DAU**-Daten nicht synchronisiert sind, kann nur der Quotient der Mittelwerte gebildet werden und nicht der Mittelwert der Quotienten; in diesem Fall würden sich die Einflüsse von Amplitudenschwankungen während der Messungen durch die Korrelation eliminieren.

Die Bestimmung der Phase ist etwas komplexer. Jeweils für das **ACS** und die **DAU** werden die Phasendifferenzen der Sensor- bzw. der Referenzsignale zum Synchronisierungssignal bestimmt.

Im **ACS** also $\overline{\varphi_{\text{REF}}(f) - \varphi_{\text{Sync}_{\text{ACS}}}(f)}$ und in der **DAU** entsprechend $\overline{\varphi_{\text{DUT}}(f) - \varphi_{\text{Sync}_{\text{DAU}}}(f)}$. Dabei werden die Mittelwerte der Differenzen bestimmt. Für die Phasenverzögerung des Sensors ergibt sich nun

$$\varphi(f) \approx \overline{\varphi_{\text{DUT}}(f) - \varphi_{\text{Sync}_{\text{DAU}}}(f)} - \overline{\varphi_{\text{REF}}(f) - \varphi_{\text{Sync}_{\text{ACS}}}(f)}. \quad (5.2)$$

Die Phasenverzögerung des **ADC** der **DAU** wurde kalibriert (siehe Abschnitt 3.3.3) und kann daher kompensiert werden. Für die Phasenverzögerung $\varphi(f)$ ergibt sich nun folgende Formel.

$$\varphi(f) = \overline{\varphi_{\text{DUT}}(f) - \varphi_{\text{Sync}_{\text{DAU}}}(f)} - \overline{\varphi_{\text{REF}}(f) - \varphi_{\text{Sync}_{\text{ACS}}}(f)} - \varphi_{\text{ADC}_{\text{DAU}}}(f) \quad (5.3)$$

5.2 Unsicherheitsbetrachtung nach GUM Typ A

Nach dem [Guide to the Expression of Uncertainty in Measurement \[10\] \(GUM\)](#) (siehe Abschnitt 5.1.2) [10] kann die kombinierte (Index: c) Messunsicherheit u_c einer aus N unabhängigen Einzelgrößen x_i berechneten Messgröße y unter Vernachlässigung von Korrelationen wie folgt genähert werden.

$$u_c^2(y) = \sum_{i=1}^N \left(\frac{\partial f}{\partial x_i} \right)^2 u^2(x_i) \quad (5.4)$$

Für die Standardabweichung der Magnitude der Transferfunktion nach Gleichung 2.12 ergibt sich danach die Näherung.

$$\sigma^2(|S(f)|) = \left(\frac{1}{\hat{a}_{\text{REF}}(f)} \right)^2 \cdot \sigma^2(\hat{a}_{\text{DUT}}(f)) + \left(\frac{\overline{\hat{a}_{\text{DUT}}(f)}}{\hat{a}_{\text{REF}}(f)^2} \right)^2 \cdot \sigma^2(\hat{a}_{\text{REF}}(f)) \quad (5.5)$$

mit den Standardabweichungen σ^2 der approximierten Amplituden $\sigma^2(\hat{a}_{\text{DUT}}(f))$ und $\sigma^2(\hat{a}_{\text{REF}}(f))$. Die Abbildungen 5.3a und 5.3b zeigen die kombinierte statistische Unsicherheit sowie die Beiträge für die Kalibrierung der [IMU MPU-9250](#) an der [PTB](#) und am [CEM](#). Für die Berechnung der Phasenverzögerung (Gleichung 5.3) ergibt sich folgende rein statistische Näherungsformel:

$$\sigma^2(\varphi(f)) = \sigma^2(\overline{\varphi_{\text{DUT}}(f) - \varphi_{\text{Sync}_{\text{DAU}}}(f)}) + \sigma^2(\overline{\varphi_{\text{REF}}(f) - \varphi_{\text{Sync}_{\text{ACS}}}(f)}) + u^2(\varphi_{\text{ADC}_{\text{DAU}}}(f)) \quad (5.6)$$

Die Abbildungen 5.4a und 5.4b zeigen die statistischen Unsicherheitsbeiträge der Phasenbestimmung für die Kalibrierung des Beschleunigungssensors der [IMU MPU-9250](#) an der [PTB](#) und am [CEM](#). Aus den Daten der [DAU](#) können im Gegensatz zu den Daten des [ACS](#) nicht nur die Standardabweichung der Phasendifferenz zwischen Signal und Synchronisierung $\sigma^2(\overline{\varphi_{\text{REF}}(f) - \varphi_{\text{Sync}_{\text{ACS}}}(f)})$, sondern auch die Standardabweichungen der Einzelbeiträge bestimmt werden. Die Streuung der Phasendifferenz $\sigma^2(\overline{\varphi_{\text{DUT}}(f) - \varphi_{\text{Sync}_{\text{DAU}}}(f)})$ setzt sich aus den Standardabweichungen der Phasen $\sigma(\varphi_{\text{DUT}}(f))$ und $\sigma(\varphi_{\text{Sync}_{\text{DAU}}}(f))$ zusammen. Dabei sind einige Störeinflüsse korreliert, wie z. B. Abweichungen der [DAU](#)-Zeitbasis. Hierdurch ist die Streuung der Phasendifferenz in der Regel kleiner als es die pythagoreische Addition der Einzelbeiträge erwarten ließe. Die bis hierher durchgeführte ausschließlich statische Unsicherheitsbetrachtung erfasst bei Weitem nicht alle Einflussgrößen. Die Norm ISO 16063-11:1999 [3] beinhaltet im Anhang 3 eine Tabelle mit einer Übersicht über die üblicherweise bei einer primären Beschleunigungskalibrierung auftretenden Unsicherheitseinflüsse.

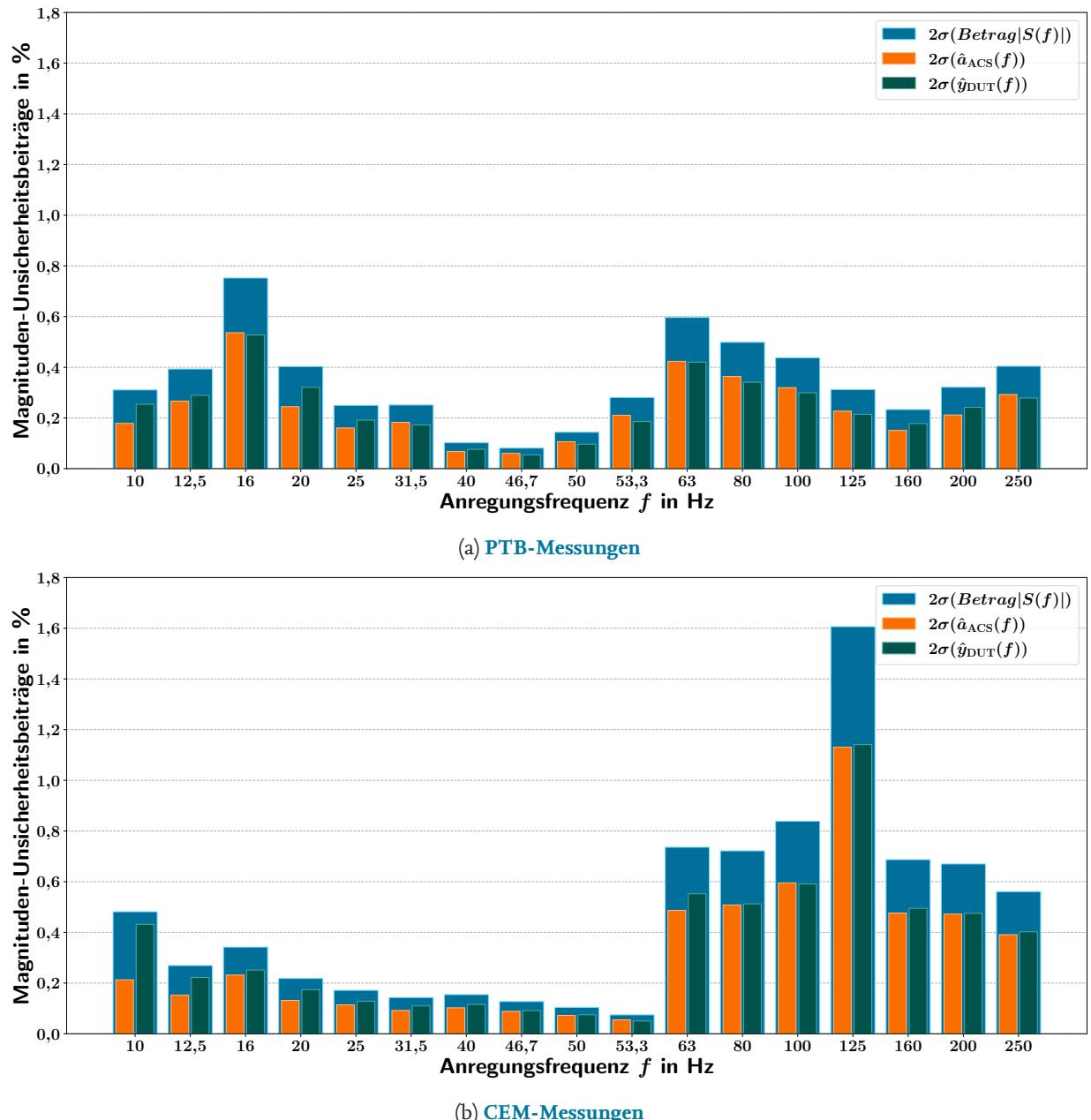


Abbildung 5.3: **Unsicherheitsbeiträge der Magnitudenberechnung für die PTB- und CEM-Messungen.** In Blau ist die gesamte statistische Messunsicherheit dargestellt. Beide Institute verwenden den gleichen Typ Schwingerreger SPEKTRA SE-09. Daher mussten bei niedrigen Frequenzen < 100 Hz die Beschleunigungsamplitude aufgrund der Schwingweglänge begrenzt werden (siehe Tab. 6.2), was zu einem schlechteren SNR führt. Ab 63 Hz setzen Kipp- und Tau-melbewegungen (*rocking motions*) ein, welche die Messungen störten. Die Magnitudenvarianzen von ACS (orange) und DAU und Sensor (grün) sind etwa gleich groß. Auch veröffentlicht in [85].

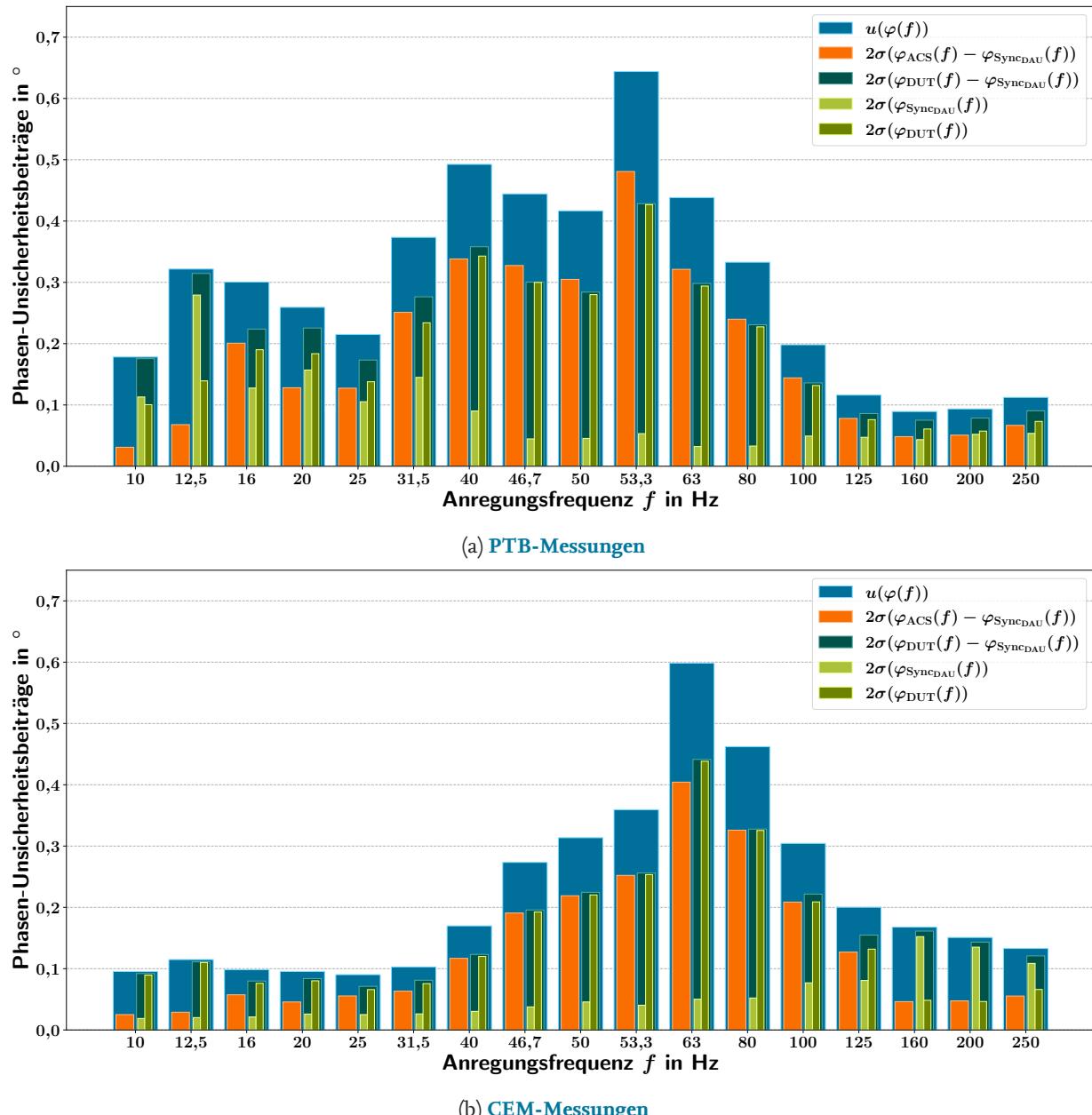


Abbildung 5.4: **Unsicherheitsbeiträge der Phasenberechnung für die PTB- und CEM-Messungen.** Blau: gesamte statistische Unsicherheit nach Gl. 5.6. Die Phasenunsicherheit der DAU-ADC-Kalibrierung ist aufgrund ihres vernachlässigbar kleinen Beitrags nicht gezeigt. Der aus dem ACS stammende Beitrag ist orange dargestellt. Der Unsicherheitsbeitrag aus der DAU (dunkelgrün) setzt sich aus den beiden teilweise korrelierten Einzelbeiträgen von Sensor (grün) und analoger Phasenreferenz (hellgrün) zusammen. Das an der PTB als analoge Phasenreferenz verwendete Schwingerreger-Anregungssignal hat für niedrigere Frequenzen eine kleinere Amplitude und daher ein schlechteres SNR. Für das COLA-Signal am CEM ist dies auf Grund der konstanten Amplitude nicht der Fall. Auch veröffentlicht in [85].

Die Tabelle 5.1 zeigt diese Einflussgrößen, erweitert um die zusätzlichen Beiträge für die Kalibrierung von digitalen **MEMS**-Sensoren nach der in dieser Arbeit vorgestellten Methodik. In der Norm wird empfohlen, zur Aufstellung eines Unsicherheitsbudgets die Einflussgrößen pythagoreisch zu addieren.

$$u_{c,rel}(S) = \frac{u_c(S)}{S} = \frac{1}{S} \sqrt{\sum_i u_i^2(S)} \quad (5.7)$$

Dabei steht $u_i(S)$ generisch für die i -te Unsicherheitskomponente der Magnitude S . wird anschließend mit $k = 2$ auf 95 % Koinzidenz erweitert, um die erweiterte Messunsicherheit zu erhalten.

$$U_{rel}(S) = k u_{c,rel}(S) \quad (5.8)$$

Die ersten drei Einflüsse $u(\hat{u}_V), u(\hat{u}_F)$ und $u(\hat{u}_D)$ betreffen die Digitalisierung der analogen Sensorsignale, welche vollständig innerhalb der **MEMS**-Sensoren stattfindet. Bei der Kalibrierung müssen diese Einflüsse daher zu einer gemeinsamen, das Sensorrauschen beschreibenden Größe zusammengefasst werden, da keine anderen Informationen zugänglich sind.

Der 4. Einfluss von Kipp- und Taumelbewegungen $u(\hat{u}_T)$ betrifft die **MEMS**-Sensoren unverändert. Die Sensoren sind zwar leichter, haben dafür aber wesentlich mehr Anschlussleitungen, und zumindest die in dieser Arbeit verwendete Sensorhalterung hatte keine radialsymmetrische Masseverteilung, was die Wahrscheinlichkeit für Kipp- und Taumelbewegungen erhöht.

Da das **LDV** zur Messung der Beschleunigungsreferenz unverändert verwendet wird, sind die 5. bis 7. Einflüsse $u(\hat{\phi}_{M,Q}), u(\hat{\phi}_{M,F})$ und $u(\hat{\phi}_{M,VD})$ weiterhin relevant.

Die **MEMS**-Sensoren werden in verschiedenen kleinen Gehäusen gefertigt, z. B. als **Quad-Flat No-Leads (QFN)** aus in der Regel schwarzem Epoxidharz. Auf diesem Gehäuse muss für die **LDV**-Messung eine reflektierende Folie angebracht werden. z.B. durch Verformung der Kelbschicht, zu unerkannten relativen Bewegungen zwischen Oberfläche und Folie kommen. Bei hermetisch abgeschlossenen **Ceramic Leadless Chip Carrier (CLCC)**-Gehäusen ist der Metalldeckel meist nicht mit dem Sensorelement verbunden und kann unerwünschte Resonanzen (wenn auch erst bei sehr hohen Frequenzen) zeigen. Derartige Effekte treten bei **MEMS**-Sensoren zusätzlich zu den unter 8. als $u(\hat{\phi}_{M,MD})$ beschriebenen relativen Bewegungen auf.

Die 9. und 10. Einflüsse $u(\hat{\phi}_{M,PD})$ und $u(\hat{\phi}_{M,RE})$ bleiben unverändert bestehen, da sie das **ACS** betreffen.

Durch die **GNSS**-Synchronisation in der **DAU** kann die tatsächliche Signalfrequenz sehr genau bestimmt werden, so dass dieser Unsicherheitsbeitrag (11.) vernachlässigbar niedrig wird.

Der zufällige Einfluss auf die Messung (12. Einfluss) $u(S_{RE})$ unverändert relevant.

Durch die Zeitabweichungen der Absolutzeitstempel ergibt sich grundlegend eine zu niedrige Amplitude (13. Einfluss) $u(t_k)$ (siehe Abschnitt 4.5.2). Bei Verwendung der **GNSS**-Synchronisation ist dieser Unsicherheitsbeitrag aber mit $< 10^{-5}$ vernachlässigbar klein.

Tabelle 5.1: Zu betrachtende Unsicherheitsbeiträge für die Magnitude in Anlehnung an ISO 16063-11:1999

A.1.3.1 [3]

i	Standard-unsicherheits-komponente $u(x_i)$	Unsicherheitsquelle	Unsicherheits-beitrag $u_i(y)$	Relevant für digitale Sensoren
1	$u(\hat{u}_V)$	Messung der Ausgangsspannung des Beschleunigungsaufnehmers (Wellenformrekorder, z. B. ADC-Auflösung)	$u_1(S)$	kombiniertes Sensorrauschen
2	$u(\hat{u}_F)$	Einfluss der Spannungsfilterung auf die Messung der Ausgangsspannung des Beschleunigungsaufnehmers (Frequenzbandbegrenzung)	$u_2(S)$	kombiniertes Sensorrauschen
3	$u(\hat{u}_D)$	Einfluss von Spannungsstörungen auf die Messung der Ausgangsspannung des Beschleunigungsaufnehmers (z. B. Brummen und Rauschen)	$u_3(S)$	kombiniertes Sensorrauschen
4	$u(\hat{u}_T)$	Einfluss von transversalen, kippenden, taumelnden und biegenden Beschleunigungen auf die Messung der Ausgangsspannung des Beschleunigungsaufnehmers (Transversalempfindlichkeit)	$u_4(S)$	Sensoroberfläche ist nicht reflektierend oder kann Resonanzen haben
5	$u(\hat{\phi}_{M,Q})$	Einfluss der Störung des Quadratur-Ausgangssignals des Interferometers auf die Messung der Phasenamplitude (z. B. Versatz, Abweichung der Spannungsamplitude, Abweichung vom nominalen Winkelunterschied von 90°)	$u_5(S)$	unverändert
6	$u(\hat{\phi}_{M,F})$	Einfluss der Signalfilterung des Interferometers auf die Messung der Phasenamplitude (Frequenzbandbegrenzung)	$u_6(S)$	unverändert
7	$u(\hat{\phi}_{M,VD})$	Einfluss von Spannungsstörungen auf die Messung der Phasenamplitude (z. B. zufälliges Rauschen in den fotoelektrischen Messketten)	$u_7(S)$	unverändert
8	$u(\hat{\phi}_{M,MD})$	Einfluss von Bewegungsstörungen auf die Messung der Phasenamplitude (z. B. Drift, relative Bewegung zwischen der Referenzfläche des Beschleunigungsaufnehmers und dem vom Interferometer erfassten Punkt)	$u_8(S)$	unverändert
9	$u(\hat{\phi}_{M,PD})$	Einfluss von Phasenstörungen auf die Messung der Phasenamplitude (z. B. Phasenrauschen der Interferometersignale)	$u_9(S)$	unverändert
10	$u(\hat{\phi}_{M,RE})$	Restliche interferometrische Effekte auf die Messung der Phasenamplitude (Interferometerfunktion)	$u_{10}(S)$	unverändert
11	$u(f_{FG})$	Messung der Schwingungsfrequenz (Frequenzgenerator und -anzeiger)	$u_{11}(S)$	nahezu irrelevant aufgrund von GNSS-Sync.
12	$u(S_{RE})$	Restliche Effekte auf die Empfindlichkeitsmessung (z. B. zufälliger Effekt bei Wiederholungsmessungen; experimentelle Standardabweichung des arithmetischen Mittels)	$u_{12}(S)$	unverändert
13	$u(t_k)$	Einfluss der Unsicherheiten bei der Zeitstempelung (DAU Phasenrauschen bekannt aus Charakterisierung)	$u_{13}(S)$	neu
Relevanz des Einflusses		sehr stark	stark	für Messkette im Sensor
			unverändert	vernachlässigbar

Analog zur Magnitudenberechnung beinhaltet die Norm ISO 16063-11:1999 [3] im Anhang eine Tabelle für die minimal zu betrachtenden Unsicherheitsbeiträge bei der Bestimmung der Phasenverzögerung. Analog zum Unsicherheitsbudget der Magnitude ergibt die pythagoreische Summe der Einflüsse

$$u_c(\Delta\varphi) = \sqrt{\sum_i u_i^2(\Delta\varphi)} \quad (5.9)$$

erweitert mit $k = 2$ als

$$U(\Delta\varphi) = k \cdot u_c(\Delta\varphi) \quad (5.10)$$

die Messunsicherheitsabschätzung für die primäre Kalibrierung der Phasenverzögerung. Die 1. bis 3. Einflüsse $u(\varphi_{u,V})$, $u(\varphi_{u,F})$ und $u(\varphi_{u,D})$ betreffen wieder die Digitalisierung der Sensordaten und müssen daher aus Mangel an anderen Informationen unter dem neuen Punkt als kombiniertes Sensorrauschen zusammengefasst werden.

Die Kipp- und Taumelbewegungen (4. Einfluss) $u(\varphi_{u,T})$ beeinflussen die Phasenbestimmung genauso wie die Magnitudenbestimmung; die Erläuterungen dazu gelten folglich ebenso für die Phase. Die 5. bis 7. Einflüsse $u(\varphi_{s,Q})$, $u(\varphi_{s,F})$ und $u(\varphi_{s,VD})$ betreffen das ACS bzw. LDV und bleiben unverändert.

Durch Gehäuseresonanzen oder die applizierte Reflektorfolie kann nicht nur die Magnitude, sondern auch die Phase beeinträchtigt werden; der 8. Einfluss $u(\varphi_{s,MD})$ muss folglich genau betrachtet werden.

Die 9. bis 11. Einflüsse $u(\varphi_{s,PD})$, $u(\varphi_{s,RE})$ und $u(\Delta\varphi_{RE})$ bleiben unverändert bestehen.

Bei der Kalibrierung mit einem analogen Referenzsignal wird der DAU-ADC verwendet. Die zur

Kompensation verwendete Transferfunktion hat eine Unsicherheit, welche als 12. Einfluss $u(\Delta\varphi_{\text{DAU}_{\text{ADC}}})$ zusätzlich berücksichtigt werden muss.

Das bei der Erfassung der analogen Phasenreferenz auftretende Rauschen führt zu zusätzlichen Phasenabweichungen; dieser Einfluss wird als 13. Einfluss $u(\varphi_{\text{Sync}_{\text{DAU}}})$ berücksichtigt.

Die Zeitabweichungen der Absolutzeitstempelung sowie Signallaufzeiten führen zu weiteren, wenn auch kleinen, Phasenabweichungen, die als 14. Unsicherheitseinfluss $u(t_k)$ berücksichtigt werden.

Tabelle 5.2: Zu betrachtende Unsicherheitsbeiträge für die Phase in Anlehnung an ISO 16063-11:1999 A.1.3.1 [3]

i	Standard-unsicherheitskomponente $u(x_i)$	Unsicherheitsquelle	Unsicherheitsbeitrag $u_i(y)$	Relevant für digitale Sensoren
1	$u(\varphi_{u,V})$	Phasenmessung des Beschleunigungssensors (Wellenformrekorder, z. B. ADC-Auflösung)	$u_1(\Delta\varphi)$	kombiniertes Sensorrauschen
2	$u(\varphi_{u,F})$	Einfluss des Spannungsfilters auf die Phasenmessung des Beschleunigungssensors (Frequenzbandbegrenzung)	$u_2(\Delta\varphi)$	kombiniertes Sensorrauschen
3	$u(\varphi_{u,D})$	Einfluss von Spannungsstörungen auf die Phasenmessung des Beschleunigungssensors (z. B. Brummen und Rauschen)	$u_3(\Delta\varphi)$	kombiniertes Sensorrauschen
4	$u(\varphi_{u,T})$	Einfluss von transversalen, kippenden und biegenden Beschleunigungen auf die Phasenmessung des Beschleunigungssensors (Querempfindlichkeit)	$u_4(\Delta\varphi)$	Sensoroberfläche ist nicht reflektierend oder kann Resonanzen haben
5	$u(\varphi_{s,Q})$	Einfluss von Störungen des Interferometerquadratursignals auf die Verschiebungs-Phasenmessung (z. B. Versätze, Spannungsamplitudenabweichung, Abweichung vom 90° Nennwinkelunterschied)	$u_5(\Delta\varphi)$	unverändert
6	$u(\varphi_{s,F})$	Einfluss der Signalfilterung des Interferometers auf die Verschiebungsphasenmessung (Frequenzbandbegrenzung)	$u_6(\Delta\varphi)$	unverändert
7	$u(\varphi_{s,VD})$	Einfluss von Spannungsstörungen auf die Verschiebungsphasenmessung (z. B. zufälliges Rauschen in den fotoelektrischen Messketten)	$u_7(\Delta\varphi)$	unverändert
8	$u(\varphi_{s,MD})$	Einfluss von Bewegungsstörungen auf die Verschiebungsphasenmessung (z. B. Drift, relative Bewegung zwischen der Referenzfläche des Beschleunigungssensors und dem vom Interferometer erfassten Punkt)	$u_8(\Delta\varphi)$	unverändert
9	$u(\varphi_{s,PD})$	Einfluss von Phasenstörungen auf die Verschiebungsphasenmessung (z. B. Phasenrauschen der Interferometersignale)	$u_9(\Delta\varphi)$	unverändert
10	$u(\varphi_{s,RE})$	Restliche interferometrische Effekte auf die Verschiebungsphasenmessung (Interferometerfunktion)	$u_{10}(\Delta\varphi)$	unverändert
11	$u(\Delta\varphi_{RE})$	Restliche Effekte auf die Phasenverschiebungsmessung (z. B. zufälliger Effekt bei Wiederholungsmessungen, experimentelle Standardabweichung des arithmetischen Mittels)	$u_{11}(\Delta\varphi)$	unverändert
12	$u(\Delta\varphi_{\text{DAU}_{\text{ADC}}})$	Unsicherheit der Übertragungsfunktion der ADCs der DAU (bekannt aus der ADC-Kalibrierung)	$u_{12}(\Delta\varphi)$	neu
13	$u(\varphi_{\text{Sync}_{\text{DAU}}})$	Einfluss von Rauschen auf die bestimmte DAU-Sync-Signalphase (z. B. elektrisches Rauschen, Brummen, ADC-Quantisierungsfehler)	$u_{13}(\Delta\varphi)$	neu
14	$u(t_k)$	Einfluss der Unsicherheiten der Zeitstempelung (DAU-Phasenrauschen bekannt aus Charakterisierung)	$u_{14}(\Delta\varphi)$	neu
Relevanz des Einflusses				
sehr stark		stark	für Messkette im Sensor	unverändert gering
vernachlässigbar				

5.2.1 Anregungs- und Approximationsfrequenz

Die Zeitbasis der **DAU** und des **ACS** sind nicht synchronisiert. Nach ISO 16063-11 [55] sollte die erzeugte Anregungsfrequenz in einem Bereich von ± 5 ppm um den Nominalwert liegen. Bei der **S3PSA** innerhalb des **ACS** führen Zeitbasisfrequenzabweichungen zu keinen Problemen, da die Signalerzeugung und -erfassung vom selben Takt abgeleitet werden. Weicht die Signalfrequenz durch einen zu schnellen Oszillator etwas nach oben ab, weicht die Abtastrate ebenfalls um den gleichen Faktor ab. Da sowohl Frequenzabweichungen als auch Phasenrauschen von Anregung und Datenerfassung korreliert sind, liefert eine **S3PSA** mit nominaler Anregungsfrequenz und als äquidistant angenommener Abtastung immer das korrekte Ergebnis, unter der Annahme, dass die Abweichung der Sensitivität durch die Frequenzabweichung von $< \pm 5$ ppm nicht verändert wird. Da die **DAU** die Zeitstempel nicht gegen die **ACS**-Zeitbasis, sondern gegen ihre eigene Absolutzeitbasis vergibt, werden diese Anregungsfrequenzabweichungen nicht kompensiert. Die Abbildung 4.3a zeigt die Phasen der **S3PSA** einer realen Kalibriermessung bei einer nominalen Anregungsfrequenz von 250 Hz. Im Beobachtungszeitraum von etwa 20 s kommt es durch eine Abweichung der Anregungsfrequenz zu einer konstanten Änderung der Phase um etwa $+4^\circ$ (siehe Abschnitt 4.2.2). Das Synchronisierungssignal zeigt die gleiche kontinuierliche Phasenabweichung. Bei der Bildung der Phasendifferenz $\varphi_{\text{DUT}}(f) - \varphi_{\text{Sync}_{\text{DAU}}(f)}$ würden sich diese Abweichungen nahezu vollständig kompensieren, wenn erst die Differenz gebildet und dann gemittelt wird. Die durch die abweichende Approximationsfrequenz verursachte Amplitudenabweichung wird jedoch nicht kompensiert. Mit dem in Abschnitt 4.2.1 beschriebenen Verfahren können die tatsächlichen Signalfrequenzen sehr genau bestimmt werden. Eine anschließende **S3PSA** zeigt dann keine Phasen- oder Amplitudenabweichungen mehr (siehe Abbildung 4.3b). Da die Frequenzabweichungen Δf unter ± 5 ppm liegen, wird für die Angabe der Übertragungsfunktion die Nominalfrequenz f_{nom} anstelle von $f_{\text{nom}} + \Delta f$ verwendet.

6 Verifizierung des Kalibrierungskonzeptes

Zur Verifikation des in dieser Arbeit entwickelten Kalibrierverfahrens wurde ein internationaler Vergleich zwischen der **PTB** und dem spanischen **CEM** durchgeführt [85]. Die digitale Datenerfassung wurde jeweils mit zwei verschiedenen **DAUs** bewerkstelligt, deren **ADCs** in der PTB nach dem in Abschnitt 3.3.3 beschriebenen Verfahren kalibriert wurden.

6.1 Sensoaraufbau, Montage und Konfiguration

Als **DUT** wurde ein **MPU-9250** der Firma **TDK InvenSense** verwendet. Die Abbildung 6.1 zeigt das Blockdiagramm dieser **IMU**. Die drei Sensorelemente für die Beschleunigungsmessung sowie die Gyroskope sind auf einem Halbleiterchip (engl. *Die*) untergebracht, das Magnetometer auf einem anderen Chip innerhalb desselben **QFN**-Gehäuses. Das Magnetometer kommuniziert über **I²C** mit dem Interface-Controller ④ der **IMU**. Laut diesem Blockdiagramm besitzt jedes Sensorelement ① einen eigenen **ADC** ②. Die Sensorelemente für Beschleunigung und Winkelgeschwindigkeit besitzen auch eine Selbsttestschaltung, welche durch ein elektrisches Feld eine statische Sensorelementauslenkung erzeugen kann. Die von den Sensor-**ADCs** erfassten Messwerte werden in einem **Signal-Conditioning-Block** ③ digital gefiltert. Die genaue Funktion der Signalkonditionierung ist undokumentiert. Allein die Registerkonfigurationen aus dem Datenblatt (siehe Tabelle 6.1) ermöglichen gewisse Rückschlüsse auf die Implementation der Signalkonditionierung.

Die **ADCs** arbeiten vermutlich mit einer Abtastfrequenz von mehr als 4 kHz. Wird das digitale Tiefpassfilter (**DLPF**) nicht verwendet (Register **ACCEL_FCHOICE=0**), wird durch eine Dezimierung im Dec1-Block die Datenrate auf nominal 4 kHz reduziert. Bei aktiver digitaler Tiefpassfilterung (Register **ACCEL_FCHOICE=1**) können verschiedene Filterkonfigurationen mit unterschiedlichen Grenzfrequenzen, Gruppenlaufzeiten und Datenraten ausgewählt werden. Außerdem kann statt des **DLPF** ein anderes Dezimationsfilter mit einer Datenrate von 1 kHz als Block Dec2 ausgewählt werden. Für die Kalibriermessungen wurde die Konfiguration **ACCEL_FCHOICE=1** sowie **A_DLPF_CFG=0b000=0** verwendet. Dies entspricht einem aktiven **DLPF** mit 1 kHz nominaler Datenrate und einer Grenzfrequenz von 218,1 Hz bei einer Gruppenlaufzeit von 1,88 ms (in Tabelle 6.1 grün markiert). Neben der Datenrate lässt sich auch der Messbereich des Beschleunigungssensors über das Register **ACCEL_FS_SEL** konfigurieren. Dabei sind folgende Messbereiche wählbar: **0b00** $\hat{=}$ $\pm 2 \text{ g}^1$, **0b01** $\hat{=}$ $\pm 4 \text{ g}^1$, **0b10** $\hat{=}$ $\pm 8 \text{ g}^1$ oder **0b11** $\hat{=}$ $\pm 16 \text{ g}^1$. Für die Kalibrierungen wurde der größte Messbereich **ACCEL_FS_SEL = 0b11** $\hat{=}$ $\pm 16 \text{ g}^1$ gewählt.

¹Wie bei vielen **MEMS**-Beschleunigungssensoren üblich, spezifiziert der Hersteller den Skalenbereich und die Sensitivität nicht in **SI**-Einheiten, sondern in Vielfachen einer Normalbeschleunigung **g**. Wir nehmen hier $g = g_n = 9,81 \frac{\text{m}}{\text{s}^2}$ an.

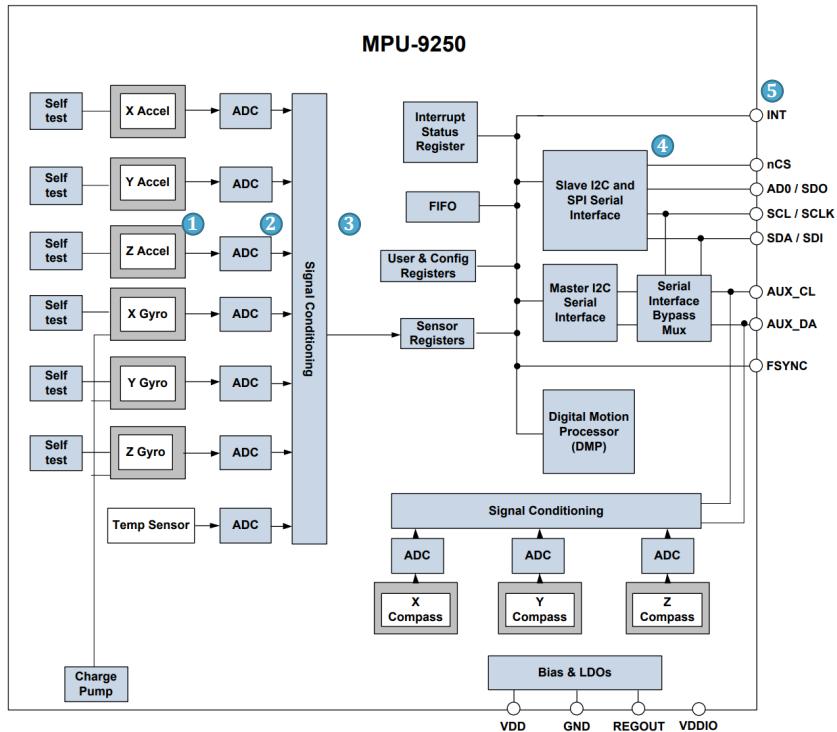


Abbildung 6.1: **Blockdiagramm der IMU MPU-9250.** Die IMU kann mit den analogen Sensorelementen ① 10 Messgrößen erfassen: Beschleunigung (X, Y, Z), Winkelgeschwindigkeit (X, Y, Z), magnetische Flussdichte (X, Y, Z) und Temperatur. Die analogen Messgrößen werden durch ADCs ② in digitale Werte überführt. Diese Werte werden anschließend in einem *Signal Conditioner* ③ tiefpassgefiltert. Über das digitale Interface ④ können die gefilterten Messwerte über I²C oder SPI ausgelesen werden. Über den Interrupt-Pin ⑤ signalisiert der Sensor, dass neue Messwerte in den Registern zum Auslesen bereitstehen. Das Magnetometer ist auf einem zweiten Halbleitersubstrat realisiert und über I²C an den Hauptsensor angebunden. Informationen und Bild aus [46].

Je nach Messbereich ändert sich die Quantisierungsauflösung, da die ADC-Auflösung immer 16 Bit beträgt. Der Sensor bietet außerdem die Möglichkeit, die Sensordaten mit einem *Offset* zu versehen, um eine Nullpunktsabweichung zu korrigieren. Diese Funktion wurde jedoch nicht verwendet. Der Interrupt-Ausgang ⑤ wurde so konfiguriert, dass dieser als **Data Ready Interrupt (DRI)** für die Beschleunigungsdaten fungiert. Der im Sensor integrierte FIFO-Ringspeicher wird nicht verwendet. Als Kommunikations-Interface ④ wird wegen der höheren möglichen Datenrate SPI verwendet. Die IMU MPU-9250 bietet ein FSYNC-Pin zur Synchronisierung der Datenerfassung mit einem externen Signal; der logische Zustand (0/1) dieses Pins kann, wenn konfiguriert, als LSB eines der Sensorkanäle (z. B. Temperatur) ausgegeben werden. Durch die Absolutzeitstempelung ist diese Art der Synchronisierung nicht nötig und wird daher nicht verwendet.

Die Konfiguration des Sensors wurde wie folgt gewählt:

- nominale Datenrate 1000 Hz
- Sensitivität $512 \frac{\text{LSB}}{\text{g}} \approx 52,191 \frac{\text{LSB}}{\frac{\text{m}}{\text{s}^2}}$
- DLPF-Grenzfrequenz 218,4 Hz
- FIFO deaktiviert
- Skalenbereich $\pm 16 \text{ g}^2 \approx \pm 156,9 \frac{\text{m}}{\text{s}^2}$
- Interruptquelle DRI RAW_RDY_EN=1
- Auflösung 16 LSB
- Versorgungsspannung 3,3 V

Tabelle 6.1: Mögliche Datenraten und Filtereinstellungen der IMU MPU-9250 [46]

Config-Register		Ausgabe		Filter-Koeffizienten		Rauschen
ACCEL_FCHOICE	A_DLDPF_CFG	3 dB BW	Datenrate	Filter	Gruppenlaufzeit	Rausch-PSD
	[2:0]	(Hz)	(kHz)	Block	(ms)	$\left(\frac{\text{mms}^{-2}}{\sqrt{\text{Hz}}} \right)$
0	x	1046	4	DEC1	0,503	300
1	0	218,1 ²	1	DLPF	1,88 ²	2,950
1	1	218,1 ²	1	DLPF	1,88 ²	2,950
1	2	99	1	DLPF	2,88	2,950
1	3	44,8	1	DLPF	4,88	2,950
1	4	21,2	1	DLPF	8,87	2,950
1	5	10,2	1	DLPF	16,83	2,950
1	6	5,05	1	DLPF	32,48	2,950
1	7	420	1	DEC2	138	2,950

6.1.1 Sensormontage

Für diese Arbeit wurde ein MPU-9250-Sensor, der auf einer kommerziell erhältlichen Entwicklungsplatine (Breakout Board, BoB) verbaut ist, verwendet. Auf dem BoB befindet sich neben dem $3 \text{ mm} \times 3 \text{ mm} \times 1 \text{ mm}$ großen Sensor im QFN-24 Gehäuse (Abb. 6.2 ①) ein LDO-Spannungsregulator zur Erzeugung der 3,3 V Versorgungsspannung aus 5 V sowie Pufferkondensatoren und Serienwiderstände. Zur Befestigung des Sensors sind Löcher in der Platine vorhanden. An einer Seite des BoB befinden sich 10 Lötkontakte im Rastermaß von $1/10" = 2,54 \text{ mm}$ zum Anschluss einer Stifteleiste oder zum Anlöten von Drähten. Für die Messungen wurden sieben ca. 20 cm lange Drähte angelötet. Das Auxiliary-I²C-Interface sowie der FSYNC-Pin bleiben ungenutzt.

Zur Montage auf dem Swingerreger vom Typ Spektra SE-09 der Kalibriersysteme (von PTB und CEM) wurde das BoB mit Cyanoacrylat-Kleber auf einem eloxierten Aluminiumblock befestigt. Das Kleben führt zu einer flächigen, relativ steifen Verbindung mit dem Aluminiumblock. Es wird daher vermutlich zu deutlich weniger Kippschwingungen kommen, als es bei der vorgesehenen Verschraubung mit nur zwei Schrauben der Fall gewesen wäre.

²Wie bei vielen MEMS-Beschleunigungssensoren üblich, spezifiziert der Hersteller den Skalenbereich und die Sensitivität nicht in SI-Einheiten sondern in Vielfachen einer Normalbeschleunigung g . Wir nehmen hier $g = g_n = 9,81 \frac{\text{m}}{\text{s}^2}$ an.

²Die Konfigurationen A_DLDPF_CFG=0b000=0 und A_DLDPF_CFG=0b001=1 sind laut Datenblatt identisch.

Durch die Eloxierung ist die Oberfläche des Aluminiumblocks elektrisch isolierend, was Kurzschlüsse an den Datenleitungen verhindert. Der Aluminiumblock hat die Maße 16 mm × 26 mm × 5 mm; an der Unterseite befindet sich ein 1/4"-28-UNF-Gewinde zur Befestigung an der Schwingarmatur. Die Masse der Sensorplatine mit dem Montageblock beträgt etwa 10 g.

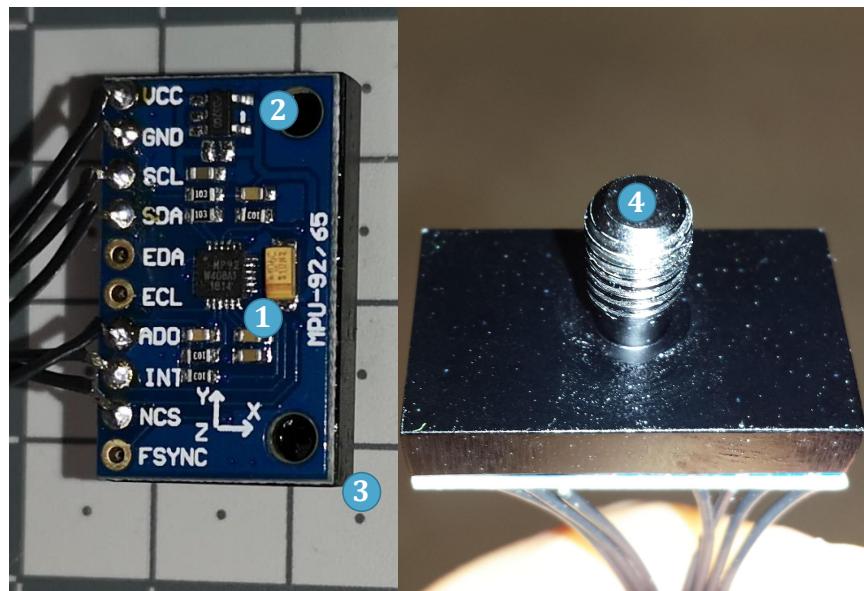


Abbildung 6.2: **MPU-9250-BoB auf Aluminiumblock mit UNF 1/4"-28-Gewinde.** Der Sensor ① im 3 mm × 3 mm × 1 mm QFN-24 Gehäuse ist in der Mitte des BoB montiert. Die 3,3 V Versorgungsspannung des Sensors wird von einem LDO-Spannungsregler ② aus 5 V erzeugt. Der Aluminiumblock ③ hat die Maße 16 mm × 6 mm × 5 mm, das Gewicht beträgt ca. 10 g; das 1/4"-28-UNF-Gewinde ④ befindet sich direkt unter dem Sensor.

6.2 Schwingungsanregung und Testablauf

Die Abbildung 6.3 zeigt eine DAU und den montierten MPU-9250-Sensor auf einem Schwingerregler vom Typ *Spektra SE-09*. Die Tabelle 6.2 zeigt die für die Kalibrierung verwendeten Amplituden und Frequenzen. Die Frequenzen wurden im Wesentlichen den Terzfolgen nach ISO 266 [47] zwischen 10 Hz und 250 Hz entnommen. Die Frequenzen 46,7 Hz und 53,3 Hz wurden zusätzlich eingefügt, da bei der Netzfrequenz von 50 Hz oft eingekoppelte Störungen auftreten. Für niedrige Frequenzen (<125 Hz) begrenzt die verfügbare Weglänge des Schwingerregers von etwa 8 mm die möglichen Beschleunigungsamplituden. Für höhere Frequenzen wurde die Beschleunigungsamplitude \hat{a}_{nom} so gewählt, dass eine Aussteuerung des Messbereichs von 80 % erreicht wird. Für Messungen mit unterschiedlichen Montagewinkeln mithilfe von unterschiedlich dicken Unterlegscheiben (siehe Abschnitt 6.4.3) wurde, aufgrund der verringerten mechanischen Stabilität durch die Unterlegscheiben, \hat{a}_{nomRed} auf $100 \frac{m}{s^2}$ begrenzt. Die Unterlegscheiben sind nötig, da das Gewinde fest am Aluminiumblock montiert ist und daher sonst der Drehwinkel nicht einstellbar wäre.

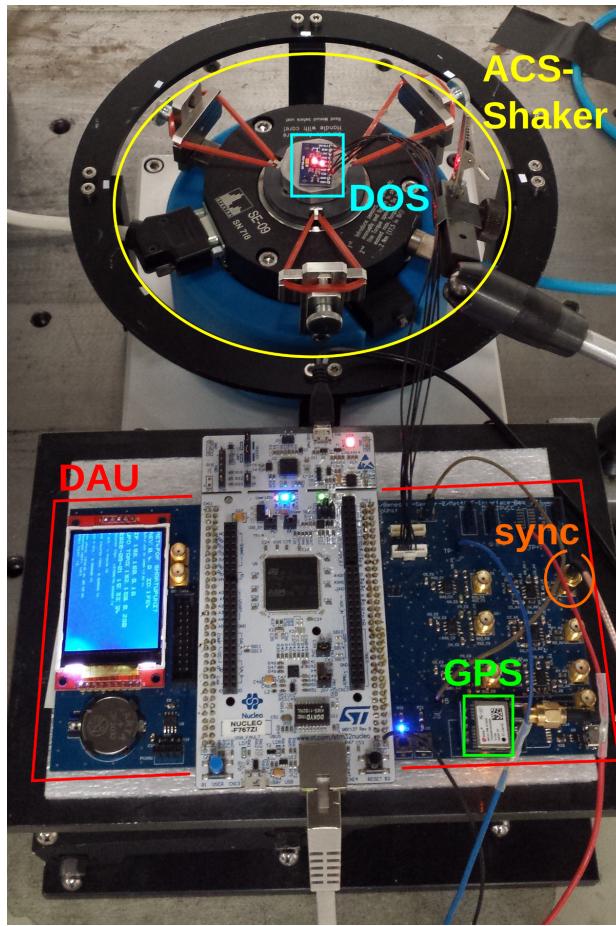


Abbildung 6.3: **HF-B-NME der PTB mit DAU und MPU-9250.** Auf dem Schwingerreger (gelb) ist auf einen Aluminiumblock das BoB mit dem Sensor montiert. Das Anregungssignal des Schwingerreger wird in die DAU zur Synchronisierung eingespeist. Der genaue Aufbau der DAU ist in Abbildung 3.2 gezeigt.

Tabelle 6.2: Nominelle Anregungsfrequenzen und Amplituden

Frequenz f in Hz	Amplitude \hat{a}_{nom} in $\frac{\text{m}}{\text{s}^2}$	reduzierte Amplitude \hat{a}_{nomRed} in $\frac{\text{m}}{\text{s}^2}$	Frequenz f in Hz	Amplitude \hat{a}_{nom} in $\frac{\text{m}}{\text{s}^2}$	reduzierte Amplitude \hat{a}_{nomRed} in $\frac{\text{m}}{\text{s}^2}$
10	2,8	2,0	53,3	59	44
12,5	2,7	2,0	63	115	45
16	5,5	2,0	80	119	46
20	5,5	3,9	100	123	95
25	11	7,9	125	127	97
31,5	11	8,1	160	128	98
40	23	17	200	127	99
46,7	59	43	250	127	99
50	59	43			

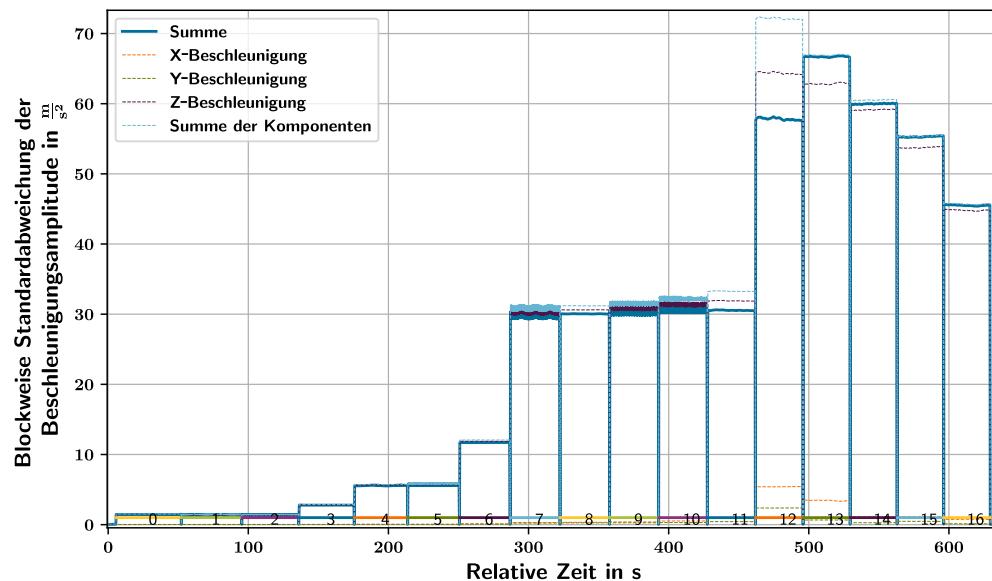


Abbildung 6.4: **Blockweise berechnete Standardabweichung zur Detektion des Anregungssignals, PTB-Messungen.**

Messungen. Die Standardabweichung der geometrischen Summe der Komponenten $\sigma(\sqrt{x^2 + y^2 + z^2})$ ist in Blau dargestellt. Liegt diese für fünf Blöcke mit je 100 Messwerten über dem Schwellwert von 1 m/s^2 , wird eine valide Anregung detektiert. Dargestellt sind verschiedene farbige Linien und die Nummern des Experiments. Durch die mögliche Phasenverschiebung kann die hellblaue dargestellte Summe der Standardabweichungen der Einzelkomponenten $\sigma(x) + \sigma(y) + \sigma(z)$ größer sein als die Standardabweichung ihrer geometrischen Summe. Bei den Anregungen 12 (100 Hz) und 13 (125 Hz) sind relativ große Querbeschleunigungen erkennbar.

Für die Kalibriermessungen wurden die in Tabelle 6.2 aufgelisteten Kombinationen aus Frequenz und Amplitude in aufsteigender Frequenzreihenfolge jeweils für etwa 20 s angeregt. Nach jeder Anregung folgte eine Pause von etwa 2 s. Die Kalibriermessungen wurden jeweils 10 Mal wiederholt. Der zeitliche Verlauf der Messungen ist in den blockweise berechneten Standardabweichungen in Grafik 6.4 erkennbar. Dieses Vorgehen war erforderlich, da das ACS und die DAU asynchron arbeiten. Anhand der Pausen in der Anregung können die einzelnen monofrequenten Anregungen in den DAU-Daten erkannt werden. Dafür wird jeweils blockweise (z. B. von je 64 Werten) die Standardabweichung berechnet. Da die Standardabweichung dem Effektivwert eines gleichanteilfreien Sinussignals entspricht, kann über einen Schwellenwert, z. B. $2 \frac{\text{m}}{\text{s}^2}$, die Anwesenheit des Anregungssignals erkannt werden. Überschreitet die Standardabweichung in vier aufeinanderfolgenden Blöcken den Schwellwert, werden die Daten als valide betrachtet und für die Kalibrierung herangezogen. Die Abbildung 6.4 zeigt die blockweise berechneten Standardabweichungen für Messungen in der PTB sowie die Validität der Daten. Zwischen den einzelnen farblich markierten Blöcken befinden sich wenige nicht valide Blöcke. Die Anregungsfrequenz eines Blockes ist der DAU-Software zu diesem Zeitpunkt noch unbekannt, da die Anregungsfrequenz alleine vom ACS gesteuert wird. Je nach Kalibriersystem beginnt die eigentliche Verarbeitung der Daten erst nach einer bestimmten Zeitverzögerung, da die Regler des Kalibriersystems eine Einschwingzeit von einigen Sekunden benötigen, um eine stabile Amplitude zu liefern.

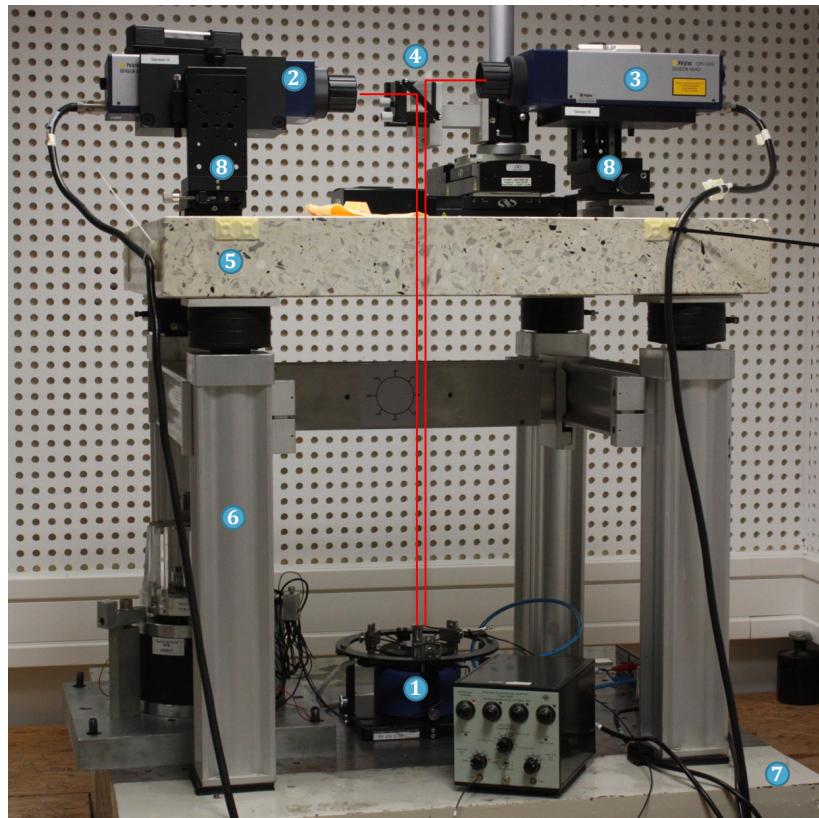


Abbildung 6.5: **Mechanischer Aufbau der HF-B-NME der PTB.** ① Schwingerreger Spektra SE-09, ②, ③ LDVs, ④ Strahlteiler zur Umlenkung der beiden LDV-Laserstrahlen auf das DUT, ⑤ Marmorplatte, ⑥ Gestell mit Schwingungsdämpfern, ⑦ Betonfundament, ⑧ LDV-Justagemechanik.

6.3 Hochfrequenz-Beschleunigungs-Normalmesseinrichtung der PTB

Die Abbildung 6.5 zeigt den mechanischen Aufbau der Hochfrequenz-Beschleunigungs-Normalmesseinrichtung (HF-B-NME) der PTB, die sich in einem klimatisierten (Lufttemperatur $(23,5 \pm 2,0)^\circ\text{C}$) Labor befindet. Auf einem sehr schweren Betonfundament ⑦ steht der luftgelagerte elektrodynamische Schwingerreger vom Typ Spektra SE-09 ①. Über eine Kombination aus einem Umlenkprisma und einem Strahlteiler ④ werden die Strahlen der in der Höhe leicht versetzten Laser-Doppler-Vibrometer ②&③ auf das Messobjekt gelenkt. Die Vibrometer können in ihrer Position und Höhe justiert werden, um die Strahlposition auf dem Messobjekt zu verändern. Die LDVs sind auf einem Marmorblock ⑤ montiert, der von pneumatischen Schwingungsisolatoren ⑥ getragen wird. Durch die Verwendung von zwei LDVs kann durch Mittelwertbildung der Einfluss von Kippschwingungen kompensiert werden [16]. Bei den Kalibrierungsmessungen wurden beide LDV-Strahlen im Abstand von etwa 2 mm auf die auf dem MEMS-Sensor angebrachte Reflektorfolie fokussiert. Für die Bestimmung der Übertragungsfunktion wird der komplexe Mittelwert der Beschleunigung beider LDVs verwendet.

Der elektrische Aufbau der HF-B-NME ist in Abbildung 6.6 für ein LDV gezeigt. Um eine primäre Messung zu realisieren, wird das Heterodynignal des LDV im Kalibriersystem demoduliert und

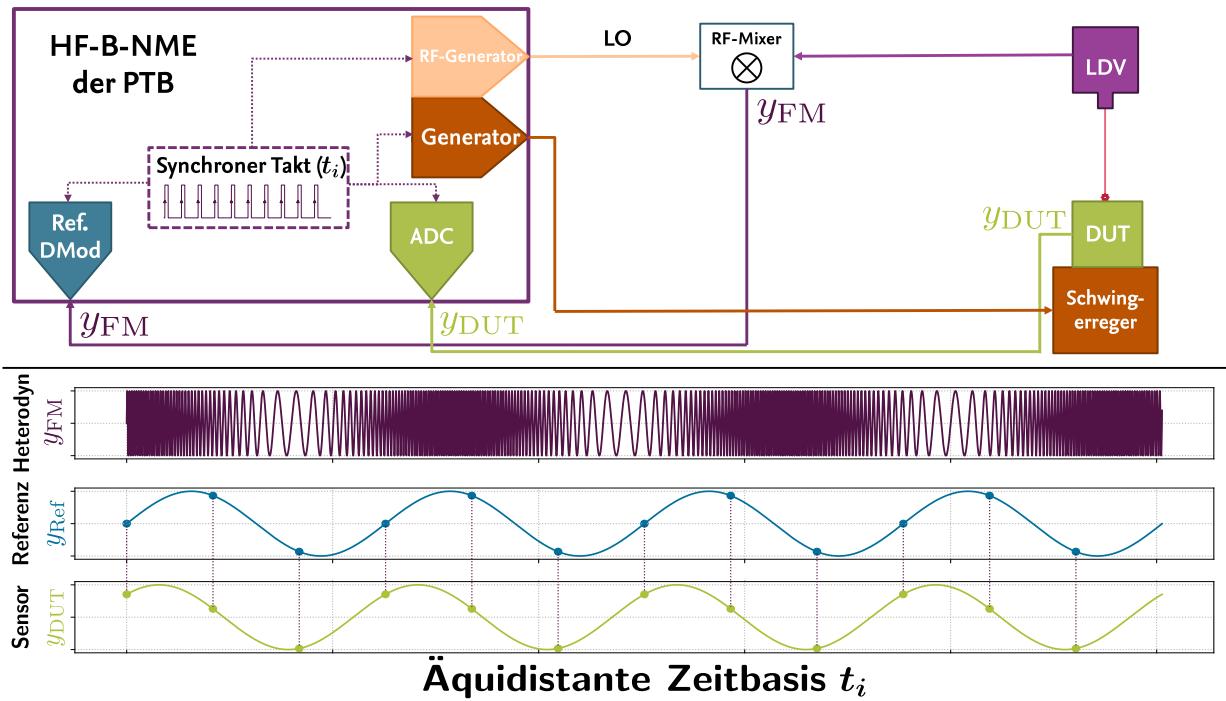


Abbildung 6.6: **Verschaltung und Signale der HF-B-NME der PTB für analoge Sensoren.** Nur eines der beiden LDVs ist dargestellt. Zur einfacheren Auswertung wird das Heterodynignal (violette) des LDV mit einem 39 MHz LO-Träger von 40 MHz auf 1 MHz heruntergemischt, und anschließend synchron zur analogen Datenerfassung (grün) demoduliert. Die Signalerzeugung (orange) ist ebenfalls dazu synchronisiert.

ausgewertet, statt den analogen Spannungsausgang des Vibrometers zu verwenden, wie es bei einer sekundären Kalibrierung üblich wäre. Die Kombination aus *Polytec OFV-505 LDV*-Kopf und *Polytec OFV-3001 Laser-Vibrometer-Controller* liefert ein phasenmoduliertes Heterodynignal mit einer Trägerfrequenz von 40 MHz (siehe Gleichung 2.21). Zur einfacheren Demodulation wird das Heterodynignal mit Hilfe eines 39 MHz Local Oscillator (LO) auf 1 MHz heruntergemischt. Das violette Signal in Abbildung 6.5 zeigt schematisch das heruntergemischte phasenmodulierte Heterodynignal.

Das heruntergemischte Heterodynignal wird mittels Arcustangens-Demodulation und Phasenentfaltung (engl. *Phase unwrap*) in ein Wegsignal $y_{\text{IF-POS}, f, t-i}$ überführt. Mit der bekannten Anregungsfrequenz f kann nun eine S3PSA durchgeführt werden. Zweimaliges Ableiten im Frequenzraum, also Multiplikation mit $-\omega^2$, siehe Gleichungen 2.11, liefert nun Betrag und Phase der Referenzbeschleunigung [103, 104]. Das analoge DUT-Signal, bzw. das Referenzsignal bei digitalen Sensoren, wird von einer *National Instruments PXI-5922* 24 Bit-Datenerfassungskarte synchron zur LDV-Datenerfassung digitalisiert. Mittels S3PSA werden die Amplituden und Initialphasen des Analogsignals bestimmt. Für die synchron approximierten Blöcke werden die komplexen Übertragungskoeffizienten und Anregungsamplituden sowie deren Standardabweichungen berechnet. Die Positionsdaten sowie die analogen Rohdaten werden vom Kalibriersystem nicht gespeichert und stehen für eine Auswertung nicht zur Verfügung. Die Berechnung der Übertragungsfunktion der digitalen Sensoren erfolgt nach der in Abschnitt 5.1 beschriebenen Methodik.



Abbildung 6.7: **Beschleunigungskalibriersystem Spektra CS-18 am CEM mit DAU und Sensor.** Das System besteht aus folgenden Komponenten: ① MEMS-Sensor auf Aluminiumblock, ② luftgelagerter Schwingerreger Spektra SE-09, ③ LDV-Kopf Polytec CLV-2534 mit ④ Controller, ⑤ Schwingungsisolation, ⑥ Granitbasisplatte, ⑦ Leistungsverstärker, ⑧ Steuerungseinheit und ⑨ DAU für digitale Sensoren.

6.4 Hochfrequenzmesseinrichtung des CEM

Das spanische nationale Metrologieinstitut **Centro Español de Metroología (CEM)** hat im Rahmen des **Met4FoF**-Projekts Kalibermessungen mit demselben MPU-9250-Sensor durchgeführt, um diese dann mit den Messungen der **PTB** zu vergleichen und so das Kalibrierverfahren zu validieren.

6.4.1 Systemaufbau

Am **CEM** wird ein kommerziell erhältliches Schwingungskalibriersystem der Firma **SPEKTRA Schwingungstechnik und Akustik GmbH Dresden** vom Typ **CS-18** verwendet. Der Aufbau dieses Systems ist zusammen mit einer **DAU** ⑨ in Abb. 6.7 gezeigt. Als Schwingerreger ② wird wie an der **PTB** ein **Spektra SE-09** mit Luflagerung verwendet. Ebenso wird ein **Spektra BAA 500** Leistungsverstärker ⑦ eingesetzt. Die Montage des Sensors ① erfolgt wieder über das $1/4"$ -28-UNF-Gewinde von Schwingarmateur und Aluminiumblock.

Als **LDV** kommt ein **Polytec CLV-2534** ③ mit entsprechendem **Controller** ④ zur primären Messung der Beschleunigung zum Einsatz. Die Heterodyn demodulation findet dabei im **Vibrometer-Controller** statt. Das demodulierte Signal wird digital über eine **S/PDIF**-Schnittstelle an das Kalibriersystem übertragen. Diese digitale Audio-Schnittstelle wird synchron zur Analog-Digital-Wandlung des **DUT**-Eingangs erfasst, um die Phasenverzögerung zu messen. Das **LDV** ist entsprechend der ISO 16063 – 11 [3] durch einen luftgelagerten Schwingungsisolator ⑤ von dem schweren Erregerfundament ⑥

getrennt. Die Steuerung des Kalibriersystems erfolgt mit der Software des *Spektra CS-18* Systems. Die Abbildung 6.8 zeigt schematisch den elektrischen Aufbau des *CEM*-Kalibriersystems.

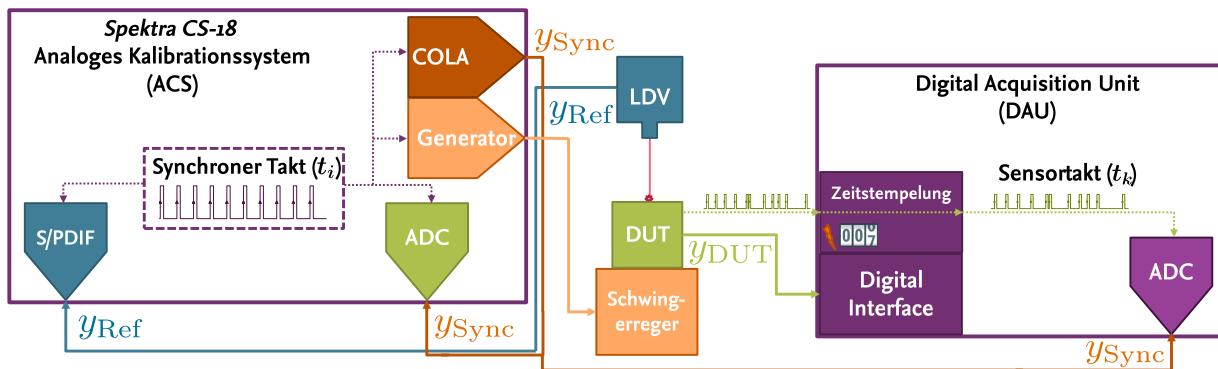


Abbildung 6.8: **Beschleunigungskalibriersystem am CEM.** Das **COLA**-Signal des *Spektra CS-18* Systems wird als analoge Phasenreferenz verwendet. Die Demodulation des Heterodynsignals findet im **LDV**-Controller statt, die demodulierten Daten werden über die optische digitale **S/PDIF**-Schnittstelle übertragen, das Abtasten der analogen Werte erfolgt hierzu synchron.

6.4.2 Kalibrierung

Entsprechend der an der **PTB** durchgeführten Messungen wurden die Übertragungskoeffizienten für die in Tabelle 6.2 gelisteten Anregungssamplituden und Frequenzen durchgeführt. Vor jeder Wiederholungsmessung wurden vom *Spektra CS-18* System Testmessungen bei 80 Hz und 250 Hz durchgeführt; diese Daten wurden nicht verwendet. Zwischen den einzelnen Frequenzpunkten wurde eine Pause von etwa 10 s eingefügt, um die einzelnen Frequenzpunkte durch die blockweise berechnete Standardabweichung unterscheiden zu können (siehe Abb. 6.4). Das CS-18-System lässt nur eine Konfiguration der Messdauer in ganzen Perioden zu. Dadurch ergibt sich für tiefere Frequenzen eine entsprechend längere Messdauer. Die Periodenzahl wurde so gewählt, dass für eine Messung mit 250 Hz Messdaten für 10 s vorhanden sind. Für 10 Hz führt dies nun zu einer Messdauer von 250 s. Die 9 Messungen am *CEM* wurden aus organisatorischen Gründen an verschiedenen Tagen durchgeführt. Dabei wurde der Sensor nicht von der Schwingarmatur demontiert. Ebenso wie im **PTB**-Labor wurde die Lufttemperatur mittels Klimaanlage auf $(23,5 \pm 2,0)^\circ\text{C}$ geregelt. Abbildung 6.9 zeigt die blockweise berechnete Standardabweichung eines Durchlaufs der *CEM*-Messungen.

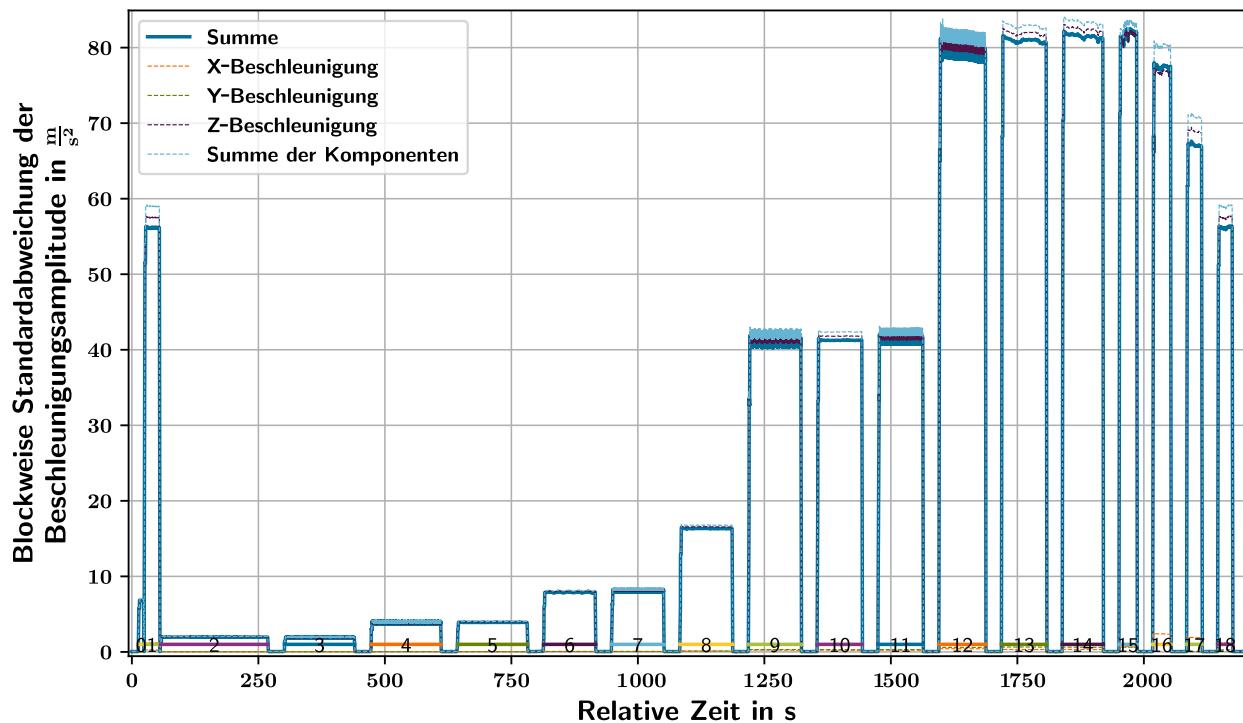


Abbildung 6.9: **Blockweise berechnete Standardabweichung zur Detektion des Anregungssignals in den CEM-Messungen.** Die Standardabweichung der pythagoreischen Summe der Komponenten $\sigma(\sqrt{x^2 + y^2 + z^2})$ ist in **Blau** dargestellt. Liegt diese für 5 Blöcke mit je 100 Messwerten über dem Schwellwert von 1 m/s^2 , wird eine valide Anregung detektiert. Dargestellt sind valide Blöcke durch verschiedene farbige Linien auf der Höhe des Schwellwerts, zusammen mit der Nummer des Experiments. Durch die mögliche Phasenverschiebung kann die **hellblaue** Summe der Standardabweichungen der Einzelkomponenten $\sigma(x) + \sigma(y) + \sigma(z)$ größer sein als die Standardabweichung ihrer pythagoreischen Summe. Die ersten beiden Messungen dienen der Systemidentifikation durch das CS-18-System. Dieses System verwendet eine Vorverzerrung des Erregersignals, daher treten weniger Querkomponenten als bei der **PTB**-Einrichtung auf.

6.4.3 Einfluss des Montagewinkels

Der luftgelagerte Schwingerreger bewegt sich nicht auf einer ideal geraden Bahn entlang der Z-Achse. Je nach Frequenz gibt es auch Beschleunigungskomponenten in der XY-Ebene. Außerdem rotiert die Armatur aufgrund der von den Kabeln erzeugten Querkräfte periodisch mit der Anregungsfrequenz, sodass sich vorwiegend eine Schraubbewegung um die Z-Achse ergibt. Diese Kipp-, Taumel und Drehbewegungen sind eine der Hauptunsicherheitsquellen bei der dynamischen Kalibrierung von Beschleunigungssensoren [16]. Dies ist auch in den tabellierten CMC-Unsicherheitsbudgets von **PTB** und **CEM** (Tab. 10.1 bis 10.4) erkennbar.

Durch den nicht rotationssymmetrischen Aufbau des Sensor-BoB auf dem Aluminiumblock und die vielen Anschlussleitungen ist es möglich, dass unerwünschte Bewegungen bei dem in dieser Arbeit verwendeten Sensor verstärkt auftreten.

Zur Untersuchung dieser und anderer Einflüsse wurde der **MPU-9250**-Sensor auf der **HF-B-NME** der **PTB** in verschiedenen Winkellagen γ , bezogen auf die Z-Achse, montiert und es wurden Kalibrierungsmessungen durchgeführt. Um verschiedene Montagewinkel zu ermöglichen, wurden unterschiedlich dicke Unterlegscheiben unter den Sensormontageblock gelegt und der Block anschließend mit $2 \text{ N} \cdot \text{m}$ Drehmoment angezogen. Abbildung 6.10 zeigt die verschiedenen Montagewinkel. Da durch die Unterlegscheiben die mechanische Stabilität der Sensorbefestigung verringert ist, wurde die Anregungsamplitude auf 100 m/s^2 reduziert (siehe Tabelle 6.2). Zur Ermittlung von statistischen Parametern wurden die Messungen für jede Winkelposition fünfmal wiederholt, wobei der Sensormontageblock nicht demontiert wurde.

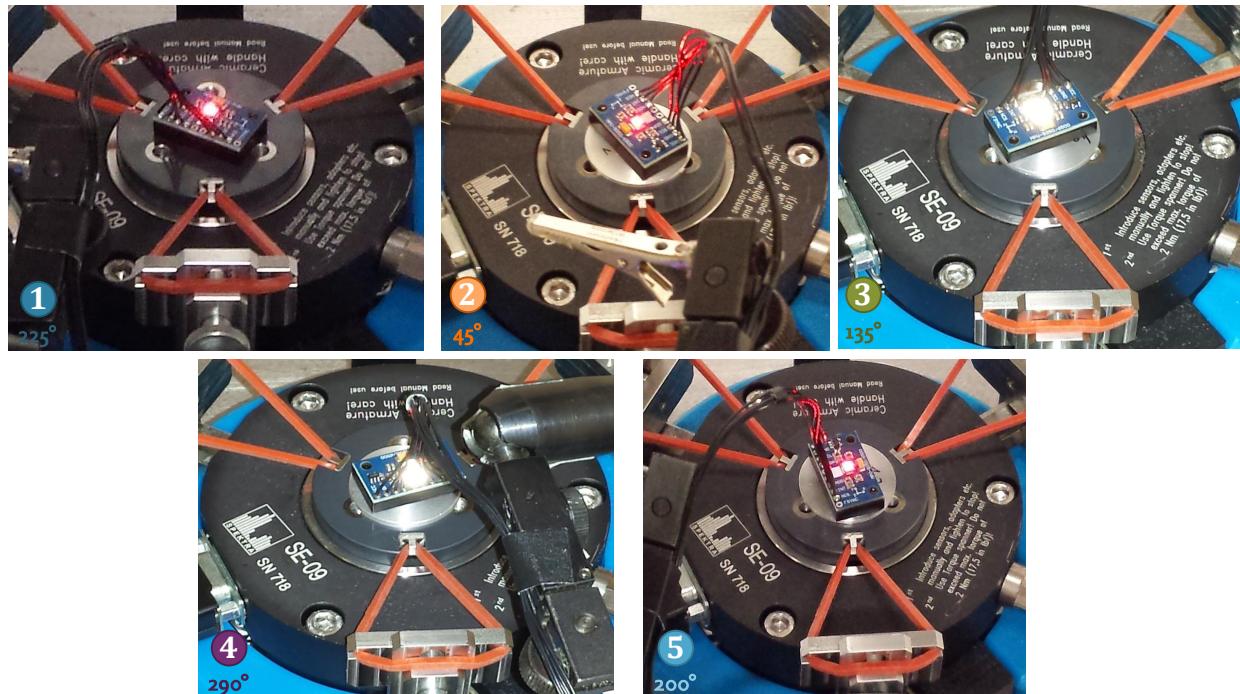


Abbildung 6.10: **Verschiedene Rotationswinkel γ des Sensors bezogen auf das Labor-Koordinatensystem des Schwingerregers.**

Die Abbildung 6.11a zeigt die für die verschiedenen Montagewinkel bestimmten Übertragungsfunktionen des MPU-9250-Sensors. Die Amplitudenabschwächung durch das aktivierte **DLPF** sowie die nahezu konstante Gruppenlaufzeit sind gut zu erkennen. Die Phasenverzögerung nimmt linear mit der Anregungsfrequenz zu, was in der logarithmischen X-Achse der Grafiken zu einem scheinbar exponentiellen Abfall der Phase führt. Die Einflüsse durch die verschiedenen Montagewinkel sind durch den relativ großen Magnituden- und Phasengang des Sensors nicht erkennbar.

Deshalb werden in Abbildung 6.11b die Differenzen der einzelnen Übertragungskoeffizienten zu ihrem nach statistischer Unsicherheit gewichteten Mittelwert aller fünf Winkelpositionen gezeigt. Die Magnituden der Übertragungsfunktion zeigen jetzt eine signifikante Beeinflussung durch den Montagewinkel. So ist beispielsweise die Magnitude der Messungen bei 45° für alle Frequenzen etwa 0,1 % kleiner als bei 200° . Diese Abweichungen sind, verglichen mit dem grau dargestellten 0,2 % breiten **CMC**-Unsicherheitsintervall der **HF-B-NME** der **PTB**, als relativ groß anzusehen. Bei 100 Hz scheint es zu einer mechanischen Resonanz oder elektromagnetischen Einkopplung der ersten Oberwelle des 50 Hz Stromnetzes zu kommen, weshalb hier die Abweichungen noch einmal deutlich größer sind. Die nach Gleichung 5.5 berechneten rein statistischen Unsicherheiten ergeben keine gegenseitige Überdeckung der Magnituden für die unterschiedlichen Montagewinkel. Denn ein systematischer Einfluss durch die Kipp- und Taumelbewegungen (*Rocking Motions*) wird in diesem Ansatz nicht berücksichtigt. Bei der Phase der Übertragungsfunktion ergibt sich ein anderes Bild. Abgesehen von dem durch eine mechanische Resonanz oder elektromagnetische Einkopplung betroffenen Messpunkt bei 100 Hz zeigen die Werte eine geringe Streuung im Bereich von $0,05^\circ$. Die statistischen Unsicherheiten nach Gleichung 5.6 ergeben eine gegenseitige Überdeckung der Messungen für unterschiedliche Montagewinkel. Die Varianz der Messungen ist außerdem klein gegen das $0,2^\circ$ breite **CMC**-Unsicherheitsintervall. Eine Verkippung des Sensors auf dem Schwingerreger gegen die Messachse der **LDVs** führt zu einer dem Kosinus des Kippwinkels entsprechenden Amplitudenabnahme. Die ermittelte Phasenverzögerung ändert sich durch diese Verkippung nur geringfügig.

Da die **MPU-9250** einen dreiachsigen Beschleunigungssensor enthält, kann aus den approximierten Beschleunigungsamplituden und Phasen unter Vernachlässigung von Rotationen die Translations-Trajektorie berechnet werden. Für die sinusförmige Beschleunigung liefert eine doppelte Integration, also Division durch $-\omega^2$, direkt die analytische Lösung für die Schwingwegkoordinate; siehe Gleichung 2.9. Es ergibt sich folglich für die Wegkoordinate im Sensor-Koordinatensystem

$$\begin{bmatrix} X_{\text{Sensor}}(t) \\ Y_{\text{Sensor}}(t) \\ Z_{\text{Sensor}}(t) \end{bmatrix} = \frac{1}{-\omega^2} \begin{bmatrix} \hat{a}_x \sin(\omega t + \varphi_x) \\ \hat{a}_y \sin(\omega t + \varphi_y) \\ \hat{a}_z \sin(\omega t + \varphi_z) \end{bmatrix}. \quad (6.1)$$

Da der Rotationswinkel γ für die Drehung um die Z-Achse bekannt ist, können die Trajektorien aus dem Sensor-Koordinatensystem durch eine einfache Rotation in das Labor-Koordinatensystem überführt werden.

$$\begin{bmatrix} X_{\text{Labor}}(t) \\ Y_{\text{Labor}}(t) \\ Z_{\text{Labor}}(t) \end{bmatrix} = R_z(\gamma) \begin{bmatrix} X_{\text{Sensor}}(t) \\ Y_{\text{Sensor}}(t) \\ Z_{\text{Sensor}}(t) \end{bmatrix} = \frac{1}{-\omega^2} \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{a}_x \sin(\omega t + \varphi_x) \\ \hat{a}_y \sin(\omega t + \varphi_y) \\ \hat{a}_z \sin(\omega t + \varphi_z) \end{bmatrix} \quad (6.2)$$

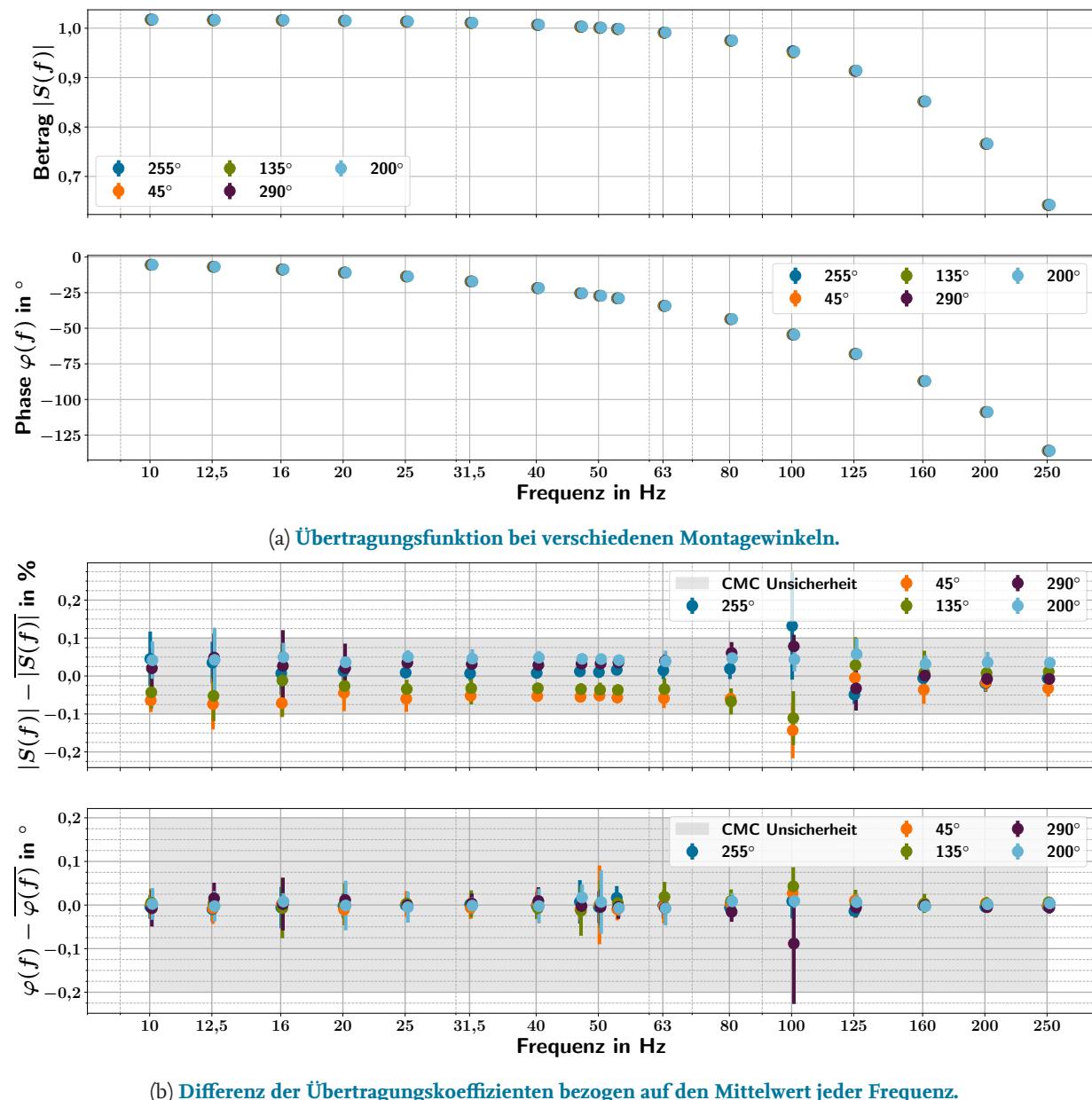


Abbildung 6.11: Übertragungsfunktion und Differenz zum Mittelwert des Z-Beschleunigungssensors der IMU MPU-9250 auf der HF-B-NME der PTB bei verschiedenen Montagewinkeln. Die obere Abbildung (a) zeigt die gemessenen Übertragungsfunktionen. Die untere Abbildung (b) zeigt die Abweichung der Übertragungskoeffizienten der jeweiligen Frequenzen bezogen auf ihren Mittelwert. Die gezeigten rein statistischen Unsicherheiten sind 2σ von je 5 Wiederholungsmessungen pro Montagewinkel. Die Phasenverzögerungen stimmen innerhalb der statistischen Unsicherheit überein, während die Magnituden nicht übereinstimmen. Durch Berücksichtigung von Montageeinflüssen und anderen statischen Einflüssen ergibt sich Übereinstimmung in den CMC-UNSicherheiten auch für die Magnitude. Die Abweichungen bei 50 Hz sowie 100 Hz könnten durch Einkopplungen aus dem Stromnetz verursacht werden.

Die Abbildung 6.12 zeigt auf der linken Seite jeweils die für verschiedene Montagewinkel ermittelten Trajektorien im Sensor-Koordinatensystem und auf der rechten Seite die Trajektorien nach Drehung um γ im Labor-Koordinatensystem. Für die niederfrequente Bewegung mit $f = 10\text{ Hz}$ ergibt sich ein Amplitudenverhältnis zwischen angeregter und nicht angeregter Achse von $\frac{4\text{ }\mu\text{m}}{400\text{ }\mu\text{m}} \approx 1\%$. Die Trajektorien der 10-Hz-Anregung (siehe 6.12a) liegen im Sensor-Koordinatensystem unabhängig vom Montagewinkel nahezu übereinander. Alle Beschleunigungssignale haben die gleiche Phase, was zu einer wenig geöffneten Trajektorie führt. Die Überlagerung im Sensor-Koordinatensystem kann im Wesentlichen durch die folgenden Effekte beschrieben werden: Verkippung des Sensors auf dem Montageblock; die Lötstellen für die Signalleitungen befinden sich nur entlang einer Seite des **BoB** (entlang der Y-Achse), daher ist die Unterseite des **BoB** nicht perfekt eben. Der verwendete Klebstoff gleicht diese Unterschiede zwar etwas aus, dennoch ist eine leichte Verkippung des Sensors gegen die Montagefläche vorhanden. Diese Kippung führt zu einer scheinbaren Querbeschleunigung im Sensor-Koordinatensystem.

Zum anderen können die relativ steifen Anschlusskabel zu einer Querbeschleunigung führen. Die Anschlussleitungen schwingen ebensowie die Armatur und der Sensor, sodass Kräfte und Biegemomente über die Leitungen auf den Sensor übertragen werden, welche zu einer Beschleunigung und auch zu einer Verschiebung entlang der X-Achse führen.

Die Verbindungsleitungen führen zu einer Kippung um die Z-Achse, und durch das Erdschwerefeld führt diese Kippung zu einer Beschleunigungskomponente entlang der Kippachse. Da Kippfrequenz und Bewegungsfrequenz starr gekoppelt sind, kann diese Signalkomponente nicht von der Querbeschleunigung unterschieden werden.

Bei 63 Hz Anregungsfrequenz (siehe 6.12b) überlagern sich die Trajektorien von 0° , 200° und 290° im Sensor-Koordinatensystem nur noch näherungsweise. Das Amplitudenverhältnis von Z-Position zur Abweichung vom Mittelpunkt in der XY-Ebene erhöht sich auf 2 %. Durch den kürzeren Schwingweg ($400\text{ }\mu\text{m}$ statt $800\text{ }\mu\text{m}$) und die deutlich höhere Beschleunigung (44 m/s^2 statt 2 m/s^2) ändern sich die Störeinflüsse deutlich. Durch die höheren Frequenzen und damit kleineren Wellenlängen kann es zur Ausbildung lokaler Resonanzen im System aus Schwingerreger, Montageblock, Sensor-**BoB** und Anschlussleitungen kommen. Das Auftreten dieser Resonanzeneinflüsse ist schwer vorherzusagen, da schon kleine Änderungen der Parameter, wie z.B. Steifigkeit und Lage der Anschlussleitungsbefestigung, zu deutlichen Änderungen der Resonanzen führen können. Bei der Anregung mit 200 Hz (siehe 6.12c) zeigt sich eine Überlagerung im Labor-Koordinatensystem für die Montagewinkel 0° , 135° und 290° , was auf das Auftreten von Resonanzen im Schwingerreger hindeutet. Zur genaueren Untersuchung dieser Effekte sind daher mehrachsige oder scannende **LDV**-Messungen notwendig, um die tatsächlich auftretende Beschleunigung von den Einflüssen durch Auslenkung und Kippung zu trennen. Die hier durchgeführte Untersuchung für unterschiedliche Montagewinkel verdeutlicht den Einfluss der Kipp- und Taumelbewegung auf die Kalibrierung von Betrag und Phase (siehe Eintrag $u(\hat{u}_T)$ bzw. $u(\varphi_{u,T})$ in den **CMC**-Tabellen 5.1).

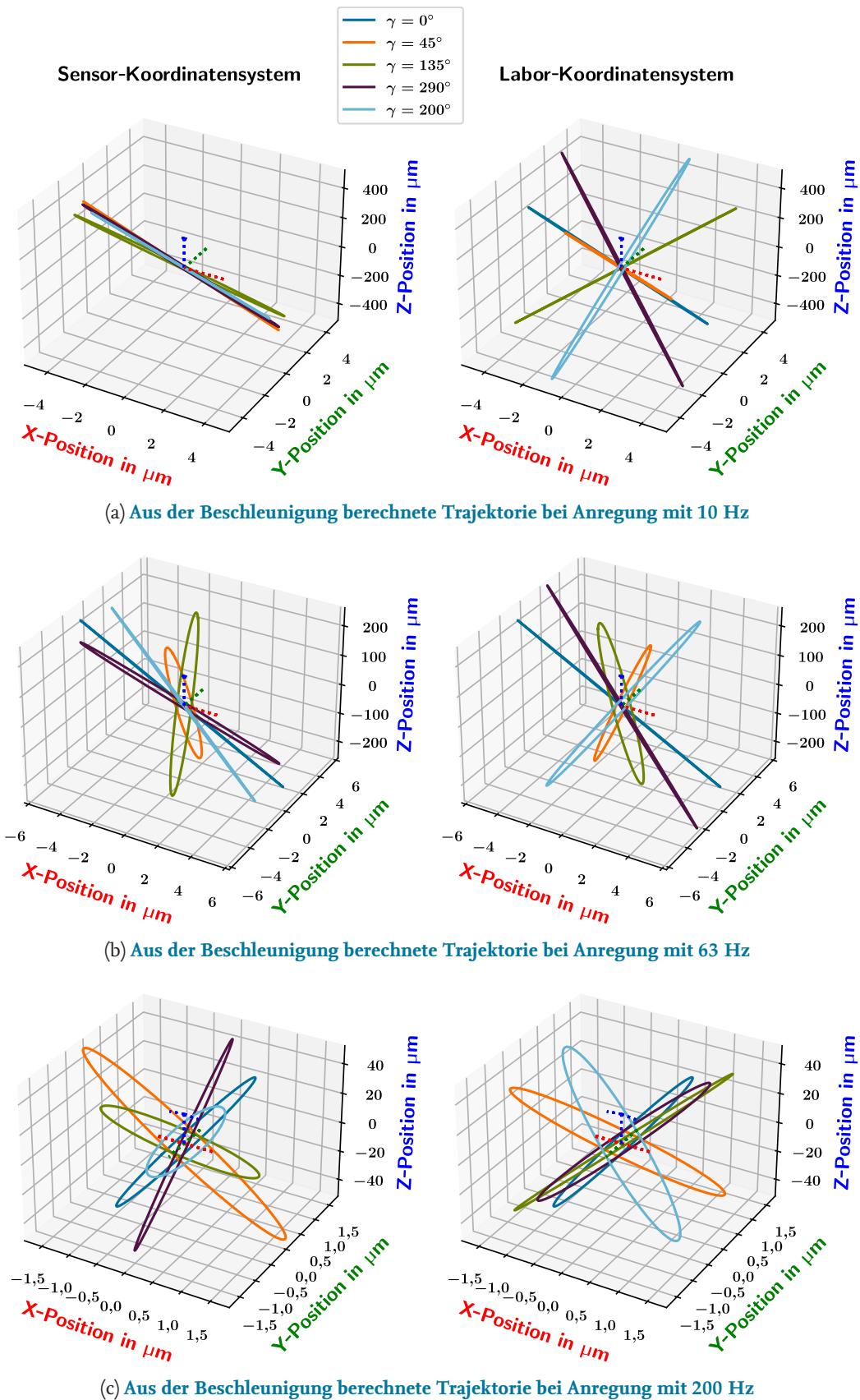


Abbildung 6.12: **Aus den approximierten Beschleunigungen berechnete Trajektorien im Sensor- und Labor-Koordinatensystem für verschiedene Anregungsfrequenzen und Montagewinkel.** Beobachtungen zeigen, dass bei 10 Hz die Trajektorien im Sensor-Koordinatensystem übereinander liegen, was auf eine Missweisung der Sensorachsen hinweist; bei 200 Hz treten Resonanzen im Schwingerreger auf.

6.5 Vergleich der Messungen von PTB und CEM

Die hier beschriebenen Messungen sowie ihr Vergleich wurden unter dem Titel *Comparison in dynamic primary calibration of digital-output accelerometer between CEM and PTB* im Journal *Metrologia* veröffentlicht [85].

Die Abbildung 6.13a zeigt Betrag $|S(f)|$ und Phase $\varphi(f)$ der ermittelten Übertragungsfunktionen sowie ihre statistischen Unsicherheiten. Die kleinen rein statistischen Unsicherheiten sind im Vergleich zur starken Magnitudenabnahme durch das aktivierte **DLPF** mit 184 Hz Grenzfrequenz bzw. der stetig ansteigenden Phasenverzögerung durch die Gruppenlaufzeit des **DLPF** nicht erkennbar. Die Messwerte liegen in dieser Grafik übereinander.

Einen genaueren Überblick über die Abweichungen der Messungen untereinander ermöglicht die Betrachtung der Abweichungen vom gewichteten Mittel der Übertragungskoeffizienten pro Frequenzpunkt.

Die Abbildung 6.13b zeigt oben die relative Abweichung der Beträge der Übertragungskoeffizienten vom gewichteten Mittel $|S(f)| - \overline{|S(f)|}$. Dabei wurden die ersten **PTB**-Messungen nicht für die Mittelwertbildung verwendet, sondern nur die **PTB**-Messungen mit verschiedenen Montagewinkeln. Die Abweichung wird, wie in der Beschleunigungsmesstechnik für die Magnituden üblich, relativ in Prozent der Magnitude angegeben.

Die unten gezeigte Abweichung der Phasenverzögerung vom Mittelwert $\varphi(f) - \overline{\varphi(f)}$ ist wie üblich in Winkelgrad ($^{\circ}$) angegeben. Das gewichtete Mittel $\overline{\varphi(f)}$ als bester Schätzwert für den Übertragungskoeffizienten wird analog zu dem Vorgehen bei Ringvergleichen (*Key Comparisons*) [13] wie folgt berechnet.

$$\overline{X(f)} = \sum \frac{X_i(f)}{u_i^2(f)} \cdot \left(\sum \frac{1}{u_i^2(f)} \right)^{-1} \quad (6.3)$$

Hierbei ist $X_i(f)$ der ermittelte Wert des i -ten Teilnehmers mit der korrespondierenden Unsicherheit $u_i^2(f)$. Die Varianz des Mittelwerts kann entsprechend mit

$$u_{\text{KC}}^2(f) = \left(\sum \frac{1}{u_i^2(f)} \right)^{-1} \quad (6.4)$$

berechnet werden. In der Veröffentlichung [85] finden sich die tabellarisierten Werte für den Mittelwert der Übertragungsfunktion. Die Messungen am **CEM** und die danach an der **PTB** durchgeführten Messungen mit verschiedenen Montagewinkeln stimmen für die Magnitude, innerhalb ihrer Typ-A-UNSicherheiten überein.

Die Phasen stimmen für alle Messungen unter 100 Hz innerhalb der Typ-A-UNSicherheiten überein. Für die Frequenzen um 50 Hz zeigen die **CEM**-Messungen eine relativ große Typ-A-UNSicherheit, welche durch Einkopplung von Netzbrummen in die analoge Phasenreferenz der **DAU** verursacht worden sein kann. Die bei der ersten Messung an der **PTB** (in den Abbildungen 6.13a bis 6.14 in **Blau**) bestimmten Magnituden liegen konstant um 0,2 % über den Werten der beiden anderen Messungen. Die Ursache dieser Magnitudenabweichung konnte nicht abschließend geklärt werden.

Bei der **IMU** vom Typ **MPU-9250** handelt es sich um einen günstigen *Consumer-Grade*-Sensor (Preis etwa 2,50 € pro Stück inklusive **BoB**). Daher ist die zeitliche Stabilität der Sensitivität, im Gegensatz zu **IMUs** für die Luft und Raumfahrt, nicht spezifiziert. Zwischen den verschiedenen Messungen liegen größere Zeiträume von einigen Wochen bis Monaten, **PTB** 26.8.2020, **CEM** 07.09.2020, **PTB** mit

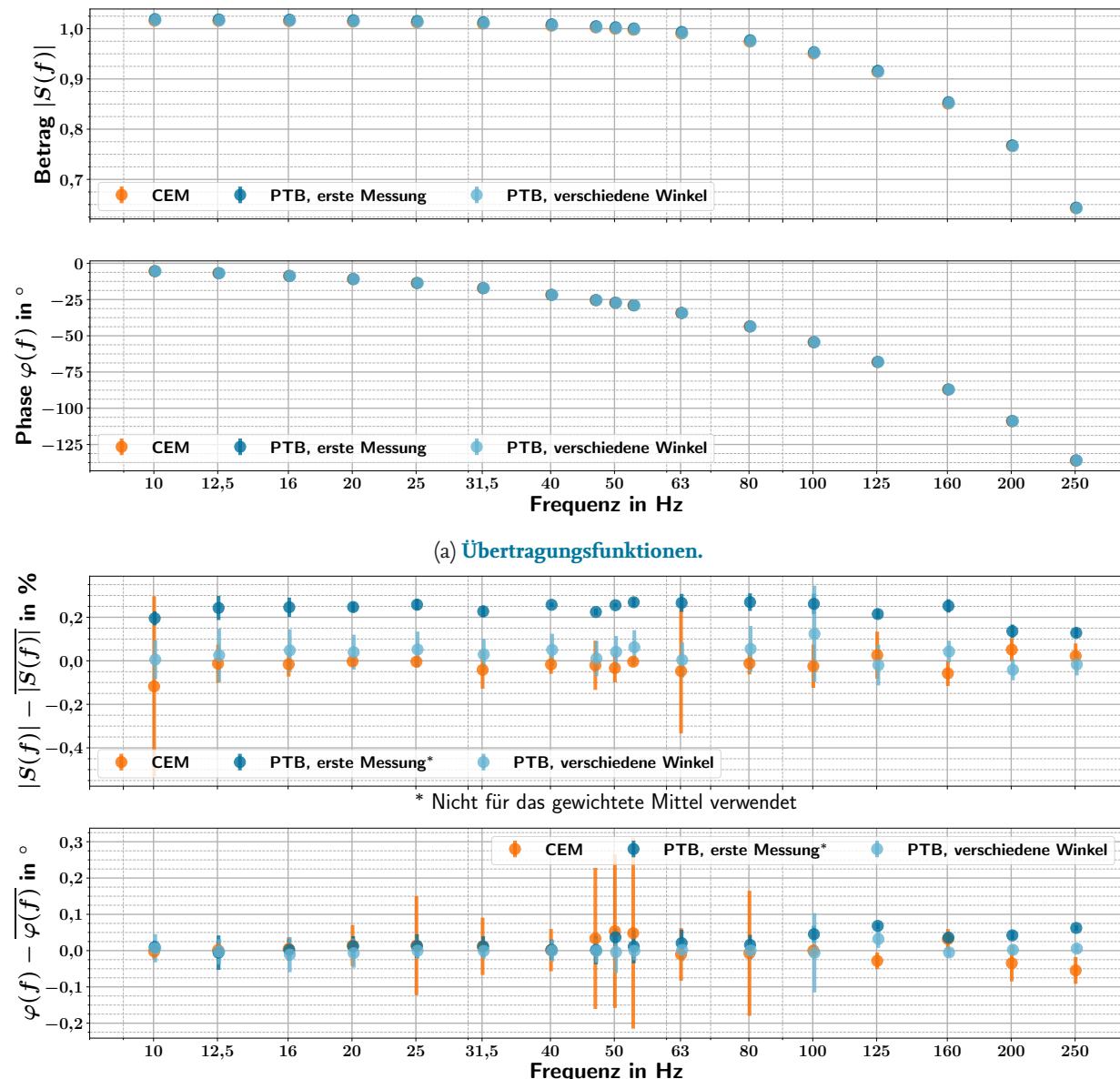


Abbildung 6.13: **Gemessene Übertragungsfunktionen und Differenz vom gewichteten Mittel des MPU-9250 im Vergleich.** Die obere Abbildung (a) zeigt die an der PTB und am CEM gemessenen Übertragungsfunktionen des Z-Beschleunigungssensors der IMU MPU-9250. Die Amplitudendämpfung durch das DLPF mit einer nominellen Grenzfrequenz von 184 Hz ist erkennbar, ebenso wie die durch die Gruppenlaufzeit des DLPF verursachte Phasenverzögerung. Die Abweichungen der einzelnen Messungen sind in (a) nicht erkennbar, da sie sehr klein gegen die Dämpfung durch das DLPF sind. Um die Abweichungen zu visualisieren, wird daher in Abbildung (b) die Differenz der Übertragungskoeffizienten zum gewichteten Mittel bei der jeweiligen Frequenz gezeigt. Die Magnitude der ersten PTB-Messung sowie die Phasen für Frequenzen >100 Hz zeigen keine Übereinstimmung innerhalb der rein statistischen 2σ -Umgebung der Typ-A-Unsicherheit.

verschiedenen Winkeln 06.08.2021. In der Literatur [59] wurde die statische Sensitivität ($f = 0$) von MEMS-Accelerometern über einen langen Zeitraum beobachtet. Dabei zeigten sich zum Teil plötzliche Änderungen der Sensitivität in der Größenordnung von 0,3 %. Ursächlich hierfür sind vermutlich Setzungseffekte im Sensorelement, oder gerade bei Kunststoffgehäusen, auch im Gehäuse. Zusätzlich könnte der Sensor beim Transport beschädigt worden sein. Schon das Fallenlassen des Sensors aus geringer Höhe (50 cm) auf eine harte Oberfläche kann Beschleunigungen von vielen km/s^2 erzeugen, welche zur Zerstörung des Sensors oder einer Änderung (vermutlich Reduktion durch Abbrechen von Kondensatorfingern) der Sensitivität führen könnte [68]. Der MPU-9250 ist mit einer *Shock Reliability* von 98,1 km/s^2 (10000 g_n) spezifiziert. Daher ist eine Beschädigung durch Stoß unwahrscheinlich, aber nicht unmöglich. Durch die Absolutzeitstemplung und Verwendung dieser Zeitstempel für die Sinus-Approximation werden die Effekte des Abtastphasenrauschens des Sensors nahezu vollständig kompensiert und führen daher nicht zu Amplitudenabweichungen. Wie die Messungen mit verschiedenen Montagewinkeln gezeigt haben, sind systematische Effekte durch die Sensormontage die Hauptunsicherheitsquelle bei Beschleunigungskalibrierungen. In den CMCs für die Kalibrierungen werden diese Einflüsse entsprechend gewichtet, siehe Tabellen 10.1 bis 10.4. Die CMCs sind für die besten verfügbaren Referenzauflnehmer ausgelegt und beinhalten daher nur kleine sensorspezifische Komponenten.

Werden, wie für akkreditierte Kalibrierungen vorgeschrieben, die CMC-Unsicherheiten statt der Typ-A-Unsicherheiten verwendet, überdecken sich sowohl die Magnituden als auch die Phasen für alle Frequenzen gegenseitig; dies ist in Abbildung 6.14 gezeigt. Da in diesen Unsicherheiten Abschätzungen für die Montageeinflüsse enthalten sind, bestätigt dies die Validität des hier vorgestellten Kalibrierverfahrens.

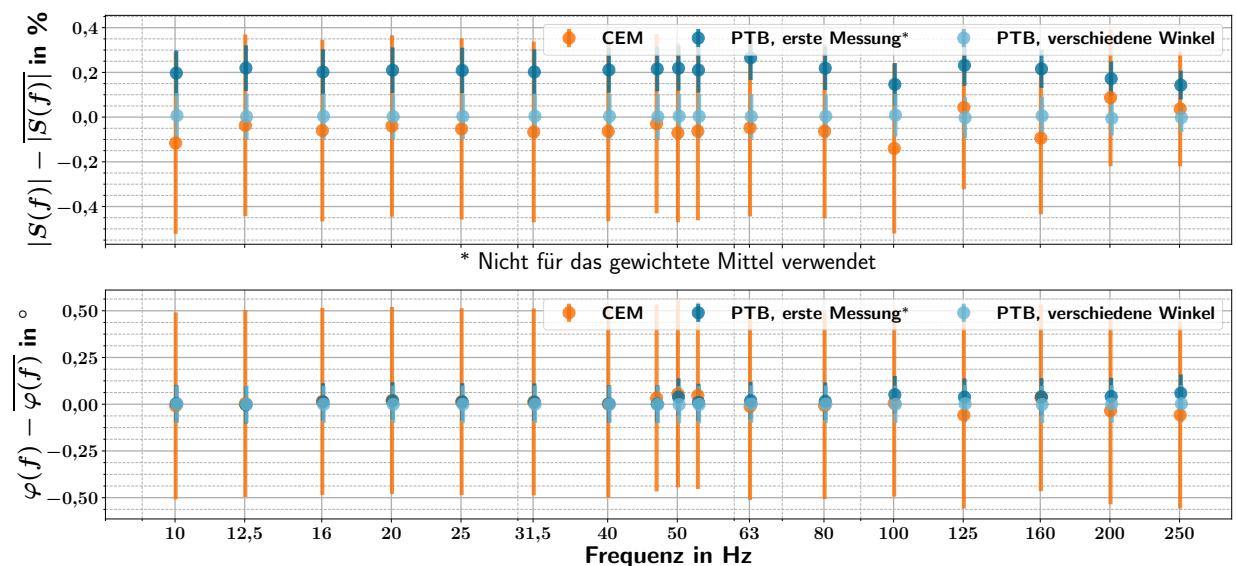


Abbildung 6.14: **Differenz vom gewichteten Mittel der Übertragungsfunktionen des MPU-9250 mit Angabe der Unsicherheit wie in den CMCs für analoge Sensoren.** Durch die Berücksichtigung der Montageeinflüsse in den CMCs sind nun auch alle Magnitudenwerte in Übereinstimmung.

7 Ausblick

Für die genauere Untersuchung der Eigenschaften der, im Vergleich zu den seit mindestens 70 Jahren etablierten piezoelektrischen Sensoren, relativ neuen **MEMS**-Sensoren werden in Zukunft viele Kalibrierungen nötig sein, insbesondere um die Stabilität der Sensoren gegenüber Umwelt-einflüssen und Alterung zu untersuchen. Für die Hersteller von **MEMS**-Sensoren ist außerdem von Interesse, inwieweit es möglich ist, von den Eigenschaften einiger Sensoren einer Produktionschar-ge auf die Eigenschaften aller Sensoren rückzuschließen [73, 72]. Auch für diese Zielsetzung ist die Kalibrierung von vielen Sensoren nötig.

In naher Zukunft werden basierend auf den hier vorgestellten Ansätzen an der **PTB** und vielen ande-rem Instituten komplexere dynamische Kalibrierverfahren für **MEMS**-Sensoren entwickelt werden.

7.1 Drei-Achsen-Multisensor-Kalibriersystem mit Absolutzeitbasis

Auf Basis der in dieser Arbeit vorgestellten **DAU** soll nach erfolgreichem Passieren des zweiten Reviewvorgangs in naher Zukunft innerhalb eines **EU**-Forschungsprojekts (**SRT**-do2 „Large-scale virtual calibration of MEMS sensors for sensor networks“) im **EMP**-Programm mit verschiedenen Industriepartnern (u. A. *STMicroelectronics* und *Infineon Technologies*) sowie **NMIs** (u. A. **INRIM** und **RISE**) ein dreiachsiges Multisensor-Beschleunigungskalibriersystem an der **PTB** entwickelt werden. Mit diesem Multisensor-Kalibriersystem sollen anschließend viele Beschleunigungssensoren (≈ 100) simultan mehrachsig kalibriert werden, um zu untersuchen, wie die Kalibrierung einer Stichprobe von **MEMS**-Beschleunigungssensoren genutzt werden kann, um Informationen und Übertragungsfunktionen für eine ganze Charge von (Tausenden) Sensoren abzuleiten.

Die Abbildung 7.1 zeigt schematisch den möglichen Aufbau eines solchen Systems. Die **DAU** hat vier **DUT**-Anschlüsse (siehe Abb. 3.3) und kann bis zu vier unterschiedliche oder gleiche Sensoren simultan auslesen. Durch die Hardwarezeitstempelung können auch bei gleichzeitigem Auslösen der **DRIs** der Sensoren stets die korrekten Absolutzeitstempel vergeben werden. Die **DAU** kommu-niziert über **Ethernet** mit dem Host-**PC**. Daher können mehrere **DAUs** Daten an denselben Host senden. Falls die Leistung eines PCs nicht ausreicht, um die Daten mehrerer **DAUs** zu erfassen, kön-nen auch mehrere **PCs** verwendet werden. Für die Anregung und Erfassung der Referenzbeschleuni-gung wird im geplanten Kalibriersystem die Dreikomponenten-Schwingbeschleunigungseinrich-tung der **PTB** verwendet. Diese Einrichtung bietet eine Schwingarmatur mit etwa 50 cm Durchmes-ser, auf welcher eine große Zahl von Sensoren montiert werden kann. Das Datenerfassungssystem dieser Einrichtung wurde im Laufe dieser Arbeit mit einem **GNSS**-Empfänger (National Instru-ments PXI-6682) ausgerüstet und verfügt daher über eine Absolutzeitbasis [28]. Die **PPS**-Signale dieser Zeitbasis werden in Zukunft zusammen mit der 10-MHz-Normalfrequenz-Distribution der **PTB** als Zeitquelle für die **DAUs** verwendet. Mittels **Network Time Protocol (NTP)** werden die ein-zelnen **DAUs** über **Ethernet** mit einer Uhrzeit (Unsicherheit < 10 ms) versorgt, sodass die Zeit des

nächsten PPS-Signals eindeutig bekannt ist. Wie in Abbildung 3.15 erkennbar ist, wird die Zeitabweichung der einzelnen DAUs untereinander bei Verwendung einer gemeinsamen Zeitbasis auf die Quantisierungsbreite reduziert. Die daraus resultierenden Amplituden- und Phasenabweichungen sind in jedem Fall vernachlässigbar klein. Die Bestimmung der Übertragungsfunktion vereinfacht sich durch die Absolutzeitbasis erheblich, da keine analoge Phasenreferenz mehr verwendet werden muss. Die Abbildung 7.2 zeigt schematisch die simulierten Beschleunigungsmesswerte einer Achse für vier Sensoren. Die Abtastraten bzw. Zeitpunkte der Sensoren sind weder untereinander noch mit der Referenz synchronisiert (siehe Tabelle 7.1). Die Messdaten der Referenz werden zu den absoluten Zeitpunkten

$$t_{\text{REF}0}, t_{\text{REF}1}, t_{\text{REF}2}, \dots, t_{\text{REF}_{n_{\text{REF}}}}$$

erfasst. Die Messdaten der Sensoren (DUTs) entsprechen den Zeitpunkten

$$\begin{aligned} t_{\text{DUT}0_0}, t_{\text{DUT}0_1}, t_{\text{DUT}0_2}, \dots, t_{\text{DUT}0_{n_{\text{DUT}0}}} \\ t_{\text{DUT}1_0}, t_{\text{DUT}1_1}, t_{\text{DUT}1_2}, \dots, t_{\text{DUT}1_{n_{\text{DUT}1}}} \end{aligned}$$

Für die Auswertung der Kalibriermessungen werden von allen im Beobachtungszeitraum liegenden Messwerten die Absolutzeiten durch Subtrahieren des Referenz-Startzeitpunkts $t_{\text{REF}0}$ in relative Zeiten konvertiert.

$$\Delta t_{\text{REF}_i} = t_{\text{REF}_i} - t_{\text{REF}0} \quad (7.1)$$

$$\Delta t_{\text{DUT}0_i} = t_{\text{DUT}0_i} - t_{\text{REF}0} \quad (7.2)$$

$$\Delta t_{\text{DUT}1_i} = t_{\text{DUT}1_i} - t_{\text{REF}0} \quad (7.3)$$

Wie in Abbildung 7.2 dargestellt, führen die unterschiedlichen Abtastraten der Sensoren und der Referenz zu einer unterschiedlichen Anzahlen an Messwerten $n_{\text{REF}}, n_{\text{DUT}0}, n_{\text{DUT}1}$ für die verschiedenen Sensoren. Mit der S3PSA werden für Zeitreihen die Amplituden \hat{a}_{REF} bzw. \hat{a}_{DUT} und die Phasenverzögerungen φ_{REF} bzw. φ_{DUT} an dem Zeitpunkt $t_{\text{REF}0}$ approximiert. Für die Magnitude der Übertragungsfunktion ergibt sich entsprechend der Kalibrierung mit der analogen Phasenreferenz (siehe Gleichung 2.13)

$$|S(f)| = \frac{\hat{a}_{\text{DUT}}(f)}{\hat{a}_{\text{REF}}(f)}. \quad (7.4)$$

Die Phasenverzögerung ergibt sich zu

$$\Delta\varphi(f) = \varphi_{\text{REF}}(f) - \varphi_{\text{DUT}}(f). \quad (7.5)$$

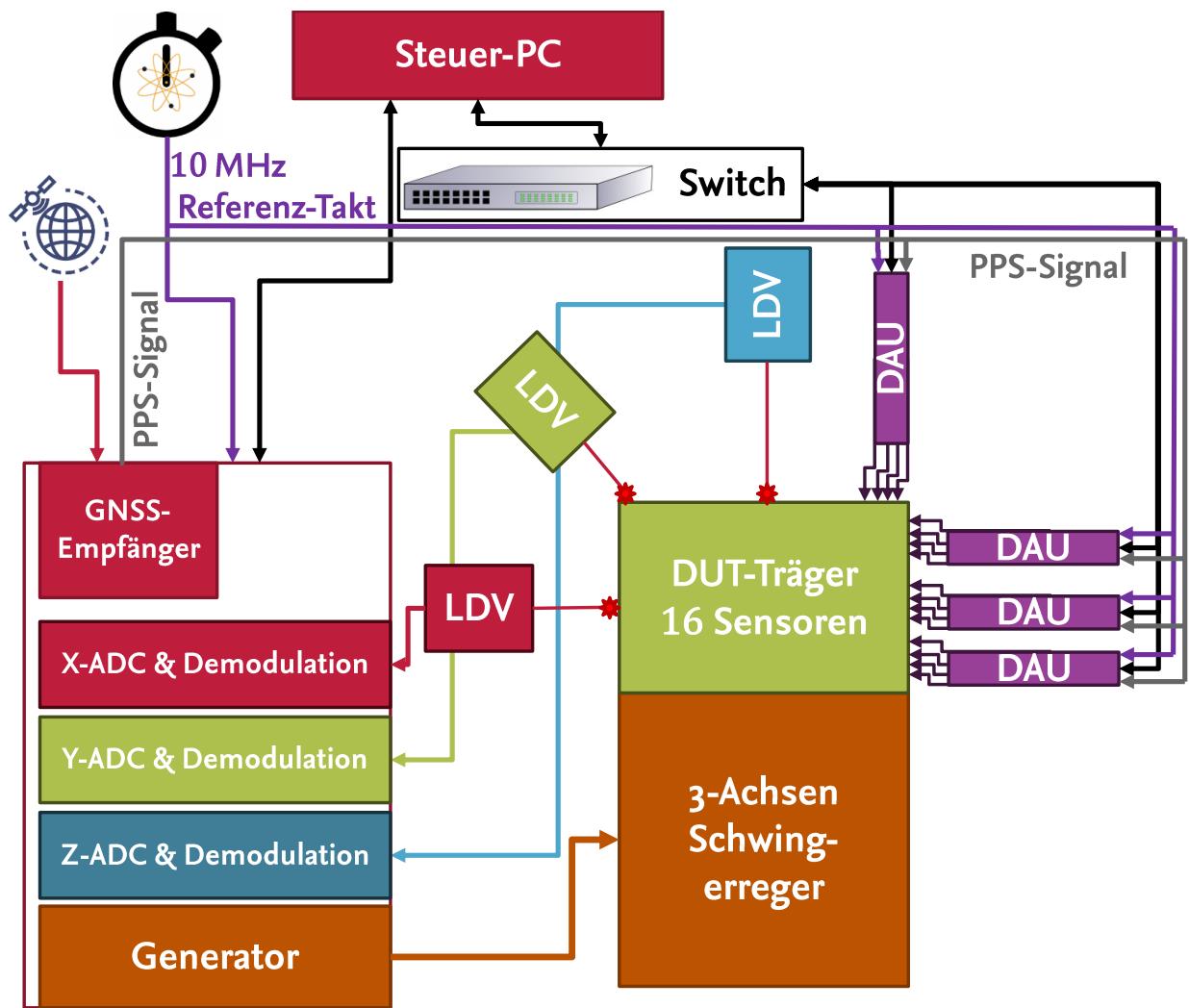


Abbildung 7.1: **Konzept für ein 3-achsiges Multisensor-Kalibriersystem mit Absolutzeitbasis**. Sowohl das um eine GNSS-Zeitbasis erweiterte ACS als auch die vier DAUs erfassen absolutzeitgestempelte Messwerte. Die Anzahl der DUTs ist nur durch den verfügbaren Platz auf der Schwingarmatur begrenzt. Veröffentlicht in [84].

Im Gegensatz zu der in dieser Arbeit verwendeten Methodik (nach Gleichung 5.3) sind keine weiteren analogen Hilfssignale nötig. Folglich entfallen sowohl die Unsicherheitsbeiträge des Synchronisierungssignals als auch die des **DAU-ADC**. In den Grafiken 5.4a und 5.4b ist erkennbar, dass die durch die Analogphasenreferenz verursachte Unsicherheit einen nicht unerheblichen Teil der gesamten statistischen Phasenunsicherheit ausmacht. Für den in **Dunkelgrün** dargestellten DAU-Phasenunsicherheitsbeitrag, $\sigma^2(\varphi_{\text{DUT}}(f) - \varphi_{\text{SyncDAU}}(f))$ ist der **hellgrün** dargestellte Einfluss des Synchronisierungssignals zum Teil dominierend. Bei den **PTB**-Messungen dominiert der Einfluss des Synchronisierungssignals bei tiefen Frequenzen; bei den **CEM**-Messungen nimmt dieser Einfluss mit höheren Frequenzen zu und wird für Frequenzen über 160 Hz dominierend. Für die **orange** dargestellten Phasenverzögerungen im **ACS** sind die Einzelkomponenten nicht genau bekannt. Das Synchronisierungssignal wird vermutlich auch in diesem Systemteil einen signifikanten Unsicherheitsbeitrag verursachen.

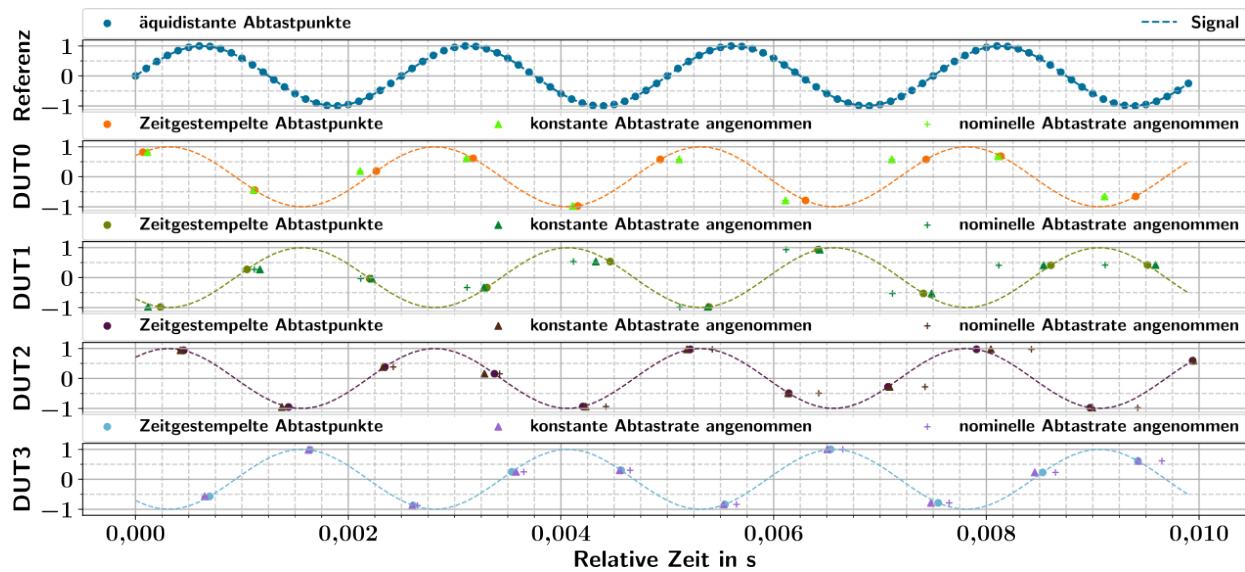


Abbildung 7.2: **Simulation der abgetasteten Daten im Multisensor-Kalibriersystem.** Aus den absolutzeitstempelten Messdaten wird die relative Zeit auf den ersten Messwert der Referenz bezogen berechnet. Hiermit wird die S3PSA durchgeführt. Veröffentlicht in [84].

Tabelle 7.1: **Parameter der simulierten Sensordaten in einem Multisensor-Kalibriersystem.**

Sensor	Mittlere Abtastfrequenz f_s in Hz	Abtastphase φ_{Abtast} in °	Phase φ in °	Abtast-Jitter-Standardabweichung σ_{Jitter} in % von $1/f_s$
Referenz	10000	0	0	0
DUT0	1000	20	45	15
DUT1	950	20	225	11
DUT2	1050	80	45	8
DUT3	1025	120	225	3

7.2 Linearität

Die in den **MEMS**-Elementen realisierten Federelemente sind nur in erster Näherung und für kleine Kräfte bzw. Auslenkungen linear. Das Gleiche gilt auch für die kapazitiven Sensorelemente (vgl. Gleichung 2.46). Daher ist die Amplitudenlinearität der **MEMS**-Sensoren wesentlich schlechter als die der klassischen piezoelektrischen Aufnehmer, bei denen die Auslenkung des Sensorelements stets sehr klein gegen die Dicke ist. Für den in dieser Arbeit verwendeten *MPU-9250* sind z. B. 0,5 % statische Nichtlinearität spezifiziert [46]. Für den *BMA280* werden 0,5 % Nichtlinearität für den kleinsten Messbereich von $\pm 20 \text{ m/s}^2$ spezifiziert [12]; für die größeren Messbereiche liegen keine Spezifikationen vor. Die Linearität wird für **MEMS**-Sensoren als Abweichung von der besten Approximation einer geraden Kennlinie angegeben (*Best Fit Straight Line*). Die nichtlineare Sensitivität des Sensors führt bei dynamischen Signalen zum Entstehen von harmonischen Spektralanteilen inklusive der nullten Harmonischen, also einem Gleichanteil. Die Abbildung 7.3 aus [49] zeigt dies schematisch.

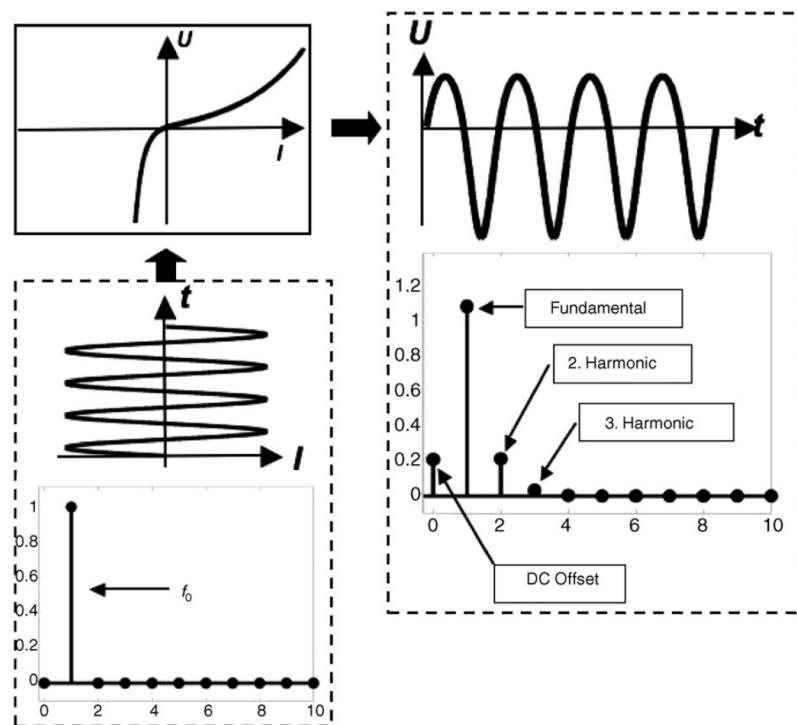


Abbildung 7.3: **Schematische Darstellung der Erzeugung der Harmonischen an einer Nichtlinearität.** Ein perfektes Sinussignal wird durch eine nichtlineare Kennlinie verzerrt; dadurch entstehen die Obertöne. Entnommen aus [49].

Die Erzeugung eines Gleichanteils aus einem Schwingungssignal wird als *Vibration Rectification* bezeichnet [71]. Dieser Effekt verursacht bei Initial-Navigationsanwendungen aufgrund der doppelten Integration der Beschleunigung erhebliche Abweichungen von berechneter Geschwindigkeit und Position [42]. Die Abbildung 7.4 zeigt farblich kodiert den Gleichanteil entlang der Z-Achse bei verschiedenen Anregungsfrequenzen und Amplituden. Der durch die Nichtlinearität erzeugte Gleichanteil ist von der Anregungsamplitude abhängig und nimmt mit ansteigender Anregungsamplitude kon-

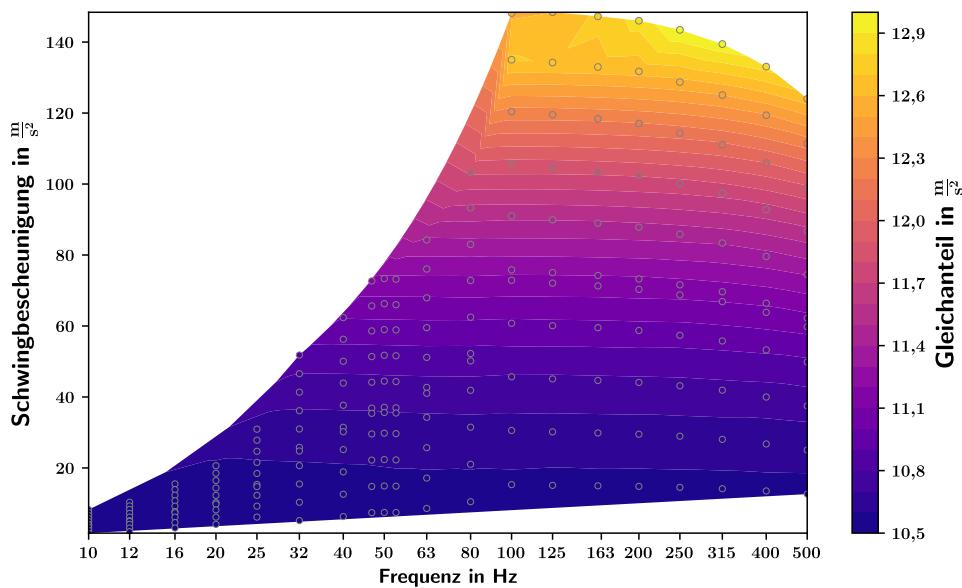


Abbildung 7.4: **Gleichanteil aus Nichtlinearität.** An der nichtlinearen Sensitivitätsfunktion der Z-Achse eines BMA280-Sensors werden Harmonische des monofrequenten sinusförmigen Anregungssignals erzeugt. Der Gleichanteil ist die nullte Harmonische. Entlang der Z-Achse wirkt auch die Erdbeschleunigung von $9,81 \text{ m/s}^2$. Der Nullpunktsfehler (engl. *Offset*) und eine Sensitivität von etwas mehr als 1,0 führen zusammen zu einer statischen Beschleunigung von etwa $10,5 \text{ m/s}^2$. An den umkreisten Messpunkten wurde der Gleichanteil bei Sinusanregung erfasst, die farbigen Flächen sind interpoliert.

tinuierlich zu. Für höhere Frequenzen scheint der Anteil der nullten Harmonischen etwas abzunehmen. **MEMS**-Beschleunigungssensoren zeigen auch Hystereseeffekte [58], welche neben der Nichtlinearität ursächlich für die Entstehung der nullten Harmonischen sein könnten. Im Gegensatz zu höheren Harmonischen können mechanische Ursachen für die nullte Harmonische, also eine scheinbar gleichförmige Beschleunigung, ausgeschlossen werden, da die Erdbeschleunigung konstant und der Schwingerreger im Mittel unbewegt ist. Die statische Kalibrierung der Aufnehmer-nichtlinearität ist durch Rotation im Erdschwerefeld für Beschleunigungen $|a| < g$ relativ leicht möglich [17]. Für höhere Beschleunigungen müssten Zentrifugen zur Erzeugung der statischen Beschleunigung verwendet werden [67]. In dem in diesem Abschnitt vorgeschlagenen Multisensor-Kalibriersystem sind für alle Sensoren und Referenzen die Messdaten im Zeitbereich vorhanden. Durch eine Multisinus-Approximation können die Amplituden und Phasen der Harmonischen bestimmt werden. Für die Kalibrierung von **ADCs** existieren Verfahren, die aus den Harmonischen die nichtlineare Sensitivitätsfunktion berechnen [19, 74, 75]. Auch eine Berechnung der Hysterese ist aus den Harmonischen möglich [63]. Diese Verfahren lassen sich analog auf die Kalibrierung von Beschleunigungssensoren anwenden. Da Teile der Nichtlinearität von der Sensorgeometrie abhängig sind, könnte es möglich sein, diese für alle Sensoren eines Typs mit einem einheitlichen Modell zu kompensieren. Durch die gleichzeitige Kalibrierung von vielen Sensoren werden die nötigen Daten zur generalisierten Modellbildung erhalten. Auch die Untersuchung des Alterungsverhaltens von vielen verschiedenen Sensoren wird durch die Zeitersparnis der Multisensor-Kalibrierung praktikabel.

7.3 Die DAU als Smart-Sensor

Im **Met4FoF**-Forschungsprojekt, innerhalb dessen diese Arbeit statt fand, wurden metrologische Methoden für industrielle Anwendungen entwickelt. Eine dieser Anwendungen ist die auf maschinellem Lernen (**ML**) basierende automatische Lebenszeitabschätzung von Verschleißteilen (engl. *Lifetime Estimation*). Als Test für diese *Lifetime Estimation* existiert am **ZeMA** ein Prüfstand für **Kugelgewindetriebe**. Der **Kugelgewindetrieb** kann in dem Prüfstand durch einen Pneumatikzylinder belastet werden, um Bewegung unter Last zu simulieren. Ein Arbeitszyklus aus belasteter Fahrt gegen den Pneumatikzylinder und anschließendem unbelasteten Zurückfahren in die Ausgangsposition dauert etwa 2,8 s. Diese Arbeitszyklen werden bis zur Zerstörung des **KGT** durch Verschleiß nach etwa 630000 Zyklen (20 d) wiederholt. Während dieser Zeit werden Daten (Motorstrom, Vibration, Lautstärke, Temperaturen) aufgezeichnet. Mittels **ML** wird aus diesen Daten dann ein Modell für die *Lifetime Estimation* des **KGT** abgeleitet. Im Rahmen des **Met4FoF**-Projekts wurde zusätzlich zu den bereits verbauten analogen Sensoren eine **DAU** mit 3 verschiedenen digitalen Sensoren (**IMU** **MPU-9250**, Triaxialbeschleunigungssensor **BMA280** und ein Druck-/Temperatursensor **MS5837**) mit einer Metallspange auf dem **KGT** installiert.

Die Abbildung 7.5 zeigt die **DAU**, die Sensoren sowie einen der analogen Sensoren (**Kistler 8712A5M1** Beschleunigungsaufnehmer) des **ZeMA**-Prüfstands.

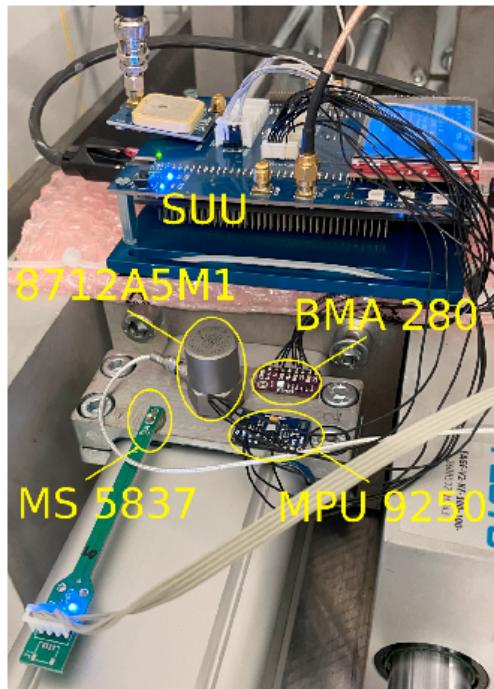
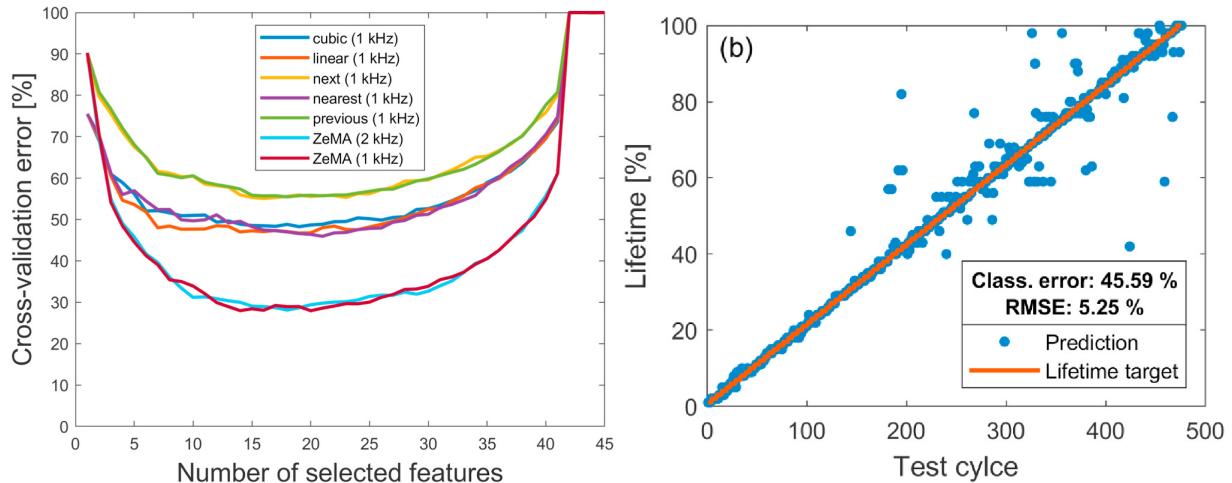


Abbildung 7.5: **DAU mit verschiedenen Sensoren im Prüfstand am ZeMA**. An die **DAU** sind eine **IMU** vom Typ **MPU-9250** (Beschleunigung (X,Y,Z), Winkelgeschwindigkeit (X,Y,Z), magnetische Flussdichte (X,Y,Z) und Temperatur) der Firma **TDK InvenSense**, ein triaxialer Beschleunigungssensor vom Typ **BMA280** der Firma **Bosch** sowie ein Druck- und Temperatursensor vom Typ **MS5837** der Firma **TE-Connectivity** angeschlossen. Diese Sensoren sind zusammen mit einem Piezobeschleunigungssensor **Kistler 8712A5M1** über eine Metallspange am **Kugelgewindetrieb** befestigt [28].

Die beiden im Prüfstand verbauten Beschleunigungssensoren haben verschiedene Orientierungen und Übertragungsfunktionen. Durch die Kalibrierung dieser Sensoren auf der Dreikomponenten-Schwingungsbeschleunigungsmesseinrichtung der PTB konnte sowohl die Orientierung der Sensoren untereinander als auch ihr Frequenzgang bestimmt werden. Damit der Frequenzgang der Sensoren nicht Teil des ML-Models zur Lebenszeitberechnung wird, wurde der Frequenzgang der Sensoren durch ein IIR-Filter kompensiert [28], und die unterschiedlichen Ausrichtungen wurden durch anwenden einer Rotationsmatrix kompensiert. Die am ZeMA entwickelte ML-Toolbox evaluiert automatisch verschiedene Verfahren zur Merkmalsextraktion aus den Messdaten. Die angewendeten Verfahren sind Adaptive Linear Approximation (ALA), Best Daubechies Wavelets (BDW), Best Fourier Coefficients (BFC), Principal Component Analysis (PCA) und Statistical Moments (SM). Dabei werden diejenigen Verfahren bevorzugt, die mit den wenigsten Merkmalen den niedrigsten Kreuz-Validierungsfehler (engl. Cross-Validation Error, CVE) liefert. Zur Bestimmung des CVE werden die aufgezeichneten Zyklen zufällig in eine Trainings-Gruppe (90%) und eine Validierungs-Gruppe (10%) aufgeteilt. Mit den Zyklen aus der Trainings-Gruppe und ihrer bekannten Lebensdauer wird das ML-System trainiert, um dann von den nicht für das Training verwendeten Validierungs-Zyklen die Lebensdauervorhersage zu testen. Die Lebensdauervorhersage erfolgt dabei nicht kontinuierlich, sondern diskret in Kategorien von 1% der Lebensdauer. Folglich führt schon eine Abweichung von 1% zu einem Fehler für die CVE-Berechnung. Die besten Ergebnisse mit den wenigsten Merkmalen wurden mittels BFC erreicht. Sowohl die BFC-Merkmalsextraktion als auch die IIR-Filterung benötigen äquidistant abgetastete Werte, welche die Sensoren aufgrund ihrer instabilen Zeitbasis nicht liefern. Daher wurden verschiedene, einfache zeitliche Interpolationsmethoden am ZeMA erprobt. Die Abbildung 7.6a zeigt den CVE gegen die Zahl der Merkmale für die analogen Daten des ZeMA-Systems (in Hellblau und Rot) und zusätzliche für verschiedene Interpolationsmethoden der DAU-Daten. Werden die DAU-Daten nicht zeitlich interpoliert, sondern einfach als äquidistant abgetastet angenommen, ist keine Lebenszeitberechnung mit diesen Daten möglich. Lineare, nächste oder kubische Interpolationen zeigen eine ähnlich gute Vorhersagequalität. Einfaches Auswählen des vorherigen oder nächsten Messwerts liefert signifikant schlechtere Ergebnisse. Die Abbildung 7.6b zeigt die Vorhersageergebnisse in 1%-Kategorien auf Basis der kubisch interpolierten DAU-Daten im Vergleich zur tatsächlichen Lebensdauer des KGTs. Durch das ML-System werden 46% der Daten in eine falsche Kategorie eingeordnet; der RMS-Abstand der berechneten Lebensdauer von der tatsächlichen beträgt für die DAU-Daten 5,25 %. Die falsch klassifizierten Zyklen werden also überwiegend in benachbarte Kategorien eingeordnet. Mit den analogen Sensoren, die auch im Prüfstand montiert sind, erreicht das ML-System 26,5 % falsche Klassifizierungen bei einem RMS-Abstand von 1,39 %. Die analogen Sensoren erfassen dabei mit wesentlich höherer Datenrate deutlich mehr Messgrößen (3 Phasen Motorstrom mit je 1 MS/s, 3 Beschleunigungssensoren mit je 100 kS/s, sowie Axialkraft, Luftdruck, Motorgeschwindigkeit und aktiver Motorstrom mit je 10 kS/s). Die Materialkosten für die analoge Datenerfassung sind dabei mit etwa 7000 € Größenordnungen größer als die der DAU und der Sensoren von etwa 70 €. Die zeitlich interpolierten DAU-Sensordaten ermöglichen somit eine kostengünstige Lebensdauervorhersage in diesem Prüfstand.



(a) **Kreuzvalidierungsfehler der Lebensdauervorhersage für verschiedene zeitliche Interpolationen.** Die ML-Software benötigt für die FFT äquidistante Messwerte; daher muss für die MEMS-Sensoren im Zeitbereich interpoliert werden.

(b) **Lebensdauervorhersage in 1-%-Klassen mittels BFC aus den kubisch interpolierten DAU-Daten für einen Kugelgewindetrieb.** Die Daten zeigen einen CVE von 46% dabei werden die Zyklen aber in der Regel einer benachbarten Klasse zugeordnet.

Abbildung 7.6: **Ergebnisse der Lebensdauervorhersage eines Kugelgewindetriebs durch Machine Learning.** Diese Berechnungen wurden von den Projektpartnern am ZeMA durchgeführt und die Grafiken in [28] veröffentlicht.

Die im Kapitel 4 erörterten Auswirkungen des Abtastzeitpunktes auf die Sinus-Approximationen und die FFT sind ursächlich für den starken Einfluss der Interpolationsverfahren auf die Vorhersagequalität. Durch den mechanischen Aufbau des KGT und die konstante Verfahrgeschwindigkeit ist davon auszugehen, dass schmalbandige Vibrationssignale in diesem Testsand auftreten. Durch Verschleiß kann es zum Auftreten von Obertönen kommen, welche durch die BFC-Methode detektiert werden. Durch unkorrigierte Abtastzeitabweichungen führt nun die Verbreiterung der Grundtöne im Spektrum dazu, dass die Obertöne nicht mehr detektiert werden können (vlg. Abb. 4.19a). Vermutlich hätte die Verwendung von Multisinus-Approximation zur Bestimmung von Amplitude und Phase der BFCs noch bessere Ergebnisse geliefert.

8 Zusammenfassung

Die fortschreitende Miniaturisierung und Digitalisierung führt in der Beschleunigungsmesstechnik zur Verbreitung von Sensorsystemen mit ausschließlich digitalem Datenausgang, den sogenannten digitalen Sensoren. Für die Messgrößen Beschleunigung und Winkelgeschwindigkeit werden mehr als eine Milliarde **MEMS**-Sensoren pro Jahr gefertigt. Durch den zum Teil günstigen Preis und die sehr kleine Bauform können diese Sensoren auch in anspruchsvoller dynamischen Messaufgaben, beispielsweise der Zustandsüberwachung auf Basis von maschinellem Lernen, eingesetzt werden.

In dieser Arbeit wurde im Rahmen des europäischen Forschungsprojekts *Metrology for the Factory of the Future* (**Met4FoF**) das erste primäre dynamische Kalibrierverfahren für die komplexwertige Übertragungsfunktion von digitalen Beschleunigungssensoren in Anlehnung an ISO 16063 entwickelt und in einem ersten internationalen Vergleich validiert. Zur Ansteuerung der Sensoren wurde eine Mikrocontroller-basierte Datenerfassungseinheit (*Data Acquisition Unit*, DAU) entwickelt. Die DAU erfasst die von den Sensoren asynchron erzeugten Messwerte und zeitstempelt diese gegen eine GNSS-Absolutzeitbasis. Die Messdaten werden in der DAU in SI-Einheiten konvertiert und zusammen mit Metadaten über *Ethernet* an einen Empfangs-PC gesendet. Durch die Verwendung eines Echtzeit-Betriebssystems und Hardware-Zeitstempelung kann jede DAU bis zu vier unterschiedliche Sensoren parallel auslesen. Die DAU wurde auch verwendet, um kalibrierte digitale Sensoren in industrienahen Anwendungen zur Zustandsüberwachung einzusetzen. Die DAU kann synchron zu einem digitalen Sensor auch analoge Spannungen erfassen. Damit kann in den bestehenden primären analogen Beschleunigungskalibriersystemen der nationalen Metrologieinstitute, wie der **PTB** oder dem spanischen **CEM**, ein Phasenbezug hergestellt werden, ohne die bestehenden Kalibriersysteme verändern zu müssen.

Zur Validation des Kalibrierverfahrens wurde dem spanischen Metrologieinstitut CEM eine DAU zur Verfügung gestellt. Der Vergleich der damit bestimmten Übertragungsfunktionen zeigt eine gute Übereinstimmung der Ergebnisse von PTB und CEM und belegt die grundsätzliche Eignung des Verfahrens.

Die asynchrone Messdatenerfassung der MEMS-Sensoren mit Absolutzeitstemplung ist ein wesentlicher Unterschied zu den analogen Sensoren mit getrennter Signalerfassung. Die Qualität der internen Sensorzeitbasis kann mit der DAU bestimmt werden. Daher wurde ein Verfahren zur Berechnung des Abtastphasenrauschens entwickelt. Die Auswirkungen des Abtastphasenrauschens auf die Auswertung dynamischer Messungen im Frequenzraum ohne Absolutzeitstemplung wurde umfassend simuliert und visualisiert. Monte-Carlo-Simulationen belegen, dass unter bestimmten Bedingungen, beispielsweise bei hoher Signalfrequenz und langer Messdauer, das Sensorabtastphasenrauschen zum Teil zu erheblichen Amplitudenabweichungen führt. Bei einigen Sensoren kommt es schon bei Messungen von 30 s Länge bei einer Anregungsfrequenz von 63 Hz zu den beschriebenen Amplitudenabweichungen. Durch die Absolutzeitstemplung und Verwendung der Zeitstempel in der Sinus-Approximation kann dieser Einfluss nahezu vollständig kompensiert werden. Bei der Anwendung kalibrierter digitaler Sensoren mit einer **DAU**, zur Lebenszeitabschätzung

von Verschleißteilen eines Projektpartners konnten deutliche Verbesserungen der Vorhersagegenauigkeit durch Verwendung der Absolutzeitstempel und einfacher Interpolation erreicht werden.

English

Progressive miniaturisation and digitalisation in acceleration measurement is also leading to the spread of sensor systems with exclusively digital data output, so-called digital sensors. More than one billion MEMS sensors are manufactured every year for measuring acceleration and angular velocity. Due to the favourable price and the very small design, these sensors can also be used in more demanding dynamic measurement tasks, such as condition monitoring based on machine learning.

In this work, the first primary dynamic calibration method for the complex-value transfer function of digital acceleration sensors was developed in accordance with ISO 16063 as part of the European research project *Metrology for the Factory of the Future* ([Met4FoF](#)) and validated in an initial international comparison. A microcontroller-based data acquisition unit (DAU) was developed to control the sensors. The DAU records the measured values generated asynchronously by the sensors and timestamps them against a GNSS absolute time base. The measurement data is converted into SI units in the DAU and sent to a receiving PC together with metadata via *Ethernet*. By using a real-time operating system and hardware timestamping, each DAU can read up to four different sensors in parallel. The DAU has also been used to use calibrated digital sensors in industrial applications for condition monitoring. The DAU can also acquire analogue voltages synchronously with a digital sensor. This means that in the existing primary analogue acceleration calibration systems of national metrology institutes, such as the [Physikalisch-Technische Bundesanstalt](#) ([PTB](#)) or the Spanish [Centro Español de Metrología](#) ([CEM](#)), a phase reference can be established without having to change the existing calibration systems.

A DAU was provided to the Spanish metrology institute CEM to validate the calibration procedure. The comparison of the transfer functions determined with it shows a good agreement between the results of PTB and CEM and demonstrates the fundamental suitability of the method.

The asynchronous measurement data acquisition of MEMS sensors with absolute timestamping is a significant difference to analogue sensors with separate signal acquisition. The quality of the internal sensor time base can be determined by the DAU. Therefore, a method for calculating the sampling phase noise was developed. The effects of the sampling phase noise on the evaluation of dynamic measurements in frequency space without absolute timestamping were comprehensively simulated and visualised. Monte Carlo simulations show that under certain conditions, for example for high signal frequencies and long measurement durations, the sensor sampling phase noise sometimes leads to considerable amplitude deviations. With some sensors, the described amplitude deviations occur even with measurement durations of 30 s at an excitation frequency of 63 Hz. This influence can be almost completely compensated for by absolute timestamping and using the timestamps in the sine approximation.

In the application, of a project partner, for estimating the service life of wear parts using calibrated digital sensors and a DAU, significant improvements in prediction accuracy were achieved by using the absolute timestamps and a simple time domain interpolation.

Abkürzungsverzeichnis

- 3PSA** Drei-Parameter-Sinus-Approximation. 12, 13, 17–19, 55, 57–59, 61, 63–66, 68, 69, 73, 79–84, 90, 91, 158–161, 167–169
- 4PSA** Vier-Parameter-Sinus-Approximation. 17, 19, 61, 65, 66, 68, 72, 76, 80, 158, 160
- AV_{CC}** Analogue Voltage at the Common Collector. 55
- DLPF** Digital Low Pass Filter. 107, 109, 118, 123, 124
- NaN** Not a Number. 69
- ADC** Analog-to-Digital Converter. 13, 21–27, 31–33, 35, 47, 55–60, 77, 87, 98, 99, 102, 104, 107, 108, 128, 131, 143
- AM** Amplitudenmodulation. 64, 94
- AOM** Akustooptischer Modulator. 16, 17
- API** Application Programming Interface. 30, 41, 162
- ASCII** American Standard Code for Information Interchange. 35, 39
- AWG** Arbitrary Waveform Generator. 51
- BeiDou** BeiDou GNSS. 7
- BFC** Best Fourier Coefficients. 134
- BoB** Breakout Board. 31, 32, 109–111, 117, 120, 121, 123
- CAN** Controller Area Network. 30–32
- CCIRQ** Timer Capture/Compare Interrupt Request. 34, 47, 48
- CDC** Charge-to-Digital Converter. 27
- CEM** Centro Español de Metrología. 4, 47, 55, 67, 99, 101, 102, 107, 109, 115–117, 123–125, 128, 135, 137, 145, 158, 160, 163, 165
- CLCC** Ceramic Leadless Chip Carrier. 28, 101
- CMC** Calibration and Measurement Capabilities. 117–121, 125, 158, 161, 164, 165
- COLA** Constant Output Level Amplitude. 55, 98, 102, 116
- CPU** Central Processing Unit. 30, 31, 34, 36, 48, 50, 52, 139
- CRC** Cyclic Redundancy Check. 36, 42, 143
- D-SI** Digital Système International (d'unités). 37, 38

- DDS** Direct Digital Synthesis. 51
- DFT** Diskrete Fourier-Transformation. 4, 18, 20, 94–96, 146, 160
- DHCP** Dynamic Host Configuration Protocol. 31, 36, 143
- DOP** Dilution of Precision. 36
- DRI** Data Ready Interrupt. 22, 23, 26, 28, 29, 32, 34, 35, 56, 84, 87, 108, 109, 127
- DUT** Device Under Test. 11, 13, 31, 56, 97, 98, 107, 113, 114, 116, 127–129
- EMPIR** European Metrology Programme for Innovation and Research. 7
- EU** Europäische Union. 127
- FFT** Fast Fourier-Transformation. 63, 80, 84, 87, 94, 134
- FreeRTOS** Free Real-Time Operating System [89]. 30, 33–36, 38, 41, 143
- FT** Fourier-Transformation. 10, 79–84, 87, 90, 91, 139, 160, 169
- Galileo** Galileo GNSS. 7
- GLONASS** russische. Transkription *Global'naya Navigatsionnaya Sputnikovaya Sistema*. 7
- GND** Ground. 32, 55
- GNSS** Global Navigation Satellite System. 7, 30–35, 47–54, 84, 86, 88, 92, 98, 103, 127, 129, 138, 139, 159
- GPDMA** General Purpose DMA. 31
- GPIO** General-Purpose Input/Output. 56
- GPS** Global Positioning System. 7, 47
- GUM** Guide to the Expression of Uncertainty in Measurement [10]. 99
- HAL** Hardware Abstraction Layer. 36
- HDF** Hierarchical Data Format. 30, 43, 45–47, 159
- HF-B-NME** Hochfrequenz-Beschleunigungs-Normalmesseinrichtung. 14, 55, 111, 113, 114, 117–119, 160
- I²C** Inter-Integrated Circuit. 23, 25, 30–33, 107–109
- I²S** Inter-IC Sound. 23, 30, 31
- ICC** Input Capture Counter. 35
- ID** Identifikator. 36–38, 42–44, 47, 158
- IIoT** Industrial Internet of Things. 7
- IIR** Infinite Impulse Response. 133, 134

- IMU** Inertial Measurement Unit. 7, 9, 36, 38, 47, 85, 88, 90, 91, 94, 96, 99, 101, 107–109, 119, 123, 124, 133, 145, 158, 160
- INRIM** Istituto Nazionale di Ricerca Metrologica. 127
- IoT** Internet of Things. 7
- IP** Internet Protocol. 30, 31, 36, 143, 144
- IRQ** Interrupt Request. 31, 32, 34, 35, 47, 159
- ISR** Interrupt Service Routine. 34, 35, 37, 41, 48, 56, 143
- ITC** Ideal Time Course. 28, 29, 51, 52, 84, 86
- JSON** JavaScript Object Notation. 42, 47
- JTAG** Joint Test Action Group. 31–33
- KGT** Kugelgewindetrieb. 132–134, 161
- KiB** Kibibyte 1 KiB=2¹⁰=1024 Byte. 31, 36, 37
- LCD** Liquid-Crystal Display. 31, 36
- LDO** Low-Dropout Regulator. 31, 109, 110
- LDV** Laser-Doppler-Vibrometer. 9, 12, 13, 15–17, 97, 98, 101, 104, 113–116, 120, 121, 143, 146, 159
- LO** Local Oscillator. 114
- LSB** Least Significant Bit. 31, 36, 37, 58, 60, 108, 109
- LTI** Linear Time-Invariant. 10
- LVTTL** Low-Voltage Transistor-Transistor Logic. 50, 53
- MAC** Media-Access-Controller. 31, 35, 36, 42
- Met4FoF** Metrology for the Factory of the Future. 7, 115, 132, 133, 135, 137, 163
- MiB** Mebibyte 1 MiB=2²⁰=1048576 Byte. 31
- ML** Machine Learning. 132–134, 145, 161
- MSB** Most Significant Bit. 37
- NMEA** National Marine Electronics Association standard 0183 [5]. 34, 35, 48
- NMI** National Metrology Institute. 127, 143
- NTP** Network Time Protocol. 127
- NVIC** Nested Vectored Interrupt Controller. 31, 34
- OCXO** Oven-Controlled Crystal Oscillator. 49

- ODR** Output Data Rate. 22, 28
- OLS** Ordinary Least Squares. 18, 20, 66, 158
- PC** Personal Computer. 35, 43, 57, 127, 143, 159
- PDF** Probability Density Function. 81
- PLL** Phase-Locked Loop. 50–54, 65, 85, 86, 88, 89, 91, 93
- ppm** Parts per Million ($\cdot 10^{-6}$). 15, 66, 68, 72, 105, 168
- PPS** Pulse-per-Second. 31, 35, 48, 49, 51, 86, 89, 127
- PSD** Power Spectral Density. 81, 84, 89, 90, 109, 169
- PTB** Physikalisch-Technische Bundesanstalt. 4, 13, 14, 49–51, 55, 99, 101, 102, 107, 109, 111–119, 123–125, 127, 128, 133–135, 137, 145, 158, 160, 163, 164
- PWM** Pulse-Width Modulation. 31
- QFN** Quad-Flat No-Leads. 24, 101, 107, 109, 110
- RAM** Random-Access Memory. 25
- RBW** Resolution Bandwidth. 87
- REF** Referenz. 11
- RISE** Research Institutes of Sweden. 127
- RMS** Root Mean Square $\frac{1}{n} \sqrt{x_0^2 + x_1^2 + \dots + x_n^2}$. 51, 77, 80, 81, 84
- ROM** Read-only Memory. 25, 31
- S3PSA** Sequentielle-Drei-Parameter-Sinus-Approximation. 5, 47, 64–68, 70, 72–74, 76, 94, 99, 105, 114, 128, 130, 159–161, 168
- S4PSA** Sequentielle-Vier-Parameter-Sinus-Approximation. 65, 67, 72, 75, 160
- S/PDIF** Sony/Philips Digital Interface. 116
- SAR** Successive-Approximation. 31, 55
- SAW** Surface Acoustic Wave. 9
- SFDR** Spurious-free Dynamic Range. 78
- SI** Système international (d'unités). 7, 9, 30, 35, 36, 38, 107, 109, 143
- SMA** SubMiniature version A. 32
- SNR** Signal-to-Noise Ratio. 58, 64, 69, 74, 77, 94, 100, 102
- SRAM** Static Random-Access Memory. 31, 32, 36

SRT Sugested Research Topic. 127

SWO Serial Wire Output. 31–33

TCP Transmission Control Protocol. 143, 144

TIE Time Intervall Error. 29, 51–54, 58, 84, 86, 160

UART Universal Asynchronous Receiver-Transmitter. 23, 25

UID Unique Device Identifier. 36, 42

IRQ Timer Update Interrupt Request. 48

USART Universal Synchronous and Asynchronous Receiver-Transmitter. 30–32, 34

UTC Universal Time Coordinated. 48, 49, 144

XML Extensible Markup Language. 143

ZeMA Zentrum für Mechatronik und Automatisierungstechnik. 7, 30, 32, 132–134, 145, 161, 163

Glossar

FreeRTOS-Mail-Queue Eine FreeRTOS-Mail-Queue kombiniert eine Nachrichten-Queue mit einem Shared Memory Pool, also eine Nachrichtenwarteschlange in einem geteilten Speicherbereich. Dadurch werden nur Zeiger auf die Daten zwischen verschiedenen Prozessen oder ISRs übertragen, statt die Daten zu kopieren. Die C-Funktion `osMailGet` lässt den datenlesenden Prozess schlafen, bis eine Nachricht eintrifft oder ein *Timeout* erreicht ist. [34–36](#), [38](#), [41](#)

ACS Das *Analogue Calibration System* nach ISO 16063-11[3] bestimmt die komplexe Übertragungsfunktion $S(\omega)$ eines analogen Sensors durch Abtasten des Frequenzraums mit monofrequenten, sinusförmigen Signalen und Quotientenbildung von Sensor- und Referenzsignal. Das Referenzsignal wird z. B. mit einem **Laser-Doppler-Vibrometer (LDV)** erfasst. [13](#), [15](#), [31](#), [47](#), [55](#), [65](#), [97–105](#), [112](#), [129](#)

Callback Ein *Callback* ist eine Referenz auf eine Funktion, die als Argument an eine andere Funktion übergeben wird. Dabei können *Callbacks* zur Programmlaufzeit geändert werden z. B. kann so das Rendern von neuen Sensordaten in eine Grafikdarstellung zusätzlich aktiviert werden. [45](#)

D-SI Der D-SI Guide enthält Definitionen für maschinenlesbare Messwertdarstellungen im **XML**-Datenformat [\[44\]](#). [36](#), [37](#), [158](#)

DAU Die *Data Acquisition Unit* ist ein auf einem STM32F767zi basierendes System zur Erfassung digitaler Messdaten verschiedener Sensoren mit absoluter Zeitstempelung, das in dieser Arbeit entwickelt und getestet wurde. Die erfassten Daten werden in **SI**-Einheiten konvertiert und mittels eines zustandslosen, auf **Protobuf** basierenden Protokolls sowie über **Ethernet** an einen Empfangs-**PC** übertragen. Neben den Digitaldaten können auch analoge Signale von einem kalibrierten 12-Bit-**ADC** erfasst werden. [7](#), [30–38](#), [41–43](#), [47–55](#), [57](#), [58](#), [84](#), [86–89](#), [91–93](#), [98–105](#), [107](#), [110–112](#), [115](#), [123](#), [127–129](#), [133](#), [134](#), [136](#), [137](#), [144](#), [145](#), [159–161](#)

DC *Direct Current* meint eigentlich Gleichstrom. In dieser Arbeit werden mit **DC** aber alle Gleichanteile von Signalen bezeichnet. Bei einer Messung mit beschränktem Beobachtungszeitraum T_W sind dies alle Frequenzkomponenten $< \frac{1}{T_W}$. [9](#), [10](#), [143](#)

Dict Ein Dict (engl. *Dictionary*) bezeichnet in verschiedenen Programmiersprachen, wie zum Beispiel *Python*, einen strukturierten Datentyp ähnlich eines C **Structs**, in dem mittels Schlüssel-Wert-Paaren (engl. *Key/Value*) Daten abgelegt werden. *Dicts* können verschachtelt werden (engl. *nested*). [44](#), [45](#)

DMA *Direct Memory Access* ermöglicht es Peripheriegeräten, Daten direkt zum und vom Hauptspeicher zu übertragen, ohne die **CPU** zu belasten. Ein **DMA**-Controller übernimmt dabei die Steuerung der Datenübertragung. [26](#), [31](#), [34–36](#), [139](#)

Ethernet Ethernet ist eine Technik, die sowohl Software (Protokolle wie **UDP** oder **TCP**) als auch Hardware (Kabel, Router, Switches, Hubs, Netzwerkkarten usw.) für kabelgebundene Datennetze spezifiziert. Daten werden in sogenannten Frames blockweise übertragen; jedes Frame enthält Start- und Ziel-**IP**-Adresse sowie eine **CRC**-Prüfsumme. Mittels **DHCP** können die **IP**-Adressen von Ethernet-Hosts automatisch bezogen werden. [7](#), [30–33](#), [35](#), [36](#), [42–44](#), [127](#), [143](#)

EURAMET EURAMET e.V. ist die Regional Metrology Organisation (RMO) von Europa und vertritt nationale Metrologieinstitute (**NMIs**) und benannte Institute (**DI**s). Die Organisation fördert die Entwicklung und Verbreitung einer integrierten und wettbewerbsfähigen Metrologie-Infrastruktur in Europa und unterstützt den Wissensaustausch und die Zusammenarbeit zwischen ihren Mitgliedern zur Lösung gesellschaftlicher Herausforderungen in den Bereichen Gesundheit, Energie und Umwelt [\[34\]](#), [144](#)

European Metrology Partnership Die European Metrology Partnership (EMP) ist eine von **EURAMET** vorgeschlagene Forschungs- und Innovationsinitiative, die von den Mitgliedstaaten und der Europäischen Union im Rahmen von Horizon Europe kofinanziert wird. Ziel ist es, eine erstklassige Metrologie-Infrastruktur in Europa zu entwickeln und aufrechtzuerhalten, indem metrologische Lösungen, grundlegende Referenzdaten und Methoden bereitgestellt werden, die die europäische Innovation unterstützen und auf gesellschaftliche Herausforderungen reagieren. Die EMP baut auf den Erfolgen der Programme EMRP und EMPIR auf und stellt sicher, dass die Zusammenarbeit und der Fortschritt im Bereich der Metrologie über 2030 hinaus fortgesetzt werden [\[34\]](#), [127](#)

FIFO Bei einem Pufferspeicher nach dem Prinzip **First In, First Out**, werden die zuerst eingefügten Daten (*Push*) auch zuerst aus dem Speicher entfernt/verarbeitet (*Pop*). Hat der Puffer eine feste Größe, werden beim Erreichen der maximalen Größe entweder alte Pakete verworfen (Ringpuffer) oder keine neuen mehr eingefügt (Blockierender Puffer). [22](#), [23](#), [26](#), [43](#), [108](#), [109](#), [139](#)

MEMS Ein microelectromechanical system (MEMS) ist ein in Dünnschichttechnik, meist auf Halbleitersubstraten, gefertigtes Bauteil, welches Sensoren (z. B. für Beschleunigung, Winkelgeschwindigkeit, magnetische Flussdichte oder Temperatur), Aktoren (z.B. Oszillatoren oder Pumpen) und Steuerungselektronik in einem kleinen Bauteil vereint [\[62\]](#). [7](#), [10](#), [12](#), [21–28](#), [85](#), [87](#), [101](#), [103](#), [107](#), [109](#), [113](#), [115](#), [125](#), [127](#), [129](#), [131](#), [134](#), [135](#), [145](#), [159](#)

Mutex Ein *Mutex*, kurz für *Mutual Exclusion* (wechselseitiger Ausschluss), ist ein Synchronisationsmechanismus. Er wird verwendet, um den gleichzeitigen Zugriff mehrerer **Threads** auf gemeinsame Ressourcen zu verhindern. Wenn die Resource noch nicht verwendet wird führt ein Zugriff auf den *Mutex* dazu, dass der anfragende Prozess den Zugriff erhält, bis dieser Prozess den Zugriff beendet hat. und den *Mutex* wieder freigibt. Alle andere Anfragen werden entweder zurückgewiesen oder in einer Warteschlange aufgereiht und der Reihe nach abgearbeitet. [45](#)

Protobuf Protocol Buffers (Protobuf) ist ein von Google entwickeltes Datenformat zur Datenserialisierung mit einer Schnittstellen-Beschreibungssprache. Code-Generatoren stehen unter anderem für C++, Python und Java zur Verfügung (siehe <https://developers.google.com/protocol-buffers/>). In dieser Arbeit wird ProtobufV2 verwendet. Zur Implementierung auf dem Mikrocontroller der DAU wird Nanopb (<https://github.com/nanopb/nanopb>) verwendet. 35, 36, 38, 39, 41, 43–45, 143, 144

Queue Eine *Queue*, vom englischen *Queue* (Warteschlange), ist eine Datenstruktur, die nach dem **FIFO**-Prinzip arbeitet. In Python stellt das *multiprocessing*-Modul eine *Queue*. Diese *Queue* ermöglicht die synchronisierte Kommunikation und Datenaustausch zwischen verschiedenen *Threads* einer Applikation. 43–45

SPI SPI ist ein synchrones, serielles Kommunikationsprotokoll zur Verbindung von Mikroprozessoren mit Peripheriegeräten. Es nutzt vier Leitungen: *Master Out Slave In* (MOSI) für ausgehende Daten, *Master In Slave Out* (MISO) für eingehende Daten, *Serial Clock* (SCLK) für das Taktsignal, und *Chip Select* (CS), um das anzusprechende Peripheriegerät auszuwählen. Mehrere Geräte können an denselben Bus angeschlossen werden, wobei jedes Gerät eine eigene CS-Leitung zur Auswahl hat. 23, 25, 30–33, 108

Struct Ein *Struct*, vom englischen *Structure*, bezeichnet in verschiedenen Programmiersprachen, insbesondere in C und C++, einen strukturierten Datentyp, der mehrere Variablen verschiedener Typen zusammenfasst. Auf die einzelnen Felder des *Structs* kann individuell zugegriffen werden. Die Anordnung der Variablen im Speicher ist architekturabhängig; daher können *Structs* nicht direkt zur Datenübertragung zwischen Systemen verwendet werden, sondern eine Serialisierung der Daten mit z. B. *Protobuf* ist nötig. 35, 36, 41, 143

Thread Ein *Thread*, vom englischen *Thread* (Faden), bezeichnet in der Informatik die kleinste Ausführungseinheit innerhalb eines *Prozesses*. Mehrere *Threads* können parallel bzw. nebenläufig ausgeführt werden, was eine effiziente Nutzung von Mehrkernprozessoren ermöglicht. Jeder *Thread* besitzt seinen eigenen Ausführungsstatus (*Stack*), teilt jedoch den gemeinsamen Speicher (*Heap*) mit anderen *Threads* desselben *Prozesses*. 43

UDP Das *User Datagram Protocol* (UDP) ist neben *TCP* und *IP* Teil der Internetprotokollfamilie. UDP ermöglicht die effiziente, zustandslose Übertragung von Datagrammen von einem Sender zu einem Empfänger über eine *IP*-Verbindung. Eine Kontrolle über den ordnungsgemäßen Empfang der Pakete ist nicht möglich. 30, 36, 37, 39, 43, 44, 143

Unixzeit Die Unixzeit zählt die seit Donnerstag, dem 1. Januar 1970, 00:00 Uhr *UTC*, vergangenen Sekunden. Sie wird als *int32_t* repräsentiert. 35

9 Eigene Veröffentlichungen

Teile dieser Arbeit wurden in folgenden Publikationen veröffentlicht:

- **Paper 1:** Konzept zur dynamischen Kalibrierung von MEMS-Sensoren mit der in dieser Arbeit vorgestellten Methodik.
Zitationen: 9¹
[88] Benedikt Seeger, Thomas Bruns, and Sascha Eichstädt. Methods for dynamic calibration and augmentation of digital acceleration MEMS sensors. In *19th International Congress of Metrology (CIM2019)*, page 22003. EDP Sciences, 2019. doi:10.1051/metrology/201922003
- **Paper 2:** Praktische Anwendung der hier vorgestellten Methodik auf die Gyroskope in den MPU-9250-IMUs.
Zitationen: 5¹
[86] B Seeger, L Klaus, and D Nordmann. Dynamic calibration of digital angular rate sensors. *ACTA IMEKO*, 9(5):394–400, 2020. doi:10.21014/acta_imeko.v9i5.1008
- **Paper 3:** Anwendung der DAU in Sensornetzen.
Zitationen: 34¹
[30] Sascha Eichstädt, Maximilian Gruber, Anupam Prasad Vedurmudi, Benedikt Seeger, Thomas Bruns, and Gertjan Kok. Toward smart traceability for digital sensors and the industrial internet of things. *Sensors*, 21(6):2019, 2021. doi:10.3390/s21062019
- **Paper 4:** Vorstellung der Methodik dieser Arbeit und Kalibrierung von Sensoren, die nach einem Triggersignal Zeitreihen aufzeichnen; in dieser Arbeit nicht gezeigt.
Zitationen: 14¹
[83] B Seeger and Th Bruns. Primary calibration of mechanical sensors with digital output for dynamic application. *ACTA IMEKO*, 10(3):177–184, sep 2021. doi:10.21014/acta_imeko.v10i3.1075
- **Paper 5:** Vergleich der Kalibrierungen am CEM und der PTB zur Validierung der Methodik.
Zitationen: 3¹
[85] B Seeger, E Jugo, A Bosnjakovic, S Ruiz, and Th Bruns. Comparison in dynamic primary calibration of digital-output accelerometer between CEM and PTB. *Metrologia*, 59(3):035002, 2022. doi:10.1088/1681-7575/ac5a5d
- **Paper 6:** Verwendung der DAU und kalibrierter Sensoren im ZeMA-ML-Prüfstand.
Zitationen: 9¹
[28] Tanja Dorst, Maximilian Gruber, Benedikt Seeger, Anupam Prasad Vedurmudi, Tizian

¹Diese Zitationszahlen wurden am 06.11.2024 aus Google Scholar entnommen.

Schneider, Sascha Eichstädt, and Andreas Schütze. Uncertainty-aware data pipeline of calibrated MEMS sensors used for machine learning. *Measurement: Sensors*, 22:100376, 2022. [doi:10.1016/j.measen.2022.100376](https://doi.org/10.1016/j.measen.2022.100376)

- **Paper 7:** Beschreibung der Einflüsse des Abtastphasenrauschens mit Videoanimationen und Vergleich von DFT und Multisinus-Approximation.

Zitationen: o¹

[80] B. Seeger. Influences of sampling time deviations in dynamic measurements with digital sensors: Mitigating signal integrity degradation. IMEKO 2024 XXIV World Congress, aug 2024

- **Talk 1:** Vorstellung des auf Absolutzeit basierenden Multisensor-Kalibriersystems.

Zitationen: o¹

[84] B Seeger and Th Bruns. Absolute time-based dynamic multi-sensor calibration for MEMS accelerometers A concept study. *5th TC22 Conference on Vibration Measurement (together with the 24th TC3 Conference on the Measurement of Force, Mass and Torque, the 14th TC5 Conference on the Measurement of Hardness, and the 6th TC16 Conference on Pressure and Vacuum Measurement)*, October 2022. [doi:10.21014/tc22-2022.054](https://doi.org/10.21014/tc22-2022.054)

- **Talk 2:** Verwendung von SFP-Netzwerkmodulen als Sender und Empfänger für die LDV-Kalibrierung.

Zitationen: o¹

[41] Volkers Henrik and Seeger Benedikt. Using commonly available network equipment for laser Doppler vibrometer excitation and calibration. In *5th TC22 Conference on Vibration Measurement (together with the 24th TC3 Conference on the Measurement of Force, Mass and Torque, the 14th TC5 Conference on the Measurement of Hardness, and the 6th TC16 Conference on Pressure and Vacuum Measurement)*, Cavtat-Dubrovnik, Croatia, October 2022. URL: [IMEKO-TC22-2022-067.pdf](https://imeko-TC22-2022-067.pdf), [doi:10.21014/tc22-2022.067](https://doi.org/10.21014/tc22-2022.067)

- **Talk 3:** Vorstellung der Kalibriermethodik.

Zitationen: o¹

[82] B. Seeger and T. Bruns. Dynamic calibration of sensors with exclusive digital output. In *SMSI 2021 - System of Units and Metrological Infrastructure*, page 320, digital, May 2021. [doi:10.5162/SMSI2021/D1.4](https://doi.org/10.5162/SMSI2021/D1.4)

Publikationen aus der Zeit der Promotion, die nicht zum direkten Inhalt der Promotion gehören:

- **Paper 8:** Methode zur Bestimmung der Richtungsabhängigkeit der Transferfunktion von optischen Rauheitsmesssystemen.

Zitationen: 9¹

[23] Gaoliang Dai, Ziyang Jiao, Lanting Xiang, Benedikt Seeger, Thomas Weimann, Weichang Xie, and Rainer Tutsch. A novel material measure for characterising two-dimensional instrument transfer functions of areal surface topography measuring instruments. *Surface Topography: Metrology and Properties*, 8(4):045025, 2020. doi:10.1088/2051-672X/abc9d2

- **Paper 9:** Python-Bibliotheken zum Umgang, insbesondere Laden und Speichern, Visualisieren sowie mathematische Operationen unter Berücksichtigung von Einheiten und Unsicherheiten mit den Daten aus dem Digitalen Kalibrierschein (DCC).

Zitationen: 0¹

[81] B. Seeger. Python-based development libraries and web applications to process and store data using a JSON representation XML-DCC. IMEKO 2024 XXIV World Congress, aug 2024

Literaturverzeichnis

- [1] IEEE standard definitions of physical quantities for fundamental frequency and time metrology—random instabilities. *IEEE Std Std 1139-2008*, pages c1–35, 2009. doi:10.1109/IEEESTD.2008.4797525.
- [2] 2021 O-S-D report a market analysis and forecast for optoelectronics, sensors/actuators, and discretes. Technical report, IC Insight, 2021. URL: <https://www.electronicsweekly.com/news/business/sales-mems-sensors-actuators-grow-16-year-2021-06/>.
- [3] Methods for the calibration of vibration and shock transducers - part 11: Primary vibration calibration by laser interferometry. Standard, International Organization for Standardization, Geneva, CH, March 1999. URL: <https://www.iso.org/standard/24951.html>.
- [4] IEEE Standards Association et al. 754-2019-IEEE standard for floating-point arithmetic, 2019. doi:10.1109/IEEESTD.2019.8766229.
- [5] National Marine Electronics Association et al. Publications and standards from the national marine electronics association (NMEA)/NMEA 0183, 2008. URL: <https://www.nmea.org/nmea-0183.html>.
- [6] Leonardo Baldasarre, Alessandro Tocchio, Mikel Azpeitia Urquia, and Sarah Zerbini. Lattice structure for a critically damped high-G MEMS accelerometer. In *2013 symposium on design, test, integration and packaging of MEMS/MOEMS (DTIP)*, pages 1–2. IEEE, 2013. URL: <https://ieeexplore.ieee.org/document/6559414>.
- [7] Minhang Bao and Heng Yang. Squeeze film air damping in MEMS. *Sensors and Actuators A: Physical*, 136(1):3–27, 2007. doi:10.1016/j.sna.2007.01.008.
- [8] Maurice S Bartlett. Periodogram analysis and continuous spectra. *Biometrika*, 37(1/2):1–16, 1950. doi:10.1093/biomet/37.1-2.1.
- [9] Mourad Benmessaoud and Mekkakia Maaza Nasreddine. Optimization of MEMS capacitive accelerometer. *Microsystem Technologies*, 19(5):713–720, 2013. doi:10.1007/s00542-013-1741-z.
- [10] BIPM, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML. Evaluation of measurement data-guide to the expression of uncertainty in measurement, JCGM 100: 2008 GUM 1995 with minor corrections. *Joint Committee for Guides in Metrology*, 2008. URL: https://www.bipm.org/documents/20126/2071204/JCGM_100_2008_E.pdf.
- [11] Boris A Boom, Alessandro Bertolini, Eric Hennes, and Johannes FJ van den Brand. Gas damping in capacitive MEMS transducers in the free molecular flow regime. *Sensors*, 21(7):2566, 2021. doi:10.3390/s21072566.

- [12] Bosch Sensotec. *BMA280 Digital, Triaxial Acceleration Sensor*, aug 2014. Rev. 1.8. URL: https://www.mouser.com/datasheet/2/783/BST-BMA280-DS000-11_published-786496.pdf.
- [13] Th Bruns, D Nordmann, GP Ripper, JH Winther, C Hof, S Ruiz, G da Silva Pineda, H Nozato, A Zuo, L Dickinson, CS Veldman, A Kozlyakovskiy, C Shan, A Ivashchenko, and RA Allen. Final report on the CIPM key comparison CCAUV.V-K5. *Metrologia*, 58(1A):09001, sep 2021. doi: [10.1088/0026-1394/58/1A/09001](https://doi.org/10.1088/0026-1394/58/1A/09001).
- [14] Th Bruns, D Nordmann, GP Ripper, JH Winther, C Hof, S Ruiz, G da Silva Pineda, H Nozato, A Zuo, L Dickinson, et al. Final report on the CIPM key comparison CCAUV. V-K5. *Metrologia*, 58(1A):09001, 2021. doi: [10.1088/0026-1394/58/1A/09001](https://doi.org/10.1088/0026-1394/58/1A/09001).
- [15] K. Burns, K. B. Adams, and J. Longwell. Interference measurements in the spectra of neon and natural mercury. *J. Opt. Soc. Am.*, 40:339–344, 1950. doi: [10.1364/JOSA.40.000339](https://doi.org/10.1364/JOSA.40.000339).
- [16] Alexandre Cabral, Maria Isabel Godinho, Vitor Oliveira, José Manuel Rebordão, and Mário Co-stanzo Nunes. Primary laser vibration metrology: Evaluation of the rocking motion impact in the accuracy of acceleration measurements. In *Eighth International Conference on Vibration Measurements by Laser Techniques: Advances and Applications*, volume 7098, pages 328–339. SPIE, 2008. doi: [10.1117/12.803151](https://doi.org/10.1117/12.803151).
- [17] Qingzhong Cai, Ningfang Song, Gongliu Yang, and Yiliang Liu. Accelerometer calibration with nonlinear scale factor based on multi-position observation. *Measurement Science and Technology*, 24(10):105002, aug 2013. URL: <https://dx.doi.org/10.1088/0957-0233/24/10/105002>, doi: [10.1088/0957-0233/24/10/105002](https://doi.org/10.1088/0957-0233/24/10/105002).
- [18] Claudio E Calosso, Yannick Gruson, and Enrico Rubiola. Phase noise and amplitude noise in DDS. In *2012 IEEE International Frequency Control Symposium Proceedings*, pages 1–6. IEEE, 2012. doi: [10.1109/FCS.2012.6243619](https://doi.org/10.1109/FCS.2012.6243619).
- [19] Domenico Luca Carni and Giuseppe Fedele. Multi-sine fitting algorithm enhancement for sinusoidal signal characterization. *Computer Standards & Interfaces*, 34(6):535–540, 2012. doi: [10.1016/j.csi.2011.03.003](https://doi.org/10.1016/j.csi.2011.03.003).
- [20] Sitan Chen and Ankur Moitra. Algorithmic foundations for the diffraction limit, 2020. arXiv: [2004.07659](https://arxiv.org/abs/2004.07659), doi: [10.48550/arXiv.2004.07659](https://arxiv.org/abs/2004.07659).
- [21] Don Coppersmith and Shmuel Winograd. Matrix multiplication via arithmetic progressions. In *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pages 1–6, 1987. doi: [10.1111/j.1467-7171\(08\)80013-2](https://doi.org/10.1111/j.1467-7171(08)80013-2).
- [22] Nicola Da Dalt and Ali Sheikholeslami. *Understanding Jitter and Phase Noise: A Circuits and Systems Perspective*. Cambridge University Press, 2018. doi: [10.1017/9781316981238](https://doi.org/10.1017/9781316981238).
- [23] Gaoliang Dai, Ziyang Jiao, Lanting Xiang, Benedikt Seeger, Thomas Weimann, Weichang Xie, and Rainer Tutsch. A novel material measure for characterising two-dimensional instrument transfer functions of areal surface topography measuring instruments. *Surface Topography: Metrology and Properties*, 8(4):045025, 2020. doi: [10.1088/2051-672X/abc9d2](https://doi.org/10.1088/2051-672X/abc9d2).

- [24] Giulio D'Emilia, Antonella Gaspari, and Emanuela Natale. Amplitude–phase calibration of tri-axial accelerometers in the low-frequency range by a LDV. *Journal of Sensors and Sensor Systems*, 8(1):223–231, 2019. doi:10.1088/1742-6596/1149/1/012018.
- [25] Alper Demir, Amit Mehrotra, and Jaijeet Roychowdhury. Phase noise in oscillators: A unifying theory and numerical methods for characterisation. In *Proceedings of the 35th annual Design Automation Conference*, pages 26–31, 1998. doi:10.1109/81.847872.
- [26] Richard H Dixon and Jérémie Bouchaud. Markets and applications for MEMS inertial sensors. In *MEMS/MOEMS Components and Their Applications III*, volume 6113, pages 33–42. SPIE, 2006. doi:10.1117/12.647765.
- [27] David L Donoho and Philip B Stark. Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, 49(3):906–931, 1989. doi:10.1137/0149053.
- [28] Tanja Dorst, Maximilian Gruber, Benedikt Seeger, Anupam Prasad Vedurmudi, Tizian Schneider, Sascha Eichstädt, and Andreas Schütze. Uncertainty-aware data pipeline of calibrated MEMS sensors used for machine learning. *Measurement: Sensors*, 22:100376, 2022. doi:10.1016/j.measen.2022.100376.
- [29] Adam Dunkels. Design and implementation of the LWIP TCP/IP stack. *Swedish Institute of Computer Science*, 2(77), 2001. URL: <https://www.artila.com/download/RIO/RIO-2010PG/lwip.pdf>.
- [30] Sascha Eichstädt, Maximilian Gruber, Anupam Prasad Vedurmudi, Benedikt Seeger, Thomas Bruns, and Gertjan Kok. Toward smart traceability for digital sensors and the industrial internet of things. *Sensors*, 21(6):2019, 2021. doi:10.3390/s21062019.
- [31] Sascha Eichstädt, Anke Keidel, and Julia Tesch. Metrology for the digital age. *Measurement: Sensors*, 18:100232, 2021. doi:10.1016/j.measen.2021.100232.
- [32] Arul Elango, René Landry Jr, et al. FPGA-based autonomous GPS-disciplined oscillators for wireless sensor network nodes. *Sensors*, 22(9):3135, 2022. doi:10.3390/s22093135.
- [33] Christine Eun, Xin Luo, Jun-Chieh Wang, Zhongmin Xiong, Mark Kushner, and Yogesh Gianchandani. A microdischarge-based monolithic pressure sensor. *Journal of Microelectromechanical Systems*, PP, mar 2014. doi:10.1109/JMEMS.2014.2312174.
- [34] EURAMET e.V. European metrology partnership, 2024. Accessed: 2024-08-10. URL: <https://www.metpart.eu/>.
- [35] Lu Fang and Lyle Alexander Menk. Gas ingress and egress of MEMS multi-chip modules and MEMS devices. *Advances in Materials*, 8(SAND-2019-9202J), 2019. doi:10.11648/j.am.20190804.17.
- [36] Franco Ferraris, Ugo Grimaldi, and Marco Parvis. Three-axis rate gyros and accelerometers. *Sensors and Materials*, 7(5):311–330, 1995. doi:10.1016/0924-4247(94)80031-6.

- [37] Waqas Amin Gill, Ian Howard, Ilyas Mazhar, and Kristoffer McKee. A review of MEMS vibrating gyroscopes and their reliability issues in harsh environments. *Sensors*, 22(19):7405, 2022. [doi:10.3390/s22197405](https://doi.org/10.3390/s22197405).
- [38] Ali Hajimiri and Thomas H Lee. A general theory of phase noise in electrical oscillators. *IEEE Journal of Solid-State Circuits*, 33(2):179–194, 1998. [doi:10.1109/4.658619](https://doi.org/10.1109/4.658619).
- [39] Peter Handel. Properties of the IEEE-STD-1057 four-parameter sine wave fit algorithm. *IEEE Transactions on Instrumentation and Measurement*, 49(6):1189–1193, 2000. [doi:10.1109/19.893254](https://doi.org/10.1109/19.893254).
- [40] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. [doi:10.1038/s41586-020-2649-2](https://doi.org/10.1038/s41586-020-2649-2).
- [41] Volkers Henrik and Seeger Benedikt. Using commonly available network equipment for laser Doppler vibrometer excitation and calibration. In *5th TC22 Conference on Vibration Measurement (together with the 24th TC3 Conference on the Measurement of Force, Mass and Torque, the 14th TC5 Conference on the Measurement of Hardness, and the 6th TC16 Conference on Pressure and Vacuum Measurement)*, Cavtat-Dubrovnik, Croatia, October 2022. URL: [IMEKO-TC22-2022-067.pdf](https://imeko-tc22-2022-067.pdf), [doi:10.21014/tc22-2022.067](https://doi.org/10.21014/tc22-2022.067).
- [42] Hexagon | NovAtel. *Inertial Navigation Systems and Vibration*, feb 2023. Rev. 1. URL: <https://web.archive.org/web/20230404084342/https://hexagondownloads.blob.core.windows.net/public/Novatel/assets/Documents/Papers/APN-112-Inertial-Navigation-Systems-and-Vibration/APN-112-Inertial-Navigation-Systems-and-Vibration.pdf>.
- [43] Christian Hirt, Sten Claessens, Thomas Fecher, Michael Kuhn, Roland Pail, and Moritz Rexer. New ultrahigh-resolution picture of earth's gravity field. *Geophysical Research Letters*, 40(16):4279–4283, 2013. [doi:10.1002/grl.50838](https://doi.org/10.1002/grl.50838).
- [44] D Hutzschenreuter, F Härtig, W Heeren, T Wiedenhöfer, A Forbes, C Brown, I Smith, S Rhodes, I Linkeová, J Sýkora, et al. Smartcom digital system of units (D-SI) guide for the use of the metadata-format used in metrology for the easy-to-use, safe, harmonised and unambiguous digital transfer of metrological data - second edition. 2020. [doi:10.5281/zenodo.3816686](https://doi.org/10.5281/zenodo.3816686).
- [45] IEEE. IEEE standard for digitizing waveform recorders. *IEEE Std 1057-2017 (Revision of IEEE Std 1057-2007)*, pages 1–o, 2018. [doi:10.1109/IEEEESTD.2018.8291741](https://doi.org/10.1109/IEEEESTD.2018.8291741).
- [46] InvenSense, a TDK Group Company. *MPU-9250 Product Specification*, jun 2016. Rev. 1.1. URL: <https://invensense.tdk.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf>.

- [47] BSEN ISO. ISO 266: 1997: Acoustics-preferred frequencies. *International Organization for Standardization, Geneva, Switzerland*, 1997. URL: <https://www.iso.org/standard/1350.html>.
- [48] Richard Kaußler. URL: https://www.richis-lab.de/MEMS_02.htm.
- [49] M Kiel, O Bohlen, and DU Sauer. Harmonic analysis for identification of nonlinearities in impedance spectroscopy. *Electrochimica Acta*, 53(25):7367–7374, 2008. doi:10.1016/j.electacta.2008.01.089.
- [50] Hayang Kim, Xiaoli Ma, and Benjamin Russell Hamilton. Tracking low-precision clocks with time-varying drifts using Kalman filtering. *IEEE/ACM Transactions on Networking*, 20(1):257–270, 2011. doi:10.1109/TNET.2011.2158656.
- [51] Leonard Klaus. *Entwicklung eines primären Verfahrens zur Kalibrierung von Drehmomentaufnehmern mit dynamischer Anregung*. PhD thesis, Fakultät für Maschinenbau der Gottfried Wilhelm Leibniz Universität Hannover, 2016. URL: <https://oar.ptb.de/files/download/110.20160714.pdf>.
- [52] Robert Kowarsch and Christian Rembe. Heterodyne interferometry at ultra-high frequencies with frequency-offset-locked semiconductor lasers. *Measurement Science and Technology*, 31(7):075201, 2020. doi:10.1088/1361-6501/ab796f.
- [53] J. N. Garlitos L. Paculanan. The easy steps to calculate sampling clock jitter for isolated, precision high speed DAQs. *analog-dialogue*, 55(4), 2021. URL: <https://www.analog.com/media/en/analog-dialogue/volume-55/number-4/the-easy-steps-to-calculate-sampling-clock-jitter-for-isolated-precision-high-speed-daq.pdf>.
- [54] Thomas H Lee and Ali Hajimiri. Oscillator phase noise: A tutorial. *IEEE Journal of Solid-State Circuits*, 35(3):326–336, 2000.
- [55] Torben Licht and Sven Erik Salbl. Iso 16063-11: Uncertainties in primary vibration calibration by laser interferometry-reference planes and transverse motion. *Technical*, page 28, 2009. URL: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=ef09eda8ea5f93ad22f9a6b5099af694b453e727>.
- [56] Thomas E Linnenbrink, Jerome Blair, Sergio Rapuano, Pasquale Daponte, Eulalia Balestrieri, Luca De Vito, Solomon Max, and Steven J Tilden. ADC testing-part 7 in a series of tutorials in instrumentation and measurements. *IEEE Instrumentation & Measurement Magazine*, 9(2):37–47, 2006. doi:10.1109/MIM.2006.1634987.
- [57] Qianbo Lu, Yinan Wang, Xiaoxu Wang, Yuan Yao, Xiaowen Wang, Wei Huang, et al. Review of micromachined optical accelerometers: From mg to sub- μ g. *Opto-Electronic Advances*, 4(3):200045–1, 2021. doi:10.29026/oea.2021.200045.
- [58] Sergiusz Luczak. Experimental studies of hysteresis in MEMS accelerometers: A commentary. *IEEE Sensors Journal*, 15(6):3492–3499, June 2015. doi:10.1109/JSEN.2015.2390778.
- [59] Sergiusz Łuczak, Jakub Wierciak, and Wojciech Credo. Effects of natural aging in biaxial MEMS accelerometers. *IEEE Sensors Journal*, 21(2):1305–1314, 2020. doi:10.1109/JSEN.2020.3017897.

- [60] Björn Ludwig, Bang Xiangyong, Anupam Prasad, Haris Lulic, Maximilian Gruber, and Gertjan Koch. Met4fof/agentmet4fof: v0.13.2. February 2022. doi:10.5281/zenodo.5965562.
- [61] ST Microelectronics. UM1905 *Description of STM32F7 HAL and Low-Layer Drivers*, sep 2020. URL: https://www.st.com/resource/en/user_manual/dm00189702-description-of-stm32f7-hal-and-lowlayer-drivers-stmicroelectronics.pdf.
- [62] Manish Kumar Mishra, Vikas Dubey, PM Mishra, and Isharat Khan. MEMS technology: A review. *Journal of Engineering Research and Reports*, pages 1–24, 2019. doi:10.9734/jerr/2019/v4i116891.
- [63] Conceigio Libano Monteiro, Pasquale Arpaia, and Antonio Cruz Serra. A comprehensive phase-spectrum approach to metrological characterization of hysteretic ADCs. *IEEE Transactions on Instrumentation and Measurement*, 51(4):756–763, 2002. doi:10.1109/TIM.2002.803303.
- [64] David B Newell, Eite Tiesinga, et al. The international system of units (SI). *NIST Special Publication*, 330:1–138, 2019. doi:10.6028/NIST.SP.330-2019.
- [65] Carmine Noviello. *Mastering STM32*. leanpub, 2018. URL: <https://leanpub.com/mastering-stm32>.
- [66] Alvaro Ortega and Federico Milano. Comparison of different PLL implementations for frequency estimation and control. In *2018 18th International Conference on Harmonics and Quality of Power (ICHQP)*. IEEE, May 2018. doi:10.1109/ichqp.2018.8378935.
- [67] Jianye Pan, Chunxi Zhang, and Qingzhong Cai. An accurate calibration method for accelerometer nonlinear scale factor on a low-cost three-axis turntable. *Measurement Science and Technology*, 25(2):025102, 2014. doi:10.1088/0957-0233/25/2/025102.
- [68] Ted Blair Parson, Danelle Mary Tanner, and Thomas Edward Buchheit. Shock margin testing of a one-axis MEMS accelerometer. jul 2008. doi:10.2172/1028884.
- [69] Frank Pedrotti, Leno Pedrotti, Werner Bausch, and Hartmut Schmidt. *Optik für Ingenieure*. Springer, 2005. doi:10.1007/b139018.
- [70] Javier Pérez Sanjurjo, Enrique Prefasi, Cesare Buffa, and Richard Gaggl. A capacitance-to-digital converter for MEMS sensors for smart applications. *Sensors*, 17(6):1312, 2017. doi:10.3390/s17061312.
- [71] Long Pham and Anthony DeSimone. Vibration rectification in MEMS accelerometers. *Analog Devices, Inc*, 2017. URL: <https://www.analog.com/en/technical-articles/vibration-rectification-in-mems-accelerometers.html>.
- [72] Andrea Prato, Fabrizio Mazzoleni, Francesca R Pennecchi, Gianfranco Genta, Maurizio Galetto, and Alessandro Schiavi. Towards large-scale calibrations: A statistical analysis on 100 digital 3-axis MEMS accelerometers. In *2021 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0&IoT)*, pages 578–582. IEEE, 2021. doi:10.1109/MetroInd4.0IoT51437.2021.9488465.

- [73] Andrea Prato, Alessandro Schiavi, Mazzoleni Fabrizio, Amara Toure, Gianfranco Genta, and Maurizio Galetto. A reliable sampling method to reduce large sets of measurements: A case study on the calibration of digital 3-axis MEMS accelerometers. jun 2020. doi:10.1109/MetroInd4.0IoT48571.2020.9138293.
- [74] Pedro M Ramos, M Fonseca da Silva, Raúl Carneiro Martins, and AM Cruz Serra. Simulation and experimental results of multiharmonic least-squares fitting algorithms applied to periodic signals. *IEEE Transactions on Instrumentation and Measurement*, 55(2):646–651, 2006. doi:10.1109/TIM.2006.864260.
- [75] Pedro M Ramos and A Cruz Serra. Least squares multiharmonic fitting: Convergence improvements. *IEEE Transactions on Instrumentation and Measurement*, 56(4):1412–1418, 2007. doi:10.1109/TIM.2007.899873.
- [76] W.J. Riley, Physics Laboratory (U.S.). Time, and Frequency Division. *Handbook of Frequency Stability Analysis*. NIST special publication. U.S. Department of Commerce, National Institute of Standards and Technology, 2007. URL: https://books.google.de/books?id=k_n0nQAACAAJ.
- [77] Xu Ru, Nian Gu, Hang Shang, and Heng Zhang. MEMS inertial sensor calibration technology: Current status and future trends. *Micromachines*, 13(6):879, 2022. doi:10.3390/mi13060879.
- [78] Alessandro Schiavi, Fabrizio Mazzoleni, and Alessandro Germak. Simultaneous 3-axis MEMS accelerometer primary calibration: Description of the test-rig and measurements. In *XXI IMEKO World Congress on Measurement in Research and Industry*, volume 30, pages 2161–2164, 2015. URL: <https://www.imeko.org/publications/wc-2015/IMEKO-WC-2015-TC22-438.pdf>.
- [79] Wolf D Schmidt. *Sensorschaltungstechnik*. Elektronik. Vogel Communications Group, Würzburg, Germany, 3 edition, aug 2007.
- [80] B. Seeger. Influences of sampling time deviations in dynamic measurements with digital sensors: Mitigating signal integrity degradation. IMEKO 2024 XXIV World Congress, aug 2024.
- [81] B. Seeger. Python-based development libraries and web applications to process and store data using a JSON representation XML-DCC. IMEKO 2024 XXIV World Congress, aug 2024.
- [82] B. Seeger and T. Bruns. Dynamic calibration of sensors with exclusive digital output. In *SMSI 2021 - System of Units and Metrological Infrastructure*, page 320, digital, May 2021. doi:10.5162/SMSI2021/D1.4.
- [83] B Seeger and Th Bruns. Primary calibration of mechanical sensors with digital output for dynamic application. *ACTA IMEKO*, 10(3):177–184, sep 2021. doi:10.21014/acta_imeko.v10i3.1075.

- [84] B Seeger and Th Bruns. Absolute time-based dynamic multi-sensor calibration for MEMS accelerometers A concept study. *5th TC22 Conference on Vibration Measurement (together with the 24th TC3 Conference on the Measurement of Force, Mass and Torque, the 14th TC5 Conference on the Measurement of Hardness, and the 6th TC16 Conference on Pressure and Vacuum Measurement)*, October 2022. [doi:10.21014/tc22-2022.054](https://doi.org/10.21014/tc22-2022.054).
- [85] B Seeger, E Jugo, A Bosnjakovic, S Ruiz, and Th Bruns. Comparison in dynamic primary calibration of digital-output accelerometer between CEM and PTB. *Metrologia*, 59(3):035002, 2022. [doi:10.1088/1681-7575/ac5a5d](https://doi.org/10.1088/1681-7575/ac5a5d).
- [86] B Seeger, L Klaus, and D Nordmann. Dynamic calibration of digital angular rate sensors. *ACTA IMEKO*, 9(5):394–400, 2020. [doi:10.21014/acta_imeko.v9i5.1008](https://doi.org/10.21014/acta_imeko.v9i5.1008).
- [87] Benedikt Seeger. Simulation of the effects of correlated and uncorrelated sampling jitter on a sinusoidal signals, sep 2023. [doi:10.5281/zenodo.10111677](https://doi.org/10.5281/zenodo.10111677).
- [88] Benedikt Seeger, Thomas Bruns, and Sascha Eichstädt. Methods for dynamic calibration and augmentation of digital acceleration MEMS sensors. In *19th International Congress of Metrology (CIM2019)*, page 22003. EDP Sciences, 2019. [doi:10.1051/metrology/201922003](https://doi.org/10.1051/metrology/201922003).
- [89] Amazon Web Services. FreeRTOS™ real-time operating system for microcontrollers, 2021. URL: <https://www.freertos.org>.
- [90] Suhas Vishwasrao Shinde. Review of oscillator phase noise models. In *International MultiConference of Engineers and Computer Scientists (IMECS)*, volume 2, 2014. URL: https://www.iaeng.org/publication/IMECS2014/IMECS2014_pp733-740.pdf.
- [91] Yuriy Shmaliy, Oscar Ibarra-Manzano, LJ Arceo-Miquel, and Jorge Munoz-Diaz. An analysis of sawtooth noise in the timing SynPaQ III GPS sensor. *Sensors and Transducers*, 79:1151–1156, may 2007. URL: https://www.sensorsportal.com/HTML/DIGEST/P_141.htm.
- [92] Isaac Skog and Peter Händel. Calibration of a MEMS inertial measurement unit. In *XVII IMEKO World Congress*, pages 1–6. Citeseer, 2006. [doi:10.3390/mi13060879](https://doi.org/10.3390/mi13060879).
- [93] T Michael Souders, Donald R Flach, Charles Hagwood, and G Yang. The effects of timing jitter in sampling systems. In *6th IEEE Conference Record., Instrumentation and Measurement Technology Conference*, pages 199–203. IEEE, 1989. [doi:10.1109/19.50421](https://doi.org/10.1109/19.50421).
- [94] ST Microelectronics. *AN2834 How to Get the Best ADC Accuracy in STM32 Microcontrollers*, may 2021. URL: https://www.st.com/resource/en/application_note/cd00211314-how-to-get-the-best-adc-accuracy-in-stm32-microcontrollers-stmicroelectronics.pdf.
- [95] ST Microelectronics. *DS11532 STM32F765xx STM32F767xx STM32F768Ax STM32F769xx Datasheet*, feb 2021. URL: <https://www.st.com/resource/en/datasheet/stm32f767zi.pdf>.
- [96] STMicroelectronics. *AIS328DQ High-Performance Ultra-Low-Power 3-Axis Accelerometer with Digital Output for Automotive Applications*, apr 2020. URL: <https://www.st.com/resource/en/datasheet/ais328dq.pdf>.

- [97] STMicroelectronics NV. *AN4776 Application Note General-Purpose Timer Cookbook for STM32 Microcontrollers*, 2019. URL: <https://www.st.com/resource/en/application-note/dm00236305-generalpurpose-timer-cookbook-for-stm32-microcontrollers-stmicroelectronics.pdf>.
- [98] Jack A Stone Jr and Jay H Zimmerman. Index of refraction of air. 2001. URL: <https://emtoolbox.nist.gov/wavelength/ciddor.asp>.
- [99] Michał Szermer, Piotr Zająć, Piotr Amrozik, Cezary Maj, Mariusz Jankowski, Grzegorz Jabłoński, Rafał Kiełbik, Jacek Nazdrowicz, Małgorzata Napieralska, and Bartosz Sakowicz. A capacitive 3-axis MEMS accelerometer for Medipost: A portable system dedicated to monitoring imbalance disorders. *Sensors*, 21(10):3564, 2021. doi:10.3390/s21103564.
- [100] The HDF Group. Hierarchical Data Format, Version 5. 2023. <https://www.hdfgroup.org/HDF5/>.
- [101] Akira Umeda, Mike Onoe, Kohji Sakata, Takehiro Fukushima, Kouichi Kanari, Hiroshi Iioka, and Toshiyuki Kobayashi. Calibration of three-axis accelerometers using a three-dimensional vibration generator and three laser interferometers. *Sensors and Actuators A: Physical*, 114(1):93–101, 2004. doi:10.1016/j.sna.2004.03.011.
- [102] SPEKTRA Schwingungstechnik und Akustik GmbH Dresden. *Technical Datasheet SE-09 Calibration Vibration Exciter for High Frequencies*, nov 2023. URL: https://www.spektra-dresden.com/en/product/details/calibration-solutions_device-testing/erreger-und-komponenten_kalibriererreger-und-komponenten/schwingung/se-09.html?file=files/spektra_relaunch/inhalte/produktgruppen/datenblaetter/SPEKTRA_SE-09_Data-Sheet.pdf.
- [103] Hans-Jürgen von Martens, Alfred Link, Hans-Joachim Schlaak, Angelika Taeubner, Wolfgang Wabinski, and Uwe Goebel. Recent advances in vibration and shock measurements and calibrations using laser interferometry. In *Sixth International Conference on Vibration Measurements by Laser Techniques: Advances and Applications*, volume 5503, pages 1–19. SPIE, 2004. doi:10.1117/12.579524.
- [104] Hans-Jürgen von Martens. Invited article: Expanded and improved traceability of vibration measurements by laser interferometry. *Review of Scientific Instruments*, 84(12):181, 2013. doi:10.1063/1.4845916.
- [105] Ruye Wang. *Introduction to Orthogonal Transforms: With Applications in Data Processing and Analysis*. Cambridge University Press, 2012. doi:10.1017/CBO9781139015158.
- [106] Leah A. Wasser. Tutorial hierarchical data formats - what is HDF5?, October 2020. URL: <https://www.neonscience.org/resources/learning-hub/tutorials/about-hdf5>.
- [107] Fu Zhang, Ehsan Keikha, R Max Moore, and Roberto Horowitz. The method of averaging using an adaptive notch filter in vibratory gyroscopes. In *2014 International Symposium on Inertial Sensors and Systems (ISISS)*, pages 1–4. IEEE, 2014. doi:10.1109/ISISS.2014.6782513.

- [108] Guo Zhanshe, Cheng Fucheng, Li Boyu, Cao Le, Lu Chao, and Ke, Song. Research development of silicon MEMS gyroscopes: A review. *Microsystem Technologies*, 21:2053–2066, 2015. doi: [10.3390/mi13081255](https://doi.org/10.3390/mi13081255).

Tabellenverzeichnis

2.1	Bevorzugte $\frac{1}{3}$-Oktaven-Frequenzskalierungsfaktoren	14
2.2	Vier-Parameter-Sinus-Approximation nach Methode der kleinsten Quadrate (OLS)	20
3.1	Verwendete Einheiten im D-SI-Format	36
3.2	Zusammensetzung der IDs	36
4.1	Vier-Parameter-Sinus-Approximation durch sequentielle 3PSA	63
4.2	Übersicht der Simulationsparameter	69
5.1	Unsicherheitsbeiträge für die Magnitude in Anlehnung an ISO 16063-11:1999 A.1.3.1	98
5.2	Unsicherheitsbeiträge für die Phase in Anlehnung an ISO 16063-11:1999 A.1.3.1	99
6.1	Mögliche Datenraten und Filtereinstellungen der IMU MPU-9250	103
6.2	Nominelle Anregungsfrequenzen und Amplituden	105
7.1	Parameter der simulierten Sensordaten in einem Multisensor-Kalibriersystem	124
10.1	CMCs der PTB für Beschleunigungssensoren: Magnitude der Transferfunktion	160
10.2	CMCs der PTB für Beschleunigungssensoren: Phase der Transferfunktion	160
10.3	CMCs des CEM für Beschleunigungssensoren: Magnitude der Transferfunktion	161
10.4	CMCs des CEM für Beschleunigungssensoren: Phase der Transferfunktion	161

Abbildungsverzeichnis

2.1	Frequenz- und Amplitudenbereiche in der Beschleunigungsmesstechnik	10
2.2	Primäres einachsiges Beschleunigungskalibriersystem nach ISO-16063-11	13
2.3	Erreichbare Beschleunigungen eines Swingerregers Typ <i>Spektra SE-09</i> laut Datenblatt	14
2.4	Aufbau eines LDV	15
2.5	Grundlegender Aufbau von MEMS-Sensoren mit digitaler Schnittstelle	21
2.6	Triaxial-MEMS-Beschleunigungssensor <i>AIS328DQ</i> von <i>STMicroelectronics</i>	23
2.7	Blockdiagramm eines triaxialen MEMS-Beschleunigungsaufnehmers <i>AIS328DQ</i> von <i>STMicroelectronics</i>	24
2.8	Funktionsprinzip eines kapazitiven MEMS-Beschleunigungsaufnehmers	26
3.1	Datenerfassung mit der Data Acquisition Unit (DAU)	30
3.2	Data Acquisition Unit mit angeschlossenem Sensor	31
3.3	Data Acquisition Unit ohne analoge Eingangsschaltungen	32
3.4	Zeitverlauf der DAU-Prozesse nach der Sensor-IRQ	33
3.5	Datenfluss innerhalb der DAU	34
3.6	Zustandsloses Daten- und Beschreibungsprotokoll	37
3.7	Datenfluss in der DataReceiver Software auf dem Empfangs-PC	42
3.8	Flussdiagramme des Datenempfangs	43
3.9	Datenstrukturierung in einer HDF-Datei	44
3.10	Struktur der HDF-Datei einer Beschleunigungskalibrierung in <i>HDFView</i>	45
3.11	Aufbau eines STM32-Timer/Counter-Eingangs	46
3.12	Testaufbau für die Absolutzeitbasis	48
3.13	Abweichung der Zeitstempel (TIE) vom idealen Verlauf	50
3.14	Abweichung der zwei DAU-Zeitstempel voneinander	51
3.15	Abweichung der Zeitstempel von zwei DAUs mit eigenem GNSS-Modul und internem Oszillatior bzw. mit gemeinsamem GNSS-Modul und externem Oszillatior	52
3.16	Analoge Signalaufbereitung für den STM32-ADC	55
3.17	Kalibrierung der analogen Phasenreferenz mittels Unterabtastung	56
3.18	Transferfunktion der analogen Phasenreferenz in Betrag und Phase	57
4.1	Darstellung der Fourier-Transformation zeitlich verschobener rechteckiger Beobachtungsfenster	60
4.2	Simulierte Übertragungsfunktion einer 3PSA mit verschiedenen Beobachtungslängen	61
4.3	Auswirkungen der Frequenzkorrektur auf die Phase der S3PSA	65
4.4	Von-Mises-Verteilungsdichtefunktionen und genäherte Normalverteilungen	67
4.5	Simulation einer S3PSA ohne Frequenzkorrektur	70
4.6	Simulation einer S3PSA mit Frequenzkorrektur mit $\frac{\Delta\varphi}{\Delta t}$	71
4.7	Simulation einer S4PSA ohne Frequenzkorrektur	71
4.8	Simulation einer S3PSA nach Frequenzkorrektur durch 4PSA	72

4.9	Auswirkung von Abtastphasenrauschen	73
4.10	Auswirkungen von Abtastphasenrauschen auf ein Sinussignal	75
4.11	Simulation der Auswirkung von normalverteiltem Abtast- <i>Jitter</i> mit $\sigma = 0,2 \Delta t$ auf die 3PSA bzw. FT	78
4.12	Simulation der Auswirkung von quadratischer Abtastzeitpunktsabweichung mit Amplitude $1 \Delta t$ auf die 3PSA bzw. FT	79
4.13	Zeitabweichung der Abtastzeitpunkte verschiedener Sensoren vom äquidistanten Verlauf über 1000 s	80
4.14	Zeitabweichungen der Abtastzeitpunkte (TIE) verschiedener Sensoren vom äquidistanten Verlauf über 10 s	82
4.15	Phasenrauschleistungsdichtespektrum verschiedener Sensoren und der DAU	85
4.16	Auswirkungen des Abtastphasenrauschens verschiedener Sensoren auf ein 80-Hz-Signal für verschiedene Beobachtungsfensterlängen	86
4.17	Monte-Carlo-Simulation der Amplitudenabnahme durch Phasenrauschen bei 10 s Beobachtungsfensterlänge	88
4.18	Monte-Carlo-Simulation der Amplitudenabnahme durch Phasenrauschen für 100 s Beobachtungsfensterlänge	89
4.19	Vergleich von Multisinus-Approximation und DFT für <i>MPU-9250</i> und <i>BMA280</i>	90
5.1	Primäres einachsiges Beschleunigungskalibrierungssystem nach ISO 16063-11	91
5.2	Erweiterung eines analogen Kalibriersystems mit einer DAU.	92
5.3	Unsicherheitsbeiträge der Magnitudenberechnung für die PTB- und CEM-Messungen	95
5.4	Unsicherheitsbeiträge der Phasenberechnung für die PTB- und CEM-Messungen	96
6.1	Blockdiagramm der IMU <i>MPU-9250</i>	102
6.2	<i>MPU-9250</i> auf Aluminiumblock	104
6.3	HF-B-NME der PTB mit DAU und <i>MPU-9250</i>	105
6.4	Blockweise berechnete Standardabweichung zur Detektion des Anregungssignals, PTB-Messungen	106
6.5	Mechanischer Aufbau der HF-B-NME der PTB	107
6.6	Verschaltung und Signale der HF-B-NME der PTB für analoge Sensoren	108
6.7	Beschleunigungskalibriersystem <i>Spektra CS-18</i> am CEM mit DAU und Sensor	109
6.8	Beschleunigungskalibriersystem am CEM	110
6.9	Blockweise berechnete Standardabweichung zur Detektion des Anregungssignals in den CEM-Messungen	111
6.10	Verschiedene Rotationswinkel γ des Sensors bezogen auf das Labor-Koordinatensystem des Schwingerregers	112
6.11	Übertragungsfunktion und Differenz zum Mittelwert des Z-Beschleunigungssensors der IMU <i>MPU-9250</i> auf der HF-B-NME der PTB bei verschiedenen Montagewinkeln	114
6.12	Trajektorien bei verschiedenen Frequenzen	116
6.13	Gemessene Übertragungsfunktionen und Differenz vom gewichteten Mittel des <i>MPU-9250</i> im Vergleich	118
6.14	Differenz vom gewichteten Mittel der Übertragungsfunktionen des <i>MPU-9250</i> mit Angabe der Unsicherheit wie in den CMCs für analoge Sensoren	119

7.1	Konzept für ein 3-achsiges Multisensor-Kalibriersystem mit Absolutzeitbasis	123
7.2	Simulation der abgetasteten Daten im Multisensor-Kalibriersystem	124
7.3	Schematische Darstellung der Erzeugung der Harmonischen an einer Nichtlinearität	125
7.4	Gleichanteil aus Nichtlinearität	126
7.5	DAU mit verschiedenen Sensoren im Prüfstand am ZeMA	127
7.6	Ergebnisse der Lebensdauervorhersage eines Kugelgewindetriebs durch Machine Learning	129
10.1	Übertragungsfunktion einer 3PSA mit verschiedenen Beobachtungslängen	163
10.2	Simulation einer S3PSA ohne Frequenzkorrektur, lange Simulationsdauer	164
10.3	Auswirkungen von Phasenrauschen auf ein 80-Hz-Signal bei verschiedenen Beobachtungszeiten. Lineare Darstellung	165
10.4	Blockschaltbild des Mikrocontrollers <i>STM32F767ZI</i>	166

Code Listings

3.1	ProtoBuf-Code DataMessage	38
3.2	ProtoBuf-Code DescriptionMessage	39
3.3	Virtuelle Klasse als gemeinsame Treiber-API in C++	40
4.1	Mittelwertbildung in Polarkoordinaten in Python	68
10.1	Berechnung und Plot der Unsicherheitsellipse mit Pearson-Korrelation in Python	162

Danksagung

An dieser Stelle möchte ich mich bei allen bedanken, die mich während meiner spannenden Promotionszeit beruflich und privat begleitet haben. Mein Dank gilt Henrik Volkers², der mich im Frühjahr 2018, als ich mit ihm Fragen zu Schwingungseinflüssen diskutierte, darauf hinwies, dass für das bald startende Met4FoF-Projekt noch ein Doktorand gesucht wird, der Interesse an Mess-technik und digitaler Datenerfassung/-verarbeitung hat, und „das wäre bestimmt mein Ding“. Mein nächster Dank gilt Thomas Bruns, der es mir dann auch als Chef ermöglicht hat, dieses Promotionsprojekt in der Arbeitsgruppe Darstellung Beschleunigung der PTB durchzuführen, und der mir auch jederzeit mit Antworten und Kommentaren zu all meinen Fragen und Ideen zur Seite stand. Zitat: „Sie sind voller Fragen mein Sohn.“ Ebenso danke ich Prof. Dr.-Ing. Rainer Tutsch, der mich nach motiviert hat, nach meiner Masterarbeit auch noch zu promovieren und dann diese Arbeit als Betreuer an der Technischen Universität Braunschweig mit großem Interesse und regelmäßigen Gesprächen begleitet. Weiterhin danke ich Herrn Prof. Dr. Meinhard Schilling für das Begutachten dieser Arbeit und seine Mitwirkung bei der Einführung des Studienganges „Metrologie und Mess-technik“ an der TU Braunschweig, welcher mir große Freude bereitet hat und mich schließlich zu dieser Promotion und an die PTB geführt hat. Auch möchte ich Prof. Michael Müller dafür danken, dass er den Vorsitz der Promotionskommission für diese Arbeit übernimmt. Mein nächster Dank gilt dem wohl fleißigsten Leser dieser Arbeit und aller meiner Fachpublikationen, Michael Kobusch, der mir durch intensives Korrekturlesen geholfen hat, tausende von Rechtschreib- und Formfehler zu beseitigen. Bedanken möchte ich mich auch bei Sascha Eichstädt, Björn Ludwig, Maximilian Gruber und Anupam Vedurmudi, die als PTB-Kollegen ebenfalls direkt am Met4FoF-Projekt beteiligt waren. Ebenso möchte ich mich bei Tanja Dorst vom ZeMA für die angenehme Zusammenarbeit bedanken, schließlich zeigen Ihre Ergebnisse, dass die Abtastzeitabweichungen, die im Mittelpunkt dieser Arbeit stehen, nicht nur ein akademisch interessantes, sondern auch ein in realen Messungen relevantes Problem sind. Mein Dank gilt auch den internationalen Projektpartnern, insbesondere Salustiano Ruiz González (CEM) für die Durchführung der Vergleichsmessungen am CEM und Ehlilmana Jugo vom Metrology Institute (IMBIH) für die Erstellung der Publikation zu diesem Vergleich. Auch Jannis Wowra möchte ich für den intensiven fachlichen Austausch danken. Meinen Kollegen Denis Nordman, Leonard Klaus und Na Yan danke ich für die Unterstützung bei den Messungen an den verschiedenen Einrichtungen. Unserem Mechaniker Sven Clusmann danke ich für die Herstellung der Sensoradapter und die schnelle Lösung aller anderen mechanischen Probleme. Insgesamt möchte ich mich bei allen meinen direkten Kolleginnen und Kollegen, Holger Schönekeß, Maik Röhr, Susanne Gazioch, Vanessa Stehr und Stephanie Koepsell für die nette und angenehme Arbeitsatmosphäre bedanken, die dafür gesorgt hat, dass sich die Zeit der Promotion nicht wie „Arbeit“ angefühlt hat, sondern ich jeden Tag an einen Ort voller interessierter und netter Menschen kommen konnte, mit denen man gerne Zeit verbringt und auf deren Unterstützung man sich verlassen kann. Mein letzter und wichtigster Dank gilt meiner Frau Rita und Tochter Johanna, die mich in dieser Zeit immer verständnissvoll begleitet und unterstützt haben.

²Die Reihenfolge der genannten Personen folgt einer zeitlichen oder organisatorischen Strukturierung und stellt keine Priorisierung irgendwelcher Art dar.

10 Anhang

10.1 CMCs für Beschleunigung der beteiligten nationalen Metrologieinstitute

Tabelle 10.1: CMCs der PTB für Beschleunigungssensoren: Magnitude der Transferfunktion aus [14].

Disturbing Component	comment	95% value	distribution	factor	combined frequency ranges			
					10 Hz to 5 kHz	to 10 kHz	to 15 kHz	to 20 kHz
frequency of SAM	deviation of sample clock from generator clock		rectangular	1,73205081	5,77E-05	5,77E-05	5,77E-05	5,77E-05
Accelerometer Voltage	sampling of HP3458A	5,00E-04	rectangular	1,73205081	2,89E-04	2,89E-04	2,89E-04	2,89E-04
Velocity amplitude	wave length, optical adjustment, deviation between the two beams	1,16E-05	normal	2	5,80E-06	5,80E-06	5,80E-06	5,80E-06
harmon. Distortion	mainly 1st harmonic		Steiner	1	7,84E-06	7,84E-06	7,84E-06	7,84E-06
Humm on Voltage	typical 1mV	5,00E-07	Steiner	1,00	5,00E-07	5,00E-07	5,00E-07	5,00E-07
Noise on Voltage	MC on influence to SAM duration 20ms, Un=1,0mV		normal	1	3,30E-06	3,30E-06	3,30E-06	3,30E-06
Transverse Motion	S(transv) = 0,7% a(transv) < 4%		u-type	1,41421356	1,98E-04	1,98E-04	1,98E-04	1,98E-04
Base strain sensitivty	S = 0,005m/s ² / μ C	0,000005	rectangular	1,73	2,89E-06	2,89E-06	2,89E-06	2,89E-06
mounting	$\epsilon < 0,1 \text{ nm/m}$	0,00012	rectangular	1,73205081	6,93E-05	6,93E-05	6,93E-05	6,93E-05
Temperature	S=2,5e-4 /K dT = 0,3 K	0,000075	rectangular	1,73	4,33E-05	4,33E-05	4,33E-05	4,33E-05
Magnetic field	S=1/a * (m/s ²)T B < 0,3mT	0,0000003	rectangular	1,73	1,73E-07	1,73E-07	1,73E-07	1,73E-07
Airborne acoustics	S=0,008 m/s ² at 154 dB max sound level 88 dB	8,00E-08	rectangular	1,73	4,62E-08	4,62E-08	4,62E-08	4,62E-08
Noise on Interferom.	noise level equiv. of 2 nm after demodulation, Monte Carlo		normal	1	1,10E-04	3,00E-04	6,20E-04	1,10E-03
a-synchronous Measurement	voltage/acceleration/voltage	1,00E-04	rectangular	1,73	5,77E-05	5,77E-05	5,77E-05	5,77E-05
charge ampl. calibration					2,12E-04	2,12E-04	2,12E-04	2,12E-04
resid. influences		1,00E-04	normal	1,41421356	7,07E-05	7,07E-05	7,07E-05	7,07E-05
exp. std. dev					2,30E-04	1,60E-04	4,40E-04	6,60E-04
rel. std. uncertainty	in %				0,0501	0,0549	0,0874	0,1353
rel. comb. exp. Uncertainty (k=2)	in %				0,1002	0,1098	0,1748	0,2707
stated rel. comb. exp. Uncertainty	in %				0,1000	0,3000	0,5000	1,0000

Tabelle 10.2: CMCs der PTB für Beschleunigungssensoren: Phase der Transferfunktion aus [14].

Disturbing Component	comment	95% value	distribution	factor	combined frequency ranges		
					10 Hz to 1 kHz	to 10 kHz	to 20 kHz
Channel a-synchronisity	all frequencies	< 10 ns	normal	2	1,80E-03	1,80E-02	3,60E-02
Hum (50 Hz)	Monte Carlo, multiples of 20ms are evaluated	equivalent displacement amp. 4 μ m	normal	1	8,00E-03	1,00E-03	1,00E-03
Noise on accelerometer							
Voltage output	Monte Carlo, SNR=500	< 2mV @ 1V	normal	1	4,00E-04	4,00E-04	4,00E-04
Transverse/Rocking motion	1 % transv. Sensitivity @ 10% transv. Excitation	rel. Phase 0 ... 2pi	U-type (by MC)	1	7,00E-04	7,00E-04	7,00E-04
delay of Laser Vibrom. + Mixer + Filter	absolut correction 1,54 μ s applied	uncert. of correction 60 ns	rectang.	1,73	1,25E-02	1,25E-01	2,49E-01
Calibration Charge Amplifier B&K 2650	including Stability, reproducibility, methode (black box)	<0,02°	normal	2	2,00E-02	2,00E-02	2,00E-02
Noise on heterodyne interferometer channel	noise level equiv. of 2 nm after demodulation, Monte Carlo	< 2nm	normal	1	1,43E-04	1,43E-02	5,73E-02
Motion disturbance	drift, relative motion evaluation as velocity and period by period	estimated < 0,02°	normal	2	1,00E-02	1,00E-02	1,00E-02
exp. Std. deviation		typical < 0,02°	normal	2	1,00E-02	1,50E-02	2,00E-02
std. uncertainty	in 1°				0,029	0,130	0,260
exp. Uncertainty (k=2)	in 1°				0,057	0,259	0,520
stated exp. Uncertainty	in 1°				0,200	0,500	1,000

Tabelle 10.3: **CMCs des CEM für Beschleunigungssensoren: Magnitude der Transferfunktion aus [14].**

Description	Frequency range (Hz)	Relative expanded uncertainty (%)	Probability distribution	Factor	Uncertainty type	Uncertainty contribution
Laser	0.4 - 20 000	0.26	normal	2	B	0.130
Voltage	0.4 - 20 000	0.06	normal	1	B	0.060
Hum and noise	0.4 - 20 000	0.02	rectangular	1.732	B	0.012
Mounting	0.4 - 20 000	0.1	rectangular	1.732	B	0.058
Transverse motion	0.4 - 20 000	0.12	special	1	B	0.120
Reproducibility and repeatability	0.4 < 10	0.1	normal	1	B	0.10
	10 - 5 000	0.05				0.05
	>5 000 - 10 000	0.35				0.35
	>10 000 - 15 000	0.5				0.70
	>15 000 - 20 000	1				1.00
Temperature response	0.4 - 20 000	0.1	rectangular	1.732	B	0.058
Relative motion, instability, geometric location of the laser beam	0.4 - 20 000	0.1	rectangular	1.732	B	0.058
Influence on voltage from magnetic field from exciter	0.4 - 20 000	0.05	rectangular	1.732	B	0.029
Temperature influence on sensor	0.4 - 20 000	0.04	rectangular	1.732	B	0.023
					Expanded uncertainty (0.4 Hz - < 10 Hz)	0.5
					Expanded uncertainty (10 Hz - 5 kHz)	0.4
					Expanded uncertainty (>5 kHz - 10 kHz)	0.8
					Expanded uncertainty (>10 kHz - 15 kHz)	1.5
					Expanded uncertainty (>15 kHz - 20 kHz)	2.0

Tabelle 10.4: **CMCs des CEM für Beschleunigungssensoren: Phase der Transferfunktion aus [14].**

Description	Frequency range (Hz)	Expanded uncertainty (1°)	Probability distribution	Factor	Uncertainty type	Uncertainty contribution		
Laser	0.4 - 20 000	0.1	normal	2	B	0.05		
Voltage	0.4 - 20 000	0.5	normal	2	B	0.25		
Hum and noise	0.4 - 20 000	0.02	rectangular	1.732	B	0.012		
Mounting	0.4 - 20 000	0.1	rectangular	1.732	B	0.058		
Reproducibility and repeatability	0.4 - 1 000	0.05	rectangular	1.732	B	0.03		
	>1 000 - 10 000	0.5				0.29		
	>10 000 - 15 000	1.1				0.64		
	>15 000 - 20 000	1.6				0.92		
Temperature response	0.4 - 20 000	0.01	rectangular	1.732	B	0.006		
Influence on voltage from magnetic field from exciter	0.4 - 20 000	0.02	rectangular	1.732	B	0.012		
Relative motion, instability, geometric location of the laser beam	0.4 - 1 000	0.05	rectangular	1.732	B	0.03		
	>1 000 - 10 000	0.5				0.29		
					Expanded uncertainty (0.4 Hz - 1 kHz)	0.5		
					Expanded uncertainty (>1 kHz - 10 kHz)	1.0		
					Expanded uncertainty (>10 kHz - 15 kHz)	1.5		
					Expanded uncertainty (>15 kHz - 20 kHz)	2.0		

10.2 Berechnung der Unsicherheitsellipsen mit Pearson-Korrelation in Python

Code Listing 10.1: Berechnung und Plot der Unsicherheitsellipse mit Pearson-Korrelation in Python

```

1 import matplotlib.pyplot as plt
2 from matplotlib.patches import Ellipse
3 import matplotlib.transforms as transforms
4
5 def confidence_ellipse(x, y, ax, n_std=3.0, facecolor='none', **kwargs):
6     """
7         Create a plot of the covariance confidence ellipse of `x` and `y`
8         Parameters
9         x, y : array_like, shape (n, )
10        Input data.
11        ax : matplotlib.axes.Axes
12        The axes object to draw the ellipse into.
13        n_std : float
14        The number of standard deviations to determine the ellipses radii.
15        Returns
16        matplotlib.patches.Ellipse
17        Other parameters
18        kwargs : `~matplotlib.patches.Patch` properties
19        Source
20        https://matplotlib.org/3.1.1/gallery/statistics/confidence_ellipse.html#sphx-←
21        glr-gallery-statistics-confidence-ellipse-py
22        """
23        if x.size != y.size:
24            raise ValueError("x and y must be the same size")
25        cov = np.cov(x, y)
26        pearson = cov[0, 1]/np.sqrt(cov[0, 0] * cov[1, 1])
27        # Using a special case to obtain the eigenvalues of this two-dimensional ←
28        # dataset.
29        ell_radius_x = np.sqrt(1 + pearson)
30        ell_radius_y = np.sqrt(1 - pearson)
31
32        # Calculating the standard deviation of x from the square root of the variance←
33        # and multiplying with the given number of standard deviations.
34        scale_x = np.sqrt(cov[0, 0]) * n_std
35        mean_x = np.mean(x)
36        # calculating the standard deviation of y ...
37        scale_y = np.sqrt(cov[1, 1]) * n_std
38        mean_y = np.mean(y)
39        ellipse = Ellipse((0, 0), width=ell_radius_x * 2, \
40                           height=ell_radius_y * 2, facecolor=facecolor, **kwargs)
41        transf = transforms.Affine2D().rotate_deg(45) \
42            .scale(scale_x, scale_y).translate(mean_x, mean_y)
43
44        ellipse.set_transform(transf + ax.transData)
45        return ax.add_patch(ellipse)

```

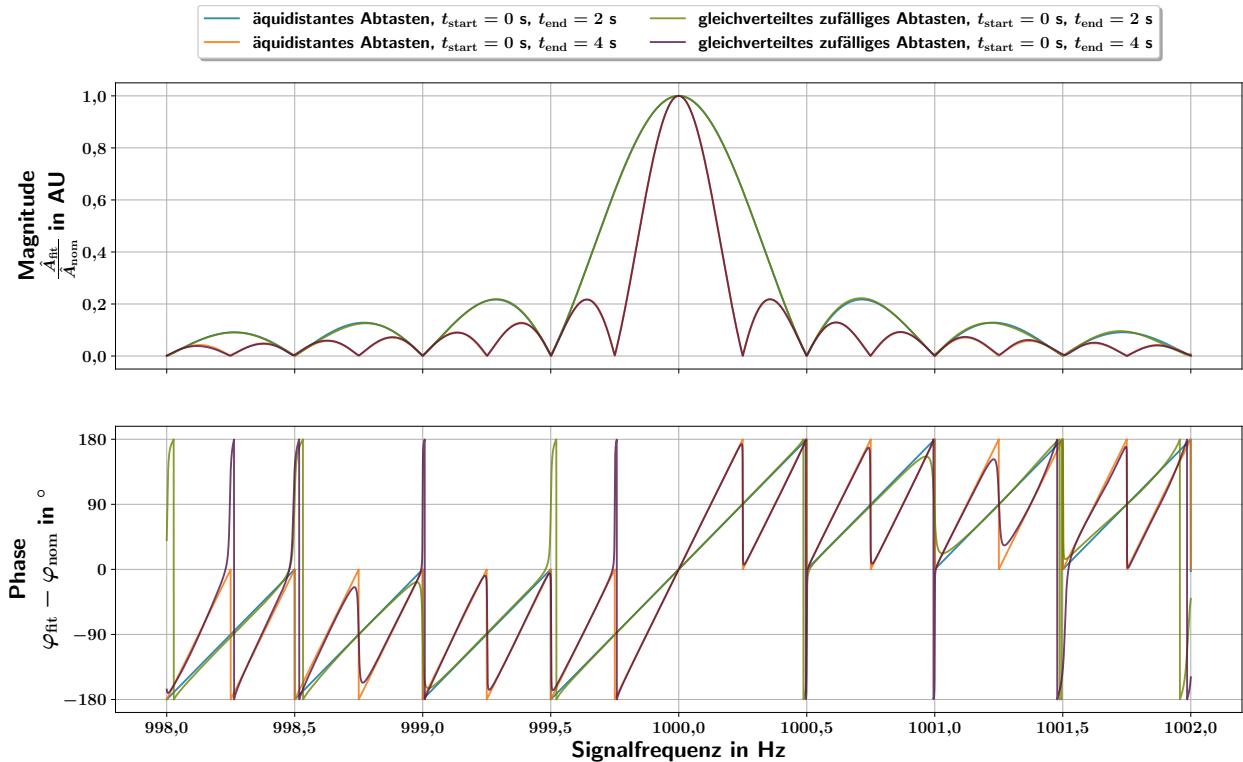


Abbildung 10.1: **Übertragungsfunktion einer 3PSA mit verschiedenen Beobachtungslängen.** Beim zufälligen Abtasten kommt es an den Stellen $n \cdot \frac{1}{T_W}$ aufgrund der gegen Null konvergierenden Amplitude zu deutlichen Abweichungen der Phase. Für die Magnitude ist keine Abweichung zwischen äquidistantem oder gleichverteiltem zufälligem Abtasten sichtbar.

10.3 Grafiken

10.3.1 Gleichverteilt zufälliges Abtasten bei Sinus-Approximation

Die Abbildung 10.1 zeigt die Amplitudensensitivitäten und Phasenabweichungen für die 3PSA für Fensterbreiten von 1 oder 4 Sekunden bei äquidistanten und gleichverteilten Zufallsabtastungen mit genau bekannten Zeitpunkten. Die Amplitudensensitivitäten sind identisch, da sie nur von der Fourier-Transformierten der Wahrscheinlichkeitsdichtefunktionen der Abtastimpulse im Beobachtungsfenster abhängen. Diese Wahrscheinlichkeitsdichtefunktionen sind für beide Abtastarten identisch und damit auch die Amplitudensensitivität und die Phasenabweichung. Ein Unterschied ergibt sich nur für Signale außerhalb des Basisbandes ($f_{sig} > \frac{f_s}{2}$), also bei Unterabtastung. Durch die Periodizität der Abtastimpulse bei äquidistanter Abtastung ist auch die Fourier-Transformierte periodisch, mit einer Wiederholung nach $\frac{f_s}{2}$; dies führt zu sogenanntem Aliasing. In der hier gezeigten Simulation wird $\frac{f_s}{2}$ mit 10 kHz gewählt, sodass bei äquidistanter Abtastung ein Signal mit einer Frequenz von 1 kHz und eines mit 11 kHz Frequenz nicht unterscheiden werden kann.

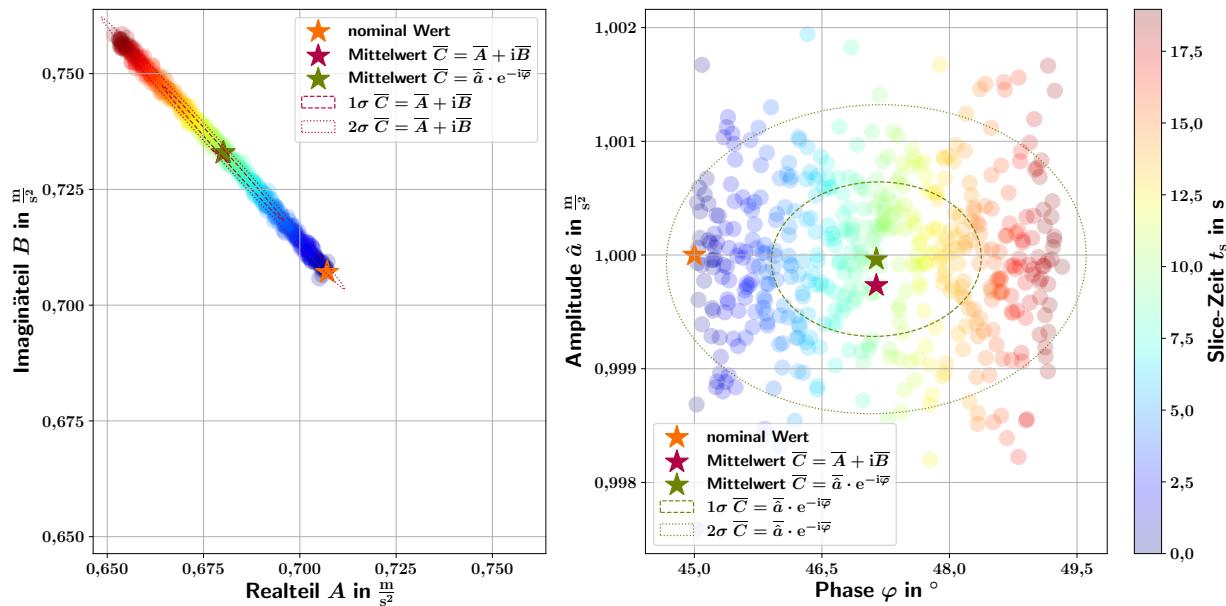


Abbildung 10.2: **Simulation einer S3PSA ohne Frequenzkorrektur, lange Simulationsdauer.** Es ist eine starke zeitlich Korrelation erkennbar; die Phasenwerte der frühen *Slices* (blau) weisen nur eine geringe Phasenabweichung zum Nominalwert von 45° auf, die späten *Slices* (rot) weichen um fast 4° vom Nominalwert ab. Die bogenförmige Verteilung mit dem Radius 1 ist durch die zehn mal so lange Simulationsdauer, verglichen mit Abb. 4.5 von 190 s, gut erkennbar.

10.3.2 Verteilung der S3PSA-Ergebnisse bei Frequenzabweichungen

Die Abbildung 10.2 zeigt die durch eine abweichende Approximationsfrequenz ($-2,1$ ppm) hervorgerufene zeitabhängige Phasenabweichung (erkennbar an der farblich gekennzeichneten Zeit der einzelnen *Slices*) für eine lange Simulationsdauer von 190 s. In Polarkoordinaten (rechts) sind Amplitude und Phase unabhängig voneinander. In kartesischen Koordinaten (links) kann nicht korrekt gemittelt werden, denn durch die bogenförmige Verteilung beschreibt selbst das Einbeziehen von Kovarianzen die Verteilung nicht korrekt, da die Kovarianzen immer nur Ellipsen beschreiben können.

10.3.3 Darstellung der Auswirkungen von Phasenrauschen mit linearer Y-Achse

Die Abbildung 10.3 zeigt die Auswirkungen des Abtastphasenrauschen auf ein 80-Hz-Signal mit einer linearen Y-Achse. Die 3PSA bestimmt das Integral des Produkts des durch Abtastphasenrauschen verbreiterten idealen Signals (durchgehende Linien) und der durch die Beobachtungslänge verursachten Sinc-förmigen Sensitivitäten im Frequenzraum (gestrichelte Linien). Für den Sensor BMA 280 zeigt sich eine breite Verteilung der Signalamplitude, die signifikante Anteile auch noch außerhalb der Breite des 30 s langen Beobachtungsfensters enthält.

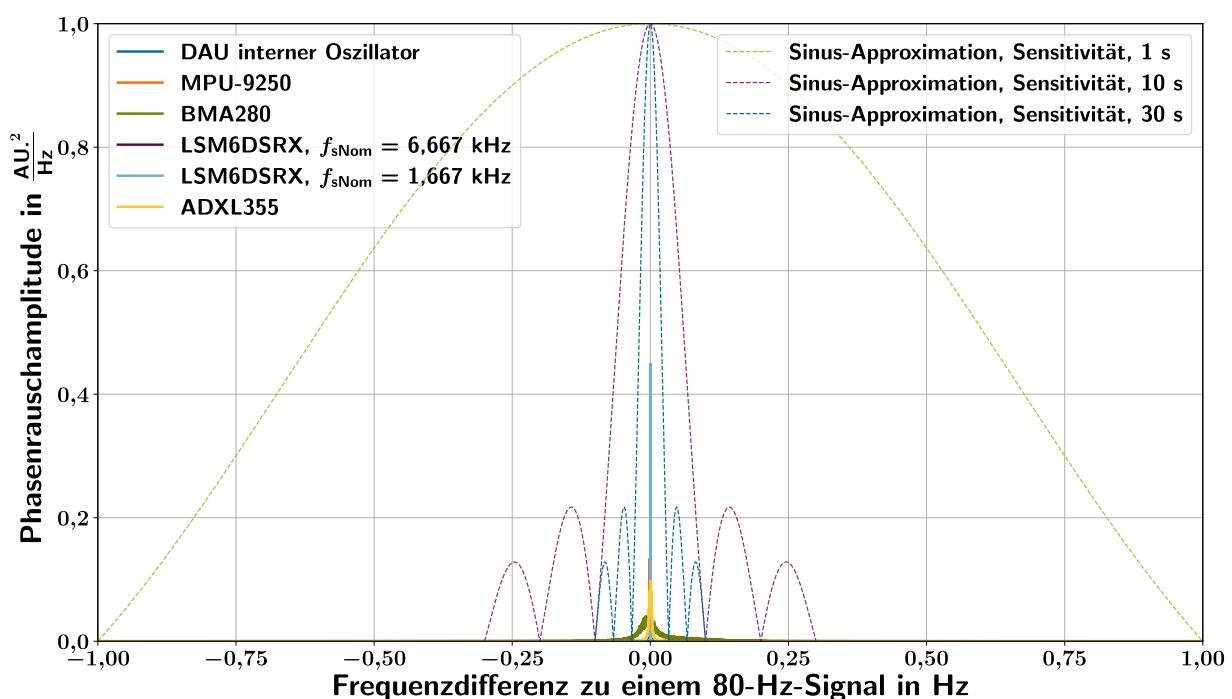
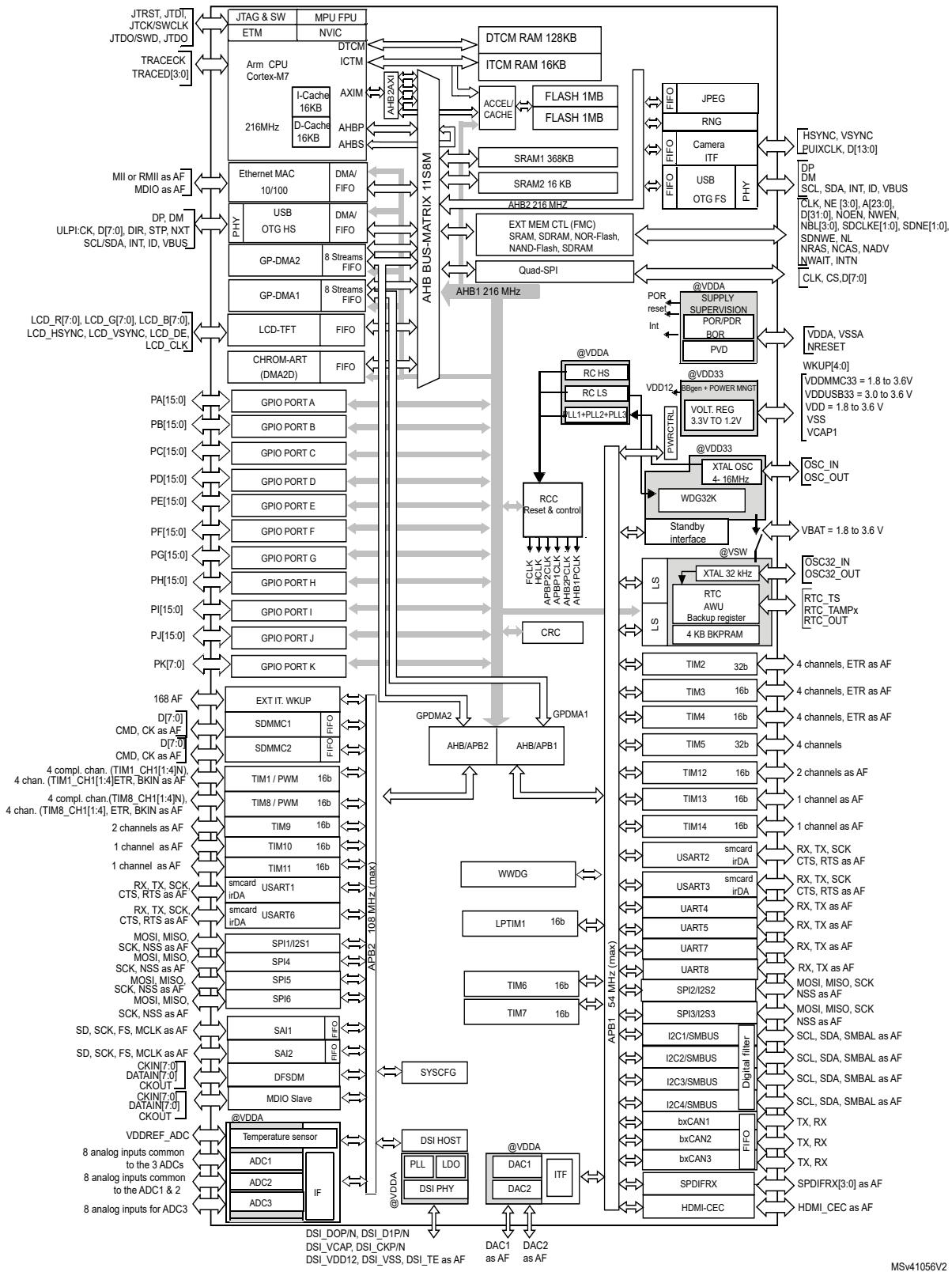
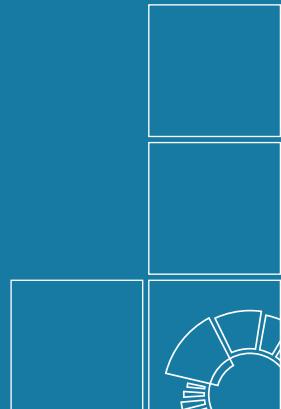


Abbildung 10.3: **Auswirkungen von Phaserauschen auf ein 80-Hz-Signal bei verschiedenen Beobachtungszeiten, lineare Darstellung.** Dargestellt ist die PSD eines simulierten 80 Hz Signals mit der Verbreiterung durch Phaserauschen der Sensoren. Die Sinc-förmigen Sensitivitäten von FT oder 3PSA mit 1, 10 und 30 Sekunden Beobachtungszeit sind gestrichelt dargestellt. Die approximierte Amplitude ist das Produkt aus Sensitivität und dem durch Phaserauschen verbreiterten Signal.



MSv41056V2

Abbildung 10.4: **Blockschaltbild des Mikrocontrollers STMicroelectronics STM32F767ZI.** [95]



Herausgeber:
Physikalisch-Technische Bundesanstalt
ISNI: 0000 0001 2186 1887
Presse und Öffentlichkeitsarbeit
Bundesallee 100
38116 Braunschweig
Telefon: (05 31) 592-93 21
Telefax: (05 31) 592-92 92
www.ptb.de