

Condition Monitoring

Anwenderhandbuch und Datenblatt X20(c)CM4810

Version: **2.21 (Februar 2021)**
Bestellnr.: **MAX20CM4810-GER**

Originalbetriebsanleitung

Alle Angaben entsprechen dem aktuellen Stand zum Zeitpunkt der Erstellung des Handbuches. Inhaltliche Änderungen dieses Handbuches behalten wir uns ohne Ankündigung vor. Die B&R Industrial Automation GmbH haftet nicht für technische oder redaktionelle Fehler und Mängel in diesem Handbuch. Außerdem übernimmt die B&R Industrial Automation GmbH keine Haftung für Schäden, die direkt oder indirekt auf Lieferung, Leistung und Nutzung dieses Materials zurückzuführen sind. Wir weisen darauf hin, dass die in diesem Dokument verwendeten Soft- und Hardwarebezeichnungen und Markennamen der jeweiligen Firmen dem allgemeinen warenzeichen-, marken- oder patentrechtlichen Schutz unterliegen.

1 Allgemeines	7
1.1 Coated Module	7
2 Bestelldaten	8
3 Technische Daten	9
3.1 Technische Daten	9
3.2 Status-LEDs	10
3.3 Firmware-Updatezeit	10
3.4 Anschlussbelegung	10
3.5 Eingangsschema	11
3.6 Schirmung	11
3.7 Derating	11
3.8 Gainverlauf	14
3.9 Einschwingzeit	14
3.10 Sensorauflösung	15
3.11 Verwendung einer B&R Compact CPU bzw. Feldbus CPU	15
4 Registerbeschreibung	16
4.1 Allgemeine Datenpunkte	16
4.2 Funktionsmodell 0 - Standard	16
4.3 Funktionsmodell 1 - Schneller Master	19
4.4 Funktionsmodell 2 - Langsamer Master	21
4.5 Funktionsmodell 254 - Bus Controller	23
4.5.1 Verwendung des Moduls am Bus Controller	24
4.5.2 CAN-I/O Bus Controller	24
4.6 Allgemeine Informationen	25
4.6.1 Signalgenerierung	25
4.6.2 Filterkonfiguration	25
4.6.3 Frequenzbänder	26
4.6.4 Automatische Berechnung des Geschwindigkeitssignals	26
4.6.5 Autogain, AutogainDelay und Überlauf	26
4.6.6 Begriffsdefinition: Abtastezeit und Abtastfrequenz	26
4.7 Allgemeine Register	27
4.7.1 ActSpeed	27
4.7.2 AutogainDelay	27
4.7.3 AutogainDelayRead	27
4.7.4 Control	28
4.7.5 SensorConfig	29
4.7.6 SensorConfigRead	29
4.7.7 Status	30
4.8 AnalogInput-Funktionen	31
4.8.1 Messwerte	31
4.8.2 Kennwertberechnung im AnalogInput	31
4.8.2.1 Kontinuierlicher Modus mit Enable (Continuous Mode)	32
4.8.2.2 Trigger Modus (Single Shot)	32
4.8.3 AnalogInput	32
4.8.4 AnalogInputConfig	33
4.8.5 AnalogInputConfigRead	33
4.8.6 AnalogInputControlByte	34
4.8.7 AnalogInputSamples	34
4.8.8 AnalogInputScale	35
4.8.9 AnalogInputScaleRead	35
4.8.10 SamplesAnalogInput	36
4.8.11 SamplesAnalogInputRead	36
4.9 Automation Runtime Unterstützung	37
4.9.1 DataConsistentWithLockedBuffers	37

4.9.2 DataToggleBit.....	37
4.9.3 OverflowAnalogInput.....	38
4.9.4 OverflowCharacteristicValues.....	38
4.9.5 OverflowFrequencyBands.....	39
4.9.6 PeakHighFrequencyRef.....	39
4.9.7 PeakHighFrequencyRefCalculated.....	39
4.9.8 PeakRawRef.....	39
4.9.9 PeakRawRefCalculated.....	39
4.9.10 RmsHighFrequencyRef.....	40
4.9.11 RmsHighFrequencyRefCalculated.....	40
4.9.12 RmsRawRef.....	40
4.9.13 RmsRawRefCalculated.....	40
4.9.14 SensitivitySensor.....	40
4.10 Kennwerte.....	41
4.10.1 Betragsmaximalwert.....	42
4.10.1.1 PeakHighFrequency.....	42
4.10.1.2 PeakRaw.....	42
4.10.2 Effektivwert.....	43
4.10.2.1 Iso10816.....	43
4.10.2.2 RmsAccEnvelope.....	44
4.10.2.3 RmsAccRaw.....	44
4.10.2.4 RmsHighFrequency.....	44
4.10.2.5 RmsRaw.....	45
4.10.2.6 RmsVelEnvelope.....	45
4.10.2.7 RmsVelRaw.....	45
4.10.3 Crest-Faktor.....	46
4.10.3.1 CrestFactorHighFrequency.....	46
4.10.3.2 CrestFactorRaw.....	47
4.10.4 K(t)-Wert.....	48
4.10.4.1 Vdi3832KtHighFrequency.....	48
4.10.4.2 Vdi3832KtRaw.....	49
4.10.5 Kurtosis-Faktor.....	49
4.10.5.1 KurtosisRaw.....	49
4.10.6 Skewness-Faktor.....	50
4.10.6.1 SkewnessRaw.....	50
4.10.7 FrequencyBand.....	50
4.11 Kennwerte (Minimal und Maximal).....	51
4.11.1 CrestFactorRawMax.....	52
4.11.2 CrestFactorRawMin.....	52
4.11.3 Iso10816Max.....	52
4.11.4 Iso10816Min.....	52
4.11.5 KurtosisRawMax.....	52
4.11.6 KurtosisRawMin.....	53
4.11.7 MinMaxCounter.....	53
4.11.8 PeakHighFrequencyMax.....	53
4.11.9 PeakHighFrequencyMin.....	53
4.11.10 PeakRawMax.....	53
4.11.11 PeakRawMin.....	54
4.11.12 RmsAccEnvelopeMax.....	54
4.11.13 RmsAccEnvelopeMin.....	54
4.11.14 RmsAccRawMax.....	54
4.11.15 RmsAccRawMin.....	54
4.11.16 RmsHighFrequencyMax.....	54
4.11.17 RmsHighFrequencyMin.....	55
4.11.18 RmsRawMax.....	55
4.11.19 RmsRawMin.....	55
4.11.20 RmsVelEnvelopeMax.....	55

4.11.21 RmsVelEnvelopeMin.....	55
4.11.22 RmsVelRawMin.....	56
4.11.23 RmsVelRawMax.....	56
4.11.24 SkewnessRawMax.....	56
4.11.25 SkewnessRawMin.....	56
4.12 Frequenzband-Konfigurationregister.....	57
4.12.1 HighFrequencyConfig.....	57
4.12.2 HighFrequencyConfigRead.....	57
4.12.3 MaxFrequencyEnvelope.....	57
4.12.4 MaxFrequencyEnvelopeRead.....	58
4.12.5 MaxFrequencyRaw.....	58
4.12.6 MaxFrequencyRawRead.....	58
4.12.7 MinFrequencyEnvelope.....	59
4.12.8 MinFrequencyEnvelopeRead.....	59
4.12.9 MinFrequencyRaw.....	60
4.12.10 MinFrequencyRawRead.....	60
4.13 Frequenzbänder.....	61
4.13.1 Breitband Effektivwert.....	62
4.13.2 Drehzahlabhängiger Effektivwert.....	63
4.13.3 Rauschen.....	63
4.13.4 Konfiguration.....	64
4.13.5 FrequencyBandMax.....	64
4.13.6 FrequencyBandMin.....	64
4.13.7 FrequencyBandConfig.....	65
4.13.8 FrequencyBandConfigRead.....	65
4.13.9 FrequencyBandDmgFreq60rpm.....	66
4.13.10 FrequencyBandDmgFreq60rpmRead.....	66
4.13.11 FrequencyBandTolerance.....	66
4.13.12 FrequencyBandToleranceRead.....	66
4.13.13 FrequencyBandLowerFrequency.....	66
4.13.14 FrequencyBandLowerFrequencyRead.....	67
4.13.15 FrequencyBandUpperFrequency.....	67
4.13.16 FrequencyBandUpperFrequencyRead.....	67
4.14 Flatstream.....	68
4.14.1 Kennwertübertragung via Flatstream.....	68
4.14.1.1 Register für den Kennwert-Flatstream.....	68
4.14.1.2 Aufbau des Kennwert-Streams.....	69
4.14.2 Pufferupload via Flatstream.....	69
4.14.2.1 Register für den Puffer-Flatstream.....	69
4.14.2.2 Ablauf des Pufferuploads.....	69
4.14.3 Die Forward-Funktion beim X20CM4810.....	70
4.14.4 Die Flatstream-Kommunikation.....	71
4.14.4.1 Einleitung.....	71
4.14.4.2 Nachricht, Segment, Sequenz, MTU.....	72
4.14.4.3 Prinzip des Flatstreams.....	73
4.14.4.4 Die Register für den Flatstream-Modus.....	74
4.14.4.5 Die "Forward"-Funktion am Beispiel des X2X Link.....	94
4.15 Benutzung des Moduls am Feldbus.....	99
4.15.1 Bus Controller mit FeldbusDESIGNER Unterstützung.....	99
4.15.2 Bus Controller ohne FeldbusDESIGNER Unterstützung.....	99
4.15.3 B&R SG4 CPU mit Schnittstellenmodul.....	99
4.15.4 CANIO Bus Controller.....	99
4.16 Zuweisen der Taskklasse.....	99
4.17 Maximale Zykluszeit.....	100
4.18 Minimale Zykluszeit.....	100

5 Condition Monitoring / Schwingungsanalyse.....	101
5.1 Grundlagen.....	101
5.1.1 Was ist Condition Monitoring.....	101
5.1.1.1 Badewannen- bzw. Abnutzungskurve.....	102
5.1.1.2 Schadensentwicklung und Schadenskette.....	103
5.1.2 Klassisches Condition Monitoring.....	104
5.1.3 Condition Monitoring Methoden in der Übersicht.....	104
5.2 Schwingungsmesstechnik.....	105
5.2.1 Sensortechnik.....	105
5.2.1.1 Grundsätzlicher Aufbau.....	105
5.2.1.2 Einflussgrößen auf den Sensor.....	105
5.2.1.3 Montage von Sensoren.....	108
5.2.2 Schwingungen - Übersicht über die Messung des Körperschalls.....	110
5.2.2.1 Schwingungen.....	110
5.2.2.2 Die Fast-Fourier-Transformation (FFT).....	114
5.2.2.3 Hüllkurve - Envelope.....	121
5.2.2.4 Weg - Geschwindigkeit - Beschleunigung.....	124
5.2.3 Grenzwertfindung und Alarmgrenzen.....	124
5.2.3.1 Vergleich mit Referenzen und Normen.....	124
5.2.3.2 Grenzwerte des Herstellers.....	124
5.2.3.3 Grenzwerte des Betreibers - Erfahrungswerte.....	125
5.2.3.4 Beurteilung des Trends.....	125
5.2.3.5 Dynamische Drehzahländerung.....	126
5.3 Praktische Anwendung Schadenserkennung.....	128
5.3.1 Kennwerte.....	129
5.3.2 Fehlermöglichkeiten.....	130
5.3.2.1 Unwucht.....	131
5.3.2.2 Fehlausrichtung.....	133
5.3.2.3 Riemenschäden.....	134
5.3.2.4 Lose oder anschlagende Teile.....	137
5.3.2.5 Gleitlagerschäden.....	138
5.3.2.6 Wälzlagerschäden.....	138
5.3.2.7 Getriebeschäden.....	147
5.3.2.8 Elektrischer Fehler.....	149
5.3.3 Einsatzmöglichkeiten typische Schadenserkennung.....	150
5.3.3.1 Ventilator starr gekoppelt.....	151
5.3.3.2 Ventilator mit Zwischenwelle.....	153
5.3.3.3 Ventilator mit Riemenantrieb.....	155
5.3.3.4 Pumpe direkt gekoppelt.....	158
5.3.3.5 Getriebe.....	160
5.3.3.6 Getriebe mit Zwischenwelle.....	162
5.4 Weiterführende Literatur.....	164
6 Zubehör.....	165
6.1 Sensoren.....	165
6.1.1 0ACS100A.00-1.....	165
6.1.1.1 Bestelldaten.....	165
6.1.1.2 Technische Daten.....	166
6.1.1.3 Abmessungen.....	166
6.1.1.4 Einbaurichtung.....	167
6.1.2 0ACS100A.90-1.....	168
6.1.2.1 Bestelldaten.....	168
6.1.2.2 Technische Daten.....	168
6.1.2.3 Abmessungen.....	169
6.1.2.4 Einbaurichtung.....	169
6.1.3 Allgemeines.....	170
6.1.3.1 Steckerbelegung.....	170

6.1.3.2 Frequenzverhalten.....	170
6.2 Sensorkabel.....	171
6.2.1 Bestelldaten.....	171
6.2.2 Technische Daten.....	171
6.2.3 Sensorkabel mit Stecker M12 Buchse.....	171
6.2.4 Kabelplan.....	172

1 Allgemeines

Das Modul vereint auf kompakter Grundfläche ein System zur Messung und Analyse von Vibrationen, wie sie beispielsweise bei der Zustandsüberwachung von Maschinen und Anlagen Verwendung findet.

Die Messeinheit besteht aus 4 Vibrationseingängen mit einer Wandlerauflösung von 24 Bit bei einer Abtastfrequenz von 51,562 kHz. Jeder der Eingänge verfügt über eine separat zuschaltbare Konstantstromquelle zur Versorgung von IEPE-Sensoren.

Das Modul integriert zahlreiche Algorithmen zur Signalaufbereitung. Diese umfassen u. a. konfigurierbare Hoch- und Tiefpassfilter, Hüllkurvengenerierung sowie Fourier-Transformation.

Für die lückenlose Signalauswertung stehen unterschiedliche Kennwerte und konfigurierbare Frequenzbänder zur Verfügung. Die modulinterne Berechnung sorgt dabei dafür, dass die resultierende Last am Bus und auf der CPU möglichst gering ausfällt.

Egal ob klassische Zustandsüberwachung oder Prozessoptimierung, durch die maximal auswertbare Frequenz von 10 kHz und eine maximale Frequenzauflösung von kleiner 63 mHz findet das Modul in fast allen Anwendungsbereichen Verwendung. Des Weiteren kann das Modul bei Bedarf wie ein gewöhnliches Eingangsmodul verwendet werden.

Für die Archivierung bzw. externe Auswertung der Messwerte bietet das Modul zusätzlich den Upload aller Werte im Zeit- und Frequenzbereich.

Information:

Das Modul X20CM4810 wird erst ab Automation Studio Version 3.0.90.x und Automation Runtime Versionen ≥J3.09, J4.01 und O4.02 unterstützt.

- 4 Kanal Vibrationsmess- und -analysem modul
- 24 Bit Auflösung bei 51,562 kHz
- IEPE-Versorgung
- Umfangreiche Kennwertberechnung im Modul
- Datenpuffer-Upload

1.1 Coated Module

Coated Module sind X20 Module mit einer Schutzbeschichtung der Elektronikbaugruppe. Die Beschichtung schützt X20c Module vor Betauung und Schadgasen.

Die Elektronik der Module ist vollständig funktionskompatibel zu den entsprechenden X20 Modulen.

In diesem Anwenderhandbuch werden zur Vereinfachung nur Bilder und Modulbezeichnungen der unbeschichteten Module verwendet.

Die Beschichtung wurde nach folgenden Normen qualifiziert:

- Betauung: BMW GS 95011-4, 2x 1 Zyklus
- Schadgas: EN 60068-2-60, Methode 4, Exposition 21 Tage



2 Bestelldaten

Bestellnummer	Kurzbeschreibung	Abbildung
	Sonstige Funktionen	
X20CM4810	X20 Analoges Eingangsmodul, zur Schwingungsmessung und -analyse von Condition-Monitoring-Aufgaben, 4 IEPE-Analogeingänge, 51,5625 kHz Abtastfrequenz, 24 Bit Wandlerauflösung	
X20cCM4810	X20 Analoges Eingangsmodul beschichtet, zur Schwingungsmessung und -analyse von Condition-Monitoring-Aufgaben, 4 IEPE-Analogeingänge, 51,5625 kHz Abtastfrequenz, 24 Bit Wandlerauflösung	
	Erforderliches Zubehör	
	Busmodule	
X20BM31	X20 Busmodul, für doppeltbreite Module, 24 VDC codiert, interne I/O-Versorgung durchverbunden	
X20cBM31	X20 Busmodul, beschichtet, für doppeltbreite Module, 24 VDC codiert, interne I/O-Versorgung durchverbunden	
	Feldklemmen	
X20TB12	X20 Feldklemme, 12-polig, 24 VDC codiert	
	Optionales Zubehör	
	Sensoren	
0ACS100A.00-1	Beschleunigungssensor, nominale Empfindlichkeit 100 mV/g, Ausgang oben	
0ACS100A.90-1	Beschleunigungssensor, nominale Empfindlichkeit 100 mV/g, Ausgang seitlich	
	Sensorkabel	
0ACC0020.01-1	Kabel für Beschleunigungssensor, Länge 2 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0050.01-1	Kabel für Beschleunigungssensor, Länge 5 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0100.01-1	Kabel für Beschleunigungssensor, Länge 10 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0150.01-1	Kabel für Beschleunigungssensor, Länge 15 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0200.01-1	Kabel für Beschleunigungssensor, Länge 20 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0500.01-1	Kabel für Beschleunigungssensor, Länge 50 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC1000.01-1	Kabel für Beschleunigungssensor, Länge 100 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	

Tabelle 1: X20CM4810, X20cCM4810 - Bestelldaten

3 Technische Daten

3.1 Technische Daten

Bestellnummer	X20CM4810	X20cCM4810
Kurzbeschreibung		
I/O-Modul	X20 4-Kanal analoges Eingangsmodul zur Schwingungs- messung und -analyse von Condition Monitoring Aufgaben	
Allgemeines		
Isolationsspannung zwischen Kanal und Bus	500 V _{eff}	
Nennspannung	24 VDC ±20%	
B&R ID-Code	0xC8F9	0xE7F0
Statusanzeigen	Run, Error, Vibrationseingänge 1 bis 4	
Diagnose		
Modul Run/Error	Ja, per Status-LED und SW-Status	
Leistungsaufnahme		
Bus	0,01 W	
I/O-intern	2,5 W	
Zulassungen		
CE	Ja	
ATEX	Zone 2, II 3G Ex nA nC IIA T5 Gc IP20, Ta (siehe X20 Anwenderhandbuch) FTZÜ 09 ATEX 0083X	
UL	cULus E115267 Industrial Control Equipment	
HazLoc	cCSAus 244665 Process Control Equipment for Hazardous Locations Class I, Division 2, Groups ABCD, T5	
EAC	Ja	
Analoge Eingänge		
Anzahl	4	
Eingangsart	IEPE-Sensor: Beschleunigung	
Digitale Wandlerauflösung	24 Bit	
Drahtbruchererkennung		
pro Kanal	Ja	
bei min. Versorgungsspannung ¹⁾	ab 17 V (oder größer) für mehr als 1 ms	
bei nom. Versorgungsspannung ²⁾	ab 21,3 V (oder größer) für mehr als 1 ms	
bei max. Versorgungsspannung ³⁾	ab 25,5 V (oder größer) für mehr als 1 ms	
Zulässiges Eingangssignal	±10 VAC	
Wandlungsverfahren	Sigma Delta	
Typ	Vibrationseingang	
Abtastfrequenz	51,5625 kHz	
Eingangshochpasseckfrequenz	34 mHz	
Eingangstiefpasseckfrequenz	19,75 kHz	
Downsampling	200 Hz, 500 Hz, 1 kHz, 2 kHz, 5 kHz, 10 kHz (konfigurierbar)	
Frequenzauflösung der Spektren	0,0629 Hz, 0,1574 Hz, 0,3147 Hz, 0,6294 Hz, 1,5736 Hz, 3,1471 Hz	
Sensorversorgung	IEPE, 5 mA Konstantstromquelle (4,9 bis 5,5 mA), pro Kanal abschaltbar	
Elektrische Eigenschaften		
Potenzialtrennung	Kanal zu Bus getrennt Kanal zu Kanal nicht getrennt	
Einsatzbedingungen		
Einbaulage		
waagrecht	Ja	
senkrecht	Ja	
Aufstellungshöhe über NN (Meeresspiegel)		
0 bis 2000 m	Keine Einschränkung	
>2000 m	Reduktion der Umgebungstemperatur um 0,5°C pro 100 m	
Schutzart nach EN 60529	IP20	
Umgebungsbedingungen		
Temperatur		
Betrieb		
waagrechte Einbaulage	-25 bis 50°C	
senkrechte Einbaulage	-25 bis 45°C	
Derating	Siehe Abschnitt "Derating"	
Lagerung	-40 bis 85°C	
Transport	-40 bis 85°C	

Tabelle 2: X20CM4810, X20cCM4810 - Technische Daten

Bestellnummer	X20CM4810	X20cCM4810
Luftfeuchtigkeit		
Betrieb	5 bis 95%, nicht kondensierend	Bis 100%, kondensierend
Lagerung	5 bis 95%, nicht kondensierend	
Transport	5 bis 95%, nicht kondensierend	
Mechanische Eigenschaften		
Anmerkung	Feldklemme 1x X20TB12 gesondert bestellen Busmodul 1x X20BM31 gesondert bestellen	Feldklemme 1x X20TB12 gesondert bestellen Busmodul 1x X20cBM31 gesondert bestellen
Rastermaß	25 ^{+0,2} mm	

Tabelle 2: X20CM4810, X20cCM4810 - Technische Daten

- 1) Eingangsspannung: 19,2 V
- 2) Eingangsspannung: 24 V
- 3) Eingangsspannung: 28,8 V

3.2 Status-LEDs

Für die Beschreibung der verschiedenen Betriebsmodi siehe X20 System Anwenderhandbuch, Abschnitt "Zusätzliche Informationen - Diagnose-LEDs".

Abbildung	LED	Farbe	Status	Beschreibung
	r	Grün	Aus	Modul nicht versorgt
			Single Flash	Modus RESET
			Double Flash	Modus BOOT (während Firmware-Update) ¹⁾
			Blinkend	Modus PREOPERATIONAL
			Ein	Modus RUN
	e	Rot	Aus	Modul nicht versorgt oder alles in Ordnung
			Ein	Warnung, Fehler oder Reset-Status
	e + r	Rot ein und grüner Single Flash		Firmware ist ungültig
	1 - 4	Grün	Ein	Status des entsprechenden Beschleunigungssensors (kein Drahtbruch)

- 1) Je nach Konfiguration kann ein [Firmware-Update](#) bis zu mehreren Minuten benötigen.

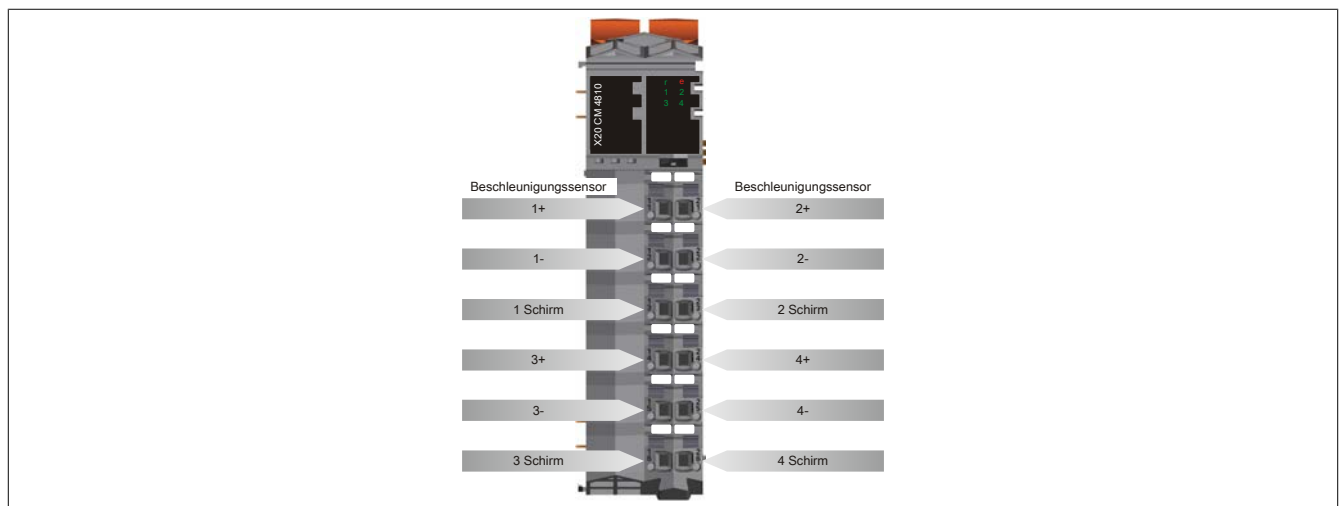
3.3 Firmware-Updatezeit

Aufgrund des großen Firmwareumfangs nimmt das Firmware-Update einige Zeit in Anspruch.

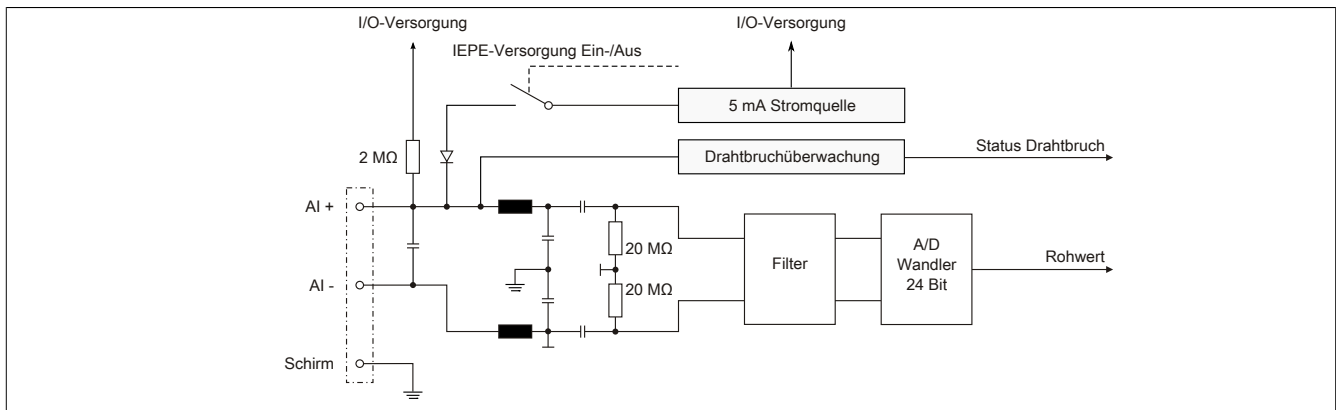
Abhängig von der eingestellten Buszykluszeit ergeben sich z. B. folgende Updatezeiten:

Buszykluszeit	Updatezeit
400 µs	ca. 3 min
2 ms	ca. 15 min

3.4 Anschlussbelegung



3.5 Eingangsschema



3.6 Schirmung

Es gibt 2 Möglichkeiten, um die Kabel für die Beschleunigungssensoren zu schirmen:

- Schirmungsmöglichkeit auf der Feldklemme. (Kontakt "B.-Sensor 1 bis 4 Schirm")
- X20 Schirmung auf dem Busmodul. (Siehe dazu X20 System Anwenderhandbuch, Abschnitt "Schirmung")

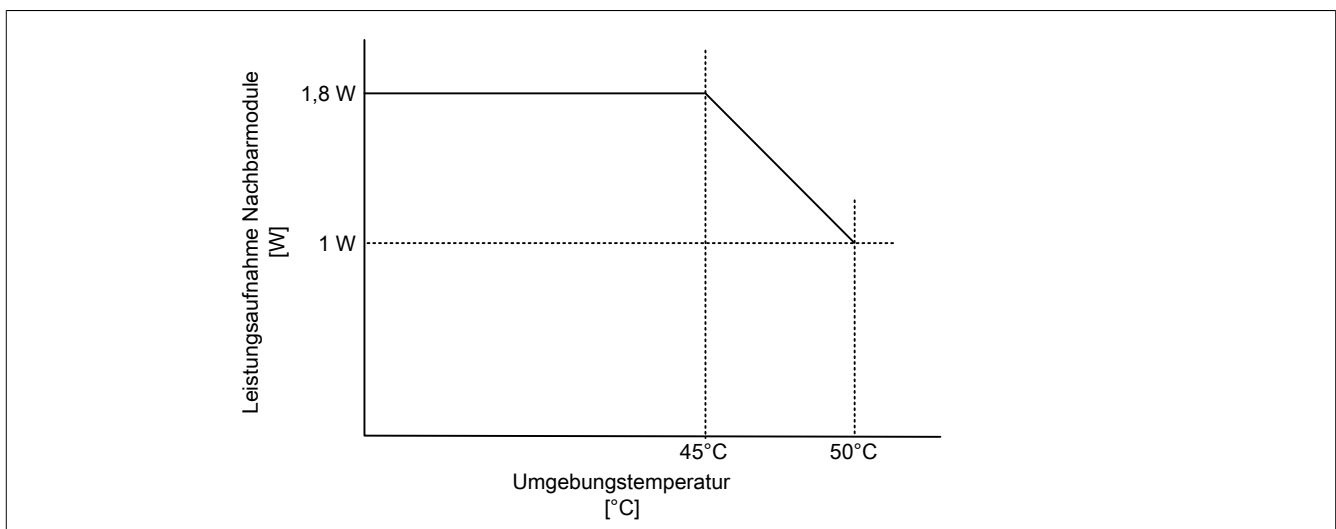
Für EMV-unkritische Umgebungsbedingungen reicht die Schirmung durch die Feldklemme völlig aus.

Wird das Modul jedoch bei EMV-kritischen Umgebungen mit hochfrequenten Störungen betrieben, dann sollten beide Schirmungsmöglichkeiten gleichzeitig verwendet werden.

3.7 Derating

Waagrechte Einbaulage

Für Umgebungstemperaturen größer 45°C ist beim X20CM4810 ein Leistungsderating einzuhalten:



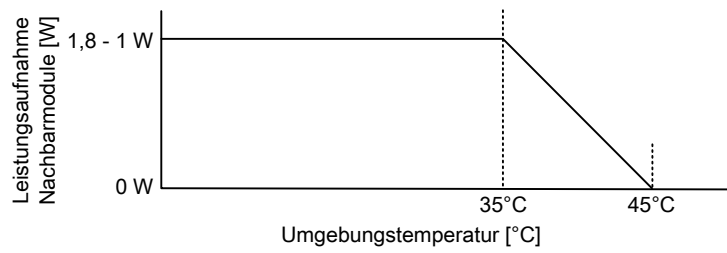
Wenn das X20CM4810 bis 50°C Umgebungstemperatur betrieben werden soll, dann dürfen die Nachbarmodule nicht mehr als 1 W Leistung aufnehmen.

Ein Beispiel zur Berechnung der Verlustleistung von I/O-Modulen ist im X20 Anwenderhandbuch, Abschnitt "Mechanische und elektrische Konfiguration - Verlustleistung von I/O-Modulen" zu finden.

Beispiele für Umgebungstemperatur bis 50°C

Senkrechte Einbaulage

Für Umgebungstemperaturen größer 35°C ist beim X20CM4810 ein Leistungsderating einzuhalten:



Wenn das X20CM4810 bis 45°C Umgebungstemperatur betrieben werden soll, dann dürfen die Nachbarmodule keine Leistung aufnehmen.

Beispiel für Umgebungstemperatur bis 45°C:

X20CM4810	Keine Leistungsaufnahme z. B. X20ZF0000	X20CM4810	Keine Leistungsaufnahme z. B. X20ZF0000	X20CM4810	Keine Leistungsaufnahme z. B. X20ZF0000
-----------	--	-----------	--	-----------	--

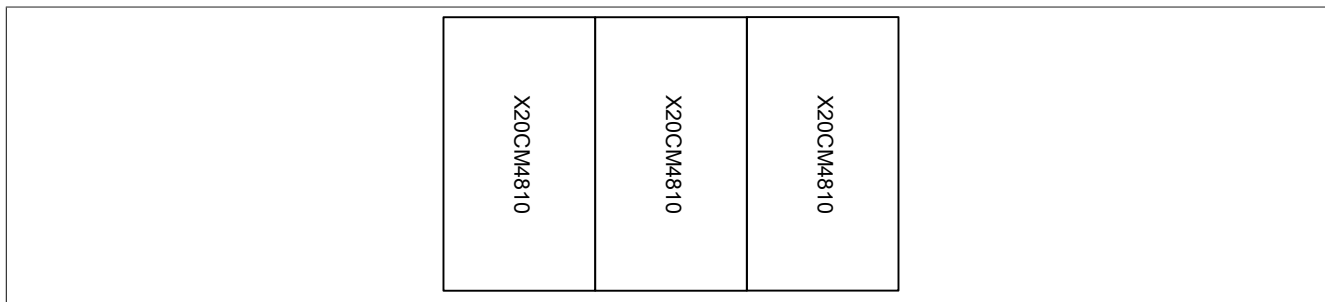
Beispiele für Umgebungstemperatur bis 35°C:

Leistungsaufnahme 1,8 W	Leistungsaufnahme 1,8 W	Leistungsaufnahme 1,8 W	Leistungsaufnahme 1,8 W	X20CM4810	Leistungsaufnahme 1,8 W	Leistungsaufnahme 1,8 W	Leistungsaufnahme 1,8 W
-------------------------	-------------------------	-------------------------	-------------------------	-----------	-------------------------	-------------------------	-------------------------

X20CM4810	Leistungsaufnahme 1,8 W	Leistungsaufnahme 1,8 W	Leistungsaufnahme 1,8 W
-----------	-------------------------	-------------------------	-------------------------

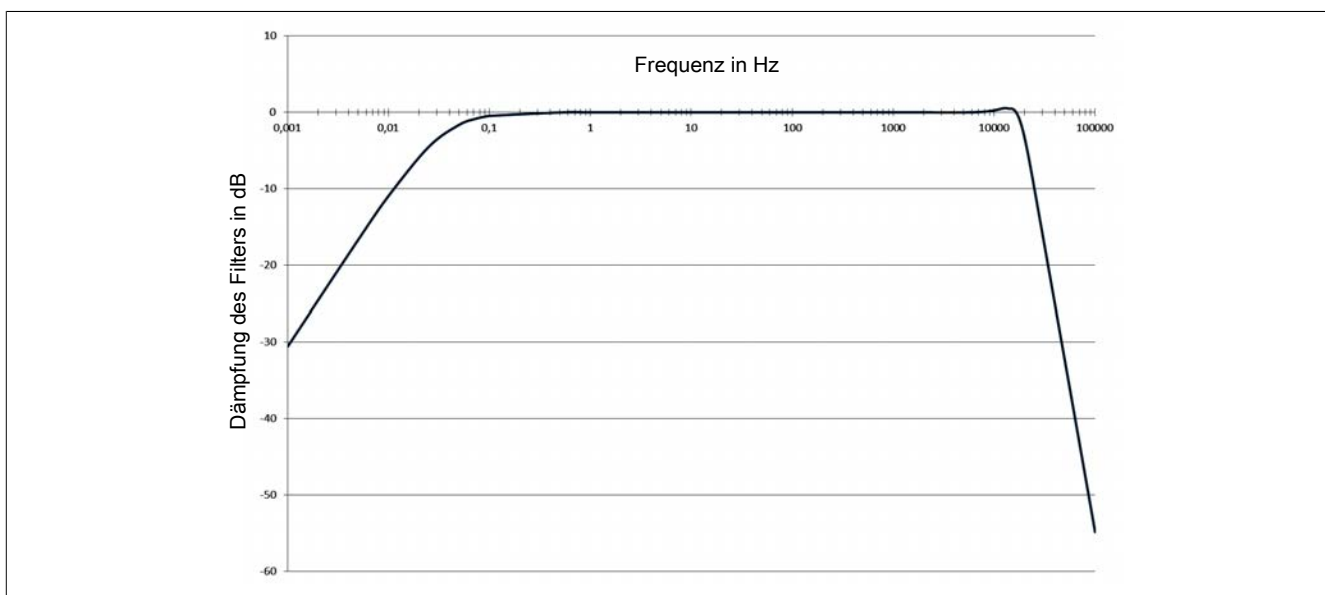
Leistungsaufnahme 1,8 W	X20CM4810	X20 Zentraleinheit z. B. X20CP1486
-------------------------	-----------	---------------------------------------

X20 doppelt breite Module sind speziell zu behandeln. So ist z. B. ein Betrieb von mehrere X20CM4810 nebeneinander bis zu einer Umgebungstemperatur von 30°C möglich.



3.8 Gainverlauf

Die nachfolgende Grafik zeigt den typischen Gainverlauf des Moduls.



3.9 Einschwingzeit

Aufgrund des Eingangshochpasses des Wechselspannungseingangs (Grenzfrequenz von 34 mHz) benötigt dieser nach Änderung des DC-Offsets des anliegenden Signals eine bestimmte Einschwingzeit.

- Einschwingzeit bei 24 V Versorgungsspannung auf eine Genauigkeit von 0,4 g nach ca. 30 Sekunden bei 100 mV/g Sensorempfindlichkeit.
- Einschwingzeit bei 24 V Versorgungsspannung auf eine Genauigkeit von 0,001 g nach ca. 60 Sekunden bei 100 mV/g Sensorempfindlichkeit.
- Um nach einem Drahtbruch ein genaues Messergebnis zu erhalten muss die entsprechende Einschwingzeit abgewartet werden. Deswegen werden vom Modul während den ersten 30 Sekunden nach einem Neustart oder Drahtbruch alle Kennwerte und analoge Eingangswerte auf 0 gesetzt.

3.10 Sensorauflösung

Das Modul rechnet immer mit einem 100 mV/g Beschleunigungssensor am Eingang. Bei Verwendung des [Funktionsmodell 0 - Standard](#) kann über das Register "[SensitivitySensor](#)" auf [Seite 40](#) eine andere Sensorauflösung eingestellt werden.

Wird ein anderes Funktionsmodell verwendet (z. B. SGC oder Bus Controller) so muss die Umrechnung auf eine andere Auflösung manuell erfolgen.

Beispiel

Faktor = $100 / (\text{Sensorauflösung in mV/g})$

Alle Werte müssen mit dem errechneten Faktor multipliziert werden. Dies gilt auch für die AnalogInput-Kennwerte, wenn die Kennwertberechnung eingeschaltet ist und für hochgeladene Zeitsignale und Amplitudenspektren. Ausgenommen sind alle einheitenlosen Kennwerte wie z. B. "[KurtosisRaw](#)" auf [Seite 49](#), "[CrestFactorRaw](#)" auf [Seite 47](#) und "[SkewnessRaw](#)" auf [Seite 50](#).

3.11 Verwendung einer B&R Compact CPU bzw. Feldbus CPU

Aufgrund der großen Firmware des Moduls werden nur CPUs mit ausreichend ROM Speicher (größer 1MB) unterstützt. (X20CP0292 bzw. X20XC0292)

4 Registerbeschreibung

4.1 Allgemeine Datenpunkte

Neben den in der Registerbeschreibung beschriebenen Registern verfügt das Modul über zusätzliche allgemeine Datenpunkte. Diese sind nicht modulspezifisch, sondern enthalten allgemeine Informationen wie z. B. Seriennummer und Hardware-Variante.

Die allgemeinen Datenpunkte sind im X20 System Anwenderhandbuch, Abschnitt "Zusätzliche Informationen - Allgemeine Datenpunkte" beschrieben.

4.2 Funktionsmodell 0 - Standard

Dies ist das Standard Funktionsmodell des Moduls. Die berechneten Kennwerte des Moduls werden über den Flatstream alle 300 ms vom Modul gestreamt und vom Automation Runtime für den Anwender aufbereitet. Wenn die gestreamten Daten nicht bis zur nächsten Übertragung abgeholt werden, gehen die Kennwerte der nächsten Messung verloren. Daher ist das Einhalten der maximalen Zykluszeit für eine fehlerfreie Auswertung zu beachten.

Die analogen Eingänge sind als zyklische Datenpunkte vorhanden.

Um den Anwender zu unterstützen, werden in diesem Funktionsmodell alle Kennwerte vom Modul im Automation Runtime aufbereitet, wie z. B. Kennwert Flatstream-Handling, Einheitskalierung usw., und dem Anwender zur Verfügung gestellt. Siehe "[Automation Runtime Unterstützung](#)" auf Seite 37

Mit diesem Funktionsmodell ist es ebenso möglich, über einen weiteren Flatstream Datenpuffer vom Modul hochzuladen. Zum Hochladen der Puffer vom Modul steht die Bibliothek **AsIOVib** zur Verfügung. Für eine Beschreibung der Bibliothek siehe "Automation Help - Programmierung - Bibliotheken - Direkter I/O Zugriff - AsIOVIB".

In diesem Funktionsmodell kann das Modul nur über die I/O-Konfiguration konfiguriert werden. Es dürfen azyklisch keine Register umkonfiguriert werden.

Register	Name	Datentyp	Lesen		Schreiben	
			Zyklisch	Azyklisch	Zyklisch	Azyklisch
Modul – Konfiguration						
-	Zykluszeit	-				
Allgemeine Register						
1310	AutogainDelay01	UINT				•
526	AutogainDelay01Read	UINT		•		
0	Control01	UINT			•	
514	SensorConfig01	UINT				•
	SensorConfig01Read			•		
0	Status01	UINT	•			
AnalogInput - Funktionen						
2 * N	AnalogInput0N (Index N = 1 bis 4)	INT	•			
1330	AnalogInputConfig01	UINT				•
570	AnalogInputConfig01Read	UINT		•		
2	AnalogInputControlByte01	UINT			•	
22 + N* 4	AnalogInputSamples0N (Index N = 1 bis 4)	UINT			•	
1298	AnalogInputScale01	UINT				•
546	AnalogInputScale01Read	UINT		•		
1310 + N*4	SamplesAnalogInput0N (Index N = 1 bis 4)	UINT				•
526 + N* 4	SamplesAnalogInput0NRead (Index N = 1 bis 4)	UINT		•		
Zusätzliche von Automation Runtime generierte Register und Kennwerte						
-	ActSpeed0N (Index N = 1 bis 4)	REAL			•	
-	CrestFactorHighFrequency0N (Index N = 1 bis 4)	REAL	•			
-	DataConsistentWithLockedBuffers0N (Index N = 1 bis 4)	BOOL	•			
-	DataToggleBit01	BOOL	•			
-	OverflowAnalogInput0N (Index N = 1 bis 4)	BOOL	•			
-	OverflowCharacteristicValues0N (Index N = 1 bis 4)	UINT	•			
-	OverflowFrequencyBands01	UDINT	•			
-	PeakHighFrequencyRef0N (Index N = 1 bis 4)	REAL			•	
-	PeakHighFrequencyRefCalculated0N (Index N = 1 bis 4)	REAL	•			
-	PeakRawRef0N (Index N = 1 bis 4)	REAL			•	
-	PeakRawRefCalculated0N (Index N = 1 bis 4)	REAL	•			
-	RmsHighFrequencyRef0N (Index N = 1 bis 4)	REAL			•	
-	RmsHighFrequencyRefCalculated0N (Index N = 1 bis 4)	REAL	•			
-	RmsRawRef0N (Index N = 1 bis 4)	REAL			•	

Register	Name	Datentyp	Lesen		Schreiben	
			Zyklisch	Azyklisch	Zyklisch	Azyklisch
-	RmsRawRefCalculated0N (Index N = 1 bis 4)	REAL	•			
-	SensitivitySensor0N (Index N = 1 bis 4)	REAL			•	
-	Vdi3832KtHighFrequency0N (Index N = 1 bis 4)	REAL	•			
-	Vdi3832KtRaw0N (Index N = 1 bis 4)	REAL	•			
Kennwerte (im Flatstream übertragen)						
-	CrestFactorRaw0N (Index N = 1 bis 4)	REAL	•			
-	Iso10816_0N (Index N = 1 bis 4)	REAL	•			
-	KurtosisRaw0N (Index N = 1 bis 4)	REAL	•			
-	PeakHighFrequency0N (Index N = 1 bis 4)	REAL	•			
-	PeakRaw0N (Index N = 1 bis 4)	REAL	•			
-	RmsAccEnvelope0N (Index N = 1 bis 4)	REAL	•			
-	RmsAccRaw0N (Index N = 1 bis 4)	REAL	•			
-	RmsHighFrequency0N (Index N = 1 bis 4)	REAL	•			
-	RmsRaw0N (Index N = 1 bis 4)	REAL	•			
-	RmsVelEnvelope0N (Index N = 1 bis 4)	REAL	•			
-	RmsVelRaw0N (Index N = 1 bis 4)	REAL	•			
-	SkewnessRaw0N (Index N = 1 bis 4)	REAL	•			
-	FrequencyBandN (Index N = 1 bis 32)	REAL	•			
Kennwerte Minimal- und Maximalwerte						
2690	MinMaxCounter01	UINT		•		
3588 + N*8	CrestFactorRawMax0N (Index N = 1 bis 4)	UDINT		•		
2948 + N*8	CrestFactorRawMin0N (Index N = 1 bis 4)	UDINT		•		
3332 + N*8	Iso10816Max0N (Index N = 1 bis 4)	UDINT		•		
2692 + N*8	Iso10816Min0N (Index N = 1 bis 4)	UDINT		•		
3556 + N*8	KurtosisRawMax0N (Index N = 1 bis 4)	DINT		•		
2916 + N*8	KurtosisRawMin0N (Index N = 1 bis 4)	DINT		•		
3492 + N*8	PeakHighFrequencyMax0N (Index N = 1 bis 4)	UDINT		•		
2852 + N*8	PeakHighFrequencyMin0N (Index N = 1 bis 4)	UDINT		•		
3684 + N*8	PeakRawMax0N (Index N = 1 bis 4)	UDINT		•		
3044 + N*8	PeakRawMin0N (Index N = 1 bis 4)	UDINT		•		
3428 + N*8	RmsAccEnvelopeMax0N (Index N = 1 bis 4)	UDINT		•		
2788 + N*8	RmsAccEnvelopeMin0N (Index N = 1 bis 4)	UDINT		•		
3364 + N*8	RmsAccRawMax0N (Index N = 1 bis 4)	UDINT		•		
2724 + N*8	RmsAccRawMin0N (Index N = 1 bis 4)	UDINT		•		
3524 + N*8	RmsHighFrequencyMax0N (Index N = 1 bis 4)	UDINT		•		
2884 + N*8	RmsHighFrequencyMin0N (Index N = 1 bis 4)	UDINT		•		
3652 + N*8	RmsRawMax0N (Index N = 1 bis 4)	UDINT		•		
3012 + N*8	RmsRawMin0N (Index N = 1 bis 4)	UDINT		•		
3460 + N*8	RmsVelEnvelopeMax0N (Index N = 1 bis 4)	UDINT		•		
2820 + N*8	RmsVelEnvelopeMin0N (Index N = 1 bis 4)	UDINT		•		
3396 + N*8	RmsVelRawMax0N (Index N = 1 bis 4)	UDINT		•		
2756 + N*8	RmsVelRawMin0N (Index N = 1 bis 4)	UDINT		•		
3620 + N*8	SkewnessRawMax0N (Index N = 1 bis 4)	DINT		•		
2980 + N*8	SkewnessRawMin0N (Index N = 1 bis 4)	DINT		•		
Frequenzkonfiguration						
1302	HighFrequencyConfig01	UINT				•
550	HighFrequencyConfig01Read	UINT		•		
1306	MaxFrequencyEnvelope01	UINT				•
558	MaxFrequencyEnvelope01Read	UINT		•		
526	MaxFrequencyRaw01	UINT				•
554	MaxFrequencyRaw01Read	UINT		•		
522	MinFrequencyEnvelope01	UINT				•
566	MinFrequencyEnvelope01Read	UINT		•		
518	MinFrequencyRaw01	UINT				•
562	MinFrequencyRaw01Read	UINT		•		
Frequenzbänder						
3716 + N*8	FrequencyBandMaxN (Index N = 01 bis 32)	UDINT		•		
3076 + N*8	FrequencyBandMinN (Index N = 01 bis 32)	UDINT		•		
506 + N*24	FrequencyBandNConfig (Index N = 01 bis 32)	UINT				•
1194 + N*24	FrequencyBandNConfigRead (Index N = 01 bis 32)	UINT		•		
514 + N*24	FrequencyBandNDmgFreq60rpm (Index N = 01 bis 32)	UINT				•
1202 + N*24	FrequencyBandNDmgFreq60rpmRead (Index N = 01 bis 32)	UINT		•		
522 + N*24	FrequencyBandNLowerFrequency (Index N = 01 bis 32)	UINT				•
1210 + N*24	FrequencyBandNLowerFrequencyRead (Index N = 01 bis 32)	UINT		•		
518 + N*24	FrequencyBandNTolerance (Index N = 01 bis 32)	UINT				•
1206 + N*24	FrequencyBandNToleranceRead (Index N = 01 bis 32)	UINT		•		
526 + N*24	FrequencyBandNUpperFrequency (Index N = 01 bis 32)	UINT				•
1214 + N*24	FrequencyBandNUpperFrequencyRead (Index N = 01 bis 32)	UINT		•		
Flatstream						
2311	BufferForward01	USINT				•
2318	BufferForwardDelay01	UINT				•
2368	BufferInputSequence01	USINT	•			

Registerbeschreibung

Register	Name	Datentyp	Lesen		Schreiben	
			Zyklisch	Azyklisch	Zyklisch	Azyklisch
2400	BufferOutputSequence01	USINT			•	
2368 + N	BufferRxByte0N (Index N = 1 bis 5)	USINT	•			
2400 + N	BufferTxByte0N (Index N = 1 bis 4)	USINT			•	
263	ParameterForward01	USINT				•

4.3 Funktionsmodell 1 - Schneller Master

Die berechneten Kennwerte des Moduls werden über den Flatstream alle 300 ms zum Master gestreamt. Wenn die gestreamten Daten nicht bis zur nächsten Übertragung abgeholt wurden gehen die Kennwerte der nächsten Messung verloren. Daher ist das Einhalten der maximalen Zykluszeit für eine fehlerfreie Auswertung zu beachten.

Weiters ist es mit diesem Funktionsmodell möglich, über einen weiteren Flatstream Datenpuffer vom Modul hochzuladen.

Die analogen Eingänge sind als zyklische Datenpunkte vorhanden.

Dieses Funktionsmodell kann nur auf Ethernet basierenden Mastern und SGC- bzw. Feldbus-CPU's verwendet werden. Es muss allerdings darauf geachtet werden, dass das Flatstream-Handling im Master implementiert werden muss und dass das Modul die Daten im Flatstream jeden X2X-Zyklus ändert.

Register	Name	Datentyp	Lesen		Schreiben	
			Zyklisch	Azyklisch	Zyklisch	Azyklisch
Modul - Konfiguration						
-	Zykluszeit	-				
Allgemeine Register						
2 + N*2	ActSpeed0N (Index N = 1 bis 4)	UINT			•	
1310	AutogainDelay01	UINT				•
526	AutogainDelay01Read	UINT		•		
0	Control01	UINT			•	
514	SensorConfig01	UINT				•
	SensorConfig01Read			•		
0	Status01	UINT	•			
AnalogInput - Funktionen						
2 * N	AnalogInput0N (Index N = 1 bis 4)	INT	•			
1330	AnalogInputConfig01	UINT				•
570	AnalogInputConfig01Read	UINT		•		
2	AnalogInputControlByte01	UINT			•	
22 + N*4	AnalogInputSamples0N (Index N = 1 bis 4)	UINT			•	
1298	AnalogInputScale01	UINT				•
546	AnalogInputScale01Read	UINT		•		
1310 + N*4	SamplesAnalogInput0N (Index N = 1 bis 4)	UINT				•
526 + N*4	SamplesAnalogInput0NRead (Index N = 1 bis 4)	UINT		•		
Kennwerte (im Flatstream übertragen)						
-	CrestFactorRaw0N (Index N = 1 bis 4)	1)	•			
-	Iso10816_0N (Index N = 1 bis 4)	1)	•			
-	KurtosisRaw0N (Index N = 1 bis 4)	1)	•			
-	PeakHighFrequency0N (Index N = 1 bis 4)	1)	•			
-	PeakRaw0N (Index N = 1 bis 4)	1)	•			
-	RmsAccEnvelope0N (Index N = 1 bis 4)	1)	•			
-	RmsAccRaw0N (Index N = 1 bis 4)	1)	•			
-	RmsHighFrequency0N (Index N = 1 bis 4)	1)	•			
-	RmsRaw0N (Index N = 1 bis 4)	1)	•			
-	RmsVelEnvelope0N (Index N = 1 bis 4)	1)	•			
-	RmsVelRaw0N (Index N = 1 bis 4)	1)	•			
-	SkewnessRaw0N (Index N = 1 bis 4)	1)	•			
-	FrequencyBandN (Index N = 1 bis 32)	1)	•			
Kennwerte Minimal- und Maximalwerte						
2690	MinMaxCounter01	UINT		•		
3588 + N*8	CrestFactorRawMax0N (Index N = 1 bis 4)	UDINT		•		
2948 + N*8	CrestFactorRawMin0N (Index N = 1 bis 4)	UDINT		•		
3332 + N*8	Iso10816Max0N (Index N = 1 bis 4)	UDINT		•		
2692 + N*8	Iso10816Min0N (Index N = 1 bis 4)	UDINT		•		
3556 + N*8	KurtosisRawMax0N (Index N = 1 bis 4)	DINT		•		
2916 + N*8	KurtosisRawMin0N (Index N = 1 bis 4)	DINT		•		
3492 + N*8	PeakHighFrequencyMax0N (Index N = 1 bis 4)	UDINT		•		
2852 + N*8	PeakHighFrequencyMin0N (Index N = 1 bis 4)	UDINT		•		
3684 + N*8	PeakRawMax0N (Index N = 1 bis 4)	UDINT		•		
3044 + N*8	PeakRawMin0N (Index N = 1 bis 4)	UDINT		•		
3428 + N*8	RmsAccEnvelopeMax0N (Index N = 1 bis 4)	UDINT		•		
2788 + N*8	RmsAccEnvelopeMin0N (Index N = 1 bis 4)	UDINT		•		
3364 + N*8	RmsAccRawMax0N (Index N = 1 bis 4)	UDINT		•		
2724 + N*8	RmsAccRawMin0N (Index N = 1 bis 4)	UDINT		•		
3524 + N*8	RmsHighFrequencyMax0N (Index N = 1 bis 4)	UDINT		•		
2884 + N*8	RmsHighFrequencyMin0N (Index N = 1 bis 4)	UDINT		•		
3652 + N*8	RmsRawMax0N (Index N = 1 bis 4)	UDINT		•		
3012 + N*8	RmsRawMin0N (Index N = 1 bis 4)	UDINT		•		
3460 + N*8	RmsVelEnvelopeMax0N (Index N = 1 bis 4)	UDINT		•		
2820 + N*8	RmsVelEnvelopeMin0N (Index N = 1 bis 4)	UDINT		•		
3396 + N*8	RmsVelRawMax0N (Index N = 1 bis 4)	UDINT		•		
2756 + N*8	RmsVelRawMin0N (Index N = 1 bis 4)	UDINT		•		

Registerbeschreibung

Register	Name	Datentyp	Lesen		Schreiben	
			Zyklisch	Azyklisch	Zyklisch	Azyklisch
3620 + N*8	SkewnessRawMax0N (Index N = 1 bis 4)	DINT		•		
2980 + N*8	SkewnessRawMin0N (Index N = 1 bis 4)	DINT		•		
Frequenzkonfiguration						
1302	HighFrequencyConfig01	UINT				•
550	HighFrequencyConfig01Read	UINT		•		
1306	MaxFrequencyEnvelope01	UINT				•
558	MaxFrequencyEnvelope01Read	UINT		•		
526	MaxFrequencyRaw01	UINT				•
554	MaxFrequencyRaw01Read	UINT		•		
522	MinFrequencyEnvelope01	UINT				•
566	MinFrequencyEnvelope01Read	UINT		•		
518	MinFrequencyRaw01	UINT				•
562	MinFrequencyRaw01Read	UINT		•		
Frequenzbänder						
3716 + N*8	FrequencyBandMaxN (Index N = 01 bis 32)	UDINT		•		
3076 + N*8	FrequencyBandMinN (Index N = 01 bis 32)	UDINT		•		
506 + N*24	FrequencyBandNConfig (Index N = 01 bis 32)	UINT				•
1194 + N*24	FrequencyBandNConfigRead (Index N = 01 bis 32)	UINT		•		
514 + N*24	FrequencyBandNDmgFreq60rpm (Index N = 01 bis 32)	UINT				•
1202 + N*24	FrequencyBandNDmgFreq60rpmRead (Index N = 01 bis 32)	UINT		•		
522 + N*24	FrequencyBandNLowerFrequency (Index N = 01 bis 32)	UINT				•
1210 + N*24	FrequencyBandNLowerFrequencyRead (Index N = 01 bis 32)	UINT		•		
518 + N*24	FrequencyBandNTolerance (Index N = 01 bis 32)	UINT				•
1206 + N*24	FrequencyBandNToleranceRead (Index N = 01 bis 32)	UINT		•		
526 + N*24	FrequencyBandNUpperFrequency (Index N = 01 bis 32)	UINT				•
1214 + N*24	FrequencyBandNUpperFrequencyRead (Index N = 01 bis 32)	UINT		•		
Flatsream						
2311	BufferForward01	USINT				•
2318	BufferForwardDelay01	UINT				•
2368	BufferInputSequence01	USINT	•			
2400	BufferOutputSequence01	USINT			•	
2368 + N	BufferRxByte0N (Index N = 1 bis 5)	USINT	•			
2400 + N	BufferTxByte0N (Index N = 1 bis 4)	USINT			•	
263	ParameterForward01	USINT				•
270	ParameterForwardDelay01	INT				•
320	ParameterInputSequence01	USINT	•			
352	ParameterOutputSequence01	USINT			•	
320 + N	ParameterRxByteN (Index N = 1 bis 13)	USINT	•			

1) Für Informationen zum Datenformat siehe "Aufbau des Kennwert-Streams" auf Seite 69.

4.4 Funktionsmodell 2 - Langsamer Master

Dieses Funktionsmodell wurde speziell für die Verwendung des Moduls mit "Langsamen Master" und zum Einsparen von Ressourcen in der SPS konzipiert.

Mit diesem Funktionsmodell ist kein Hochladen von Datenpuffern am Modul möglich.

Die analogen Eingänge sind als zyklische Datenpunkte vorhanden. Die Skalierung der Kennwerte muss manuell erfolgen.

Die vom Modul alle 300 ms berechneten Kennwerte können nur über azyklische Zugriffe ausgelesen werden. Um alle Kennwerte konsistent zueinander zu halten ist es möglich, die Kennwerte während des Auslesens einzufrieren. Eine lückenlose Messung ist mit diesem Funktionsmodell allerdings nicht möglich. Zur lückenlosen Erfassung kann jedoch die Min- und Max-Funktionalität verwendet werden. Siehe ["Kennwerte \(Minimal und Maximal\)" auf Seite 51](#)

Dieses Funktionsmodell wird für alle langsamen Busse und Master empfohlen. Zu beachten ist hierbei noch, dass der azyklische Registerzugriff im Master implementiert werden muss, sofern kein B&R Master verwendet wird.

Register	Name	Datentyp	Lesen		Schreiben	
			Zyklisch	Azyklisch	Zyklisch	Azyklisch
Allgemeine Register						
2 + N*2	ActSpeed0N (Index N = 1 bis 4)	UINT			•	
1310	AutogainDelay01	UINT				•
526	AutogainDelay01Read	UINT		•		
0	Control01	UINT			•	
514	SensorConfig01	UINT		•		•
	SensorConfig01Read					
0	Status01	UINT	•			
AnalogInput - Funktionen						
2 * N	AnalogInput0N (Index N = 1 bis 4)	INT	•			
1330	AnalogInputConfig01	UINT				•
570	AnalogInputConfig01Read	UINT		•		
2	AnalogInputControlByte01	UINT			•	
22 + N*4	AnalogInputSamples0N (Index N = 1 bis 4)	UINT			•	
1298	AnalogInputScale01	UINT				•
546	AnalogInputScale01Read	UINT		•		
1310 + N*4	SamplesAnalogInput0N (Index N = 1 bis 4)	UINT				•
526 + N*4	SamplesAnalogInput0NRead (Index N = 1 bis 4)	UINT		•		
Kennwerte						
828 + N*8	CrestFactorRaw0N (Index N = 1 bis 4)	UDINT		•		
572 + N*8	Iso10816_0N (Index N = 1 bis 4)	UDINT		•		
796 + N*8	KurtosisRaw0N (Index N = 1 bis 4)	DINT		•		
732 + N*8	PeakHighFrequency0N (Index N = 1 bis 4)	UDINT		•		
924 + N*8	PeakRaw0N (Index N = 1 bis 4)	UDINT		•		
668 + N*8	RmsAccEnvelope0N (Index N = 1 bis 4)	UDINT		•		
604 + N*8	RmsAccRaw0N (Index N = 1 bis 4)	UDINT		•		
764 + N*8	RmsHighFrequency0N (Index N = 1 bis 4)	UDINT		•		
982 + N*8	RmsRaw0N (Index N = 1 bis 4)	UDINT		•		
700 + N*8	RmsVelEnvelope0N (Index N = 1 bis 4)	UDINT		•		
636 + N*8	RmsVelRaw0N (Index N = 1 bis 4)	UDINT		•		
860 + N*8	SkewnessRaw0N (Index N = 1 bis 4)	DINT		•		
Kennwerte Minimal- und Maximalwerte						
2690	MinMaxCounter01	UINT		•		
3588 + N*8	CrestFactorRawMax0N (Index N = 1 bis 4)	UDINT		•		
2948 + N*8	CrestFactorRawMin0N (Index N = 1 bis 4)	UDINT		•		
3332 + N*8	Iso10816Max0N (Index N = 1 bis 4)	UDINT		•		
2692 + N*8	Iso10816Min0N (Index N = 1 bis 4)	UDINT		•		
3556 + N*8	KurtosisRawMax0N (Index N = 1 bis 4)	DINT		•		
2916 + N*8	KurtosisRawMin0N (Index N = 1 bis 4)	DINT		•		
3492 + N*8	PeakHighFrequencyMax0N (Index N = 1 bis 4)	UDINT		•		
2852 + N*8	PeakHighFrequencyMin0N (Index N = 1 bis 4)	UDINT		•		
3684 + N*8	PeakRawMax0N (Index N = 1 bis 4)	UDINT		•		
3044 + N*8	PeakRawMin0N (Index N = 1 bis 4)	UDINT		•		
3428 + N*8	RmsAccEnvelopeMax0N (Index N = 1 bis 4)	UDINT		•		
2788 + N*8	RmsAccEnvelopeMin0N (Index N = 1 bis 4)	UDINT		•		
3364 + N*8	RmsAccRawMax0N (Index N = 1 bis 4)	UDINT		•		
2724 + N*8	RmsAccRawMin0N (Index N = 1 bis 4)	UDINT		•		
3524 + N*8	RmsHighFrequencyMax0N (Index N = 1 bis 4)	UDINT		•		
2884 + N*8	RmsHighFrequencyMin0N (Index N = 1 bis 4)	UDINT		•		
3652 + N*8	RmsRawMax0N (Index N = 1 bis 4)	UDINT		•		
3012 + N*8	RmsRawMin0N (Index N = 1 bis 4)	UDINT		•		
3460 + N*8	RmsVelEnvelopeMax0N (Index N = 1 bis 4)	UDINT		•		
2820 + N*8	RmsVelEnvelopeMin0N (Index N = 1 bis 4)	UDINT		•		
3396 + N*8	RmsVelRawMax0N (Index N = 1 bis 4)	UDINT		•		

Registerbeschreibung

Register	Name	Datentyp	Lesen		Schreiben	
			Zyklisch	Azyklisch	Zyklisch	Azyklisch
2756 + N*8	RmsVelRawMin0N (Index N = 1 bis 4)	UDINT		•		
3620 + N*8	SkewnessRawMax0N (Index N = 1 bis 4)	DINT		•		
2980 + N*8	SkewnessRawMin0N (Index N = 1 bis 4)	DINT		•		
Frequenzkonfiguration						
1302	HighFrequencyConfig01	UINT				•
550	HighFrequencyConfig01Read	UINT		•		
1306	MaxFrequencyEnvelope01	UINT				•
558	MaxFrequencyEnvelope01Read	UINT		•		
526	MaxFrequencyRaw01	UINT				•
554	MaxFrequencyRaw01Read	UINT		•		
522	MinFrequencyEnvelope01	UINT				•
566	MinFrequencyEnvelope01Read	UINT		•		
518	MinFrequencyRaw01	UINT				•
562	MinFrequencyRaw01Read	UINT		•		
Frequenzbänder						
3716 + N*8	FrequencyBandMaxN (Index N = 01 bis 32)	UDINT		•		
3076 + N*8	FrequencyBandMinN (Index N = 01 bis 32)	UDINT		•		
956 + N*8	FrequencyBandN (Index N = 01 bis 32)	UDINT		•		
506 + N*24	FrequencyBandNConfig (Index N = 01 bis 32)	UINT				•
1194 + N*24	FrequencyBandNConfigRead (Index N = 01 bis 32)	UINT		•		
514 + N*24	FrequencyBandNDmgFreq60rpm (Index N = 01 bis 32)	UINT				•
1202 + N*24	FrequencyBandNDmgFreq60rpmRead (Index N = 01 bis 32)	UINT		•		
522 + N*24	FrequencyBandNLowerFrequency (Index N = 01 bis 32)	UINT				•
1210 + N*24	FrequencyBandNLowerFrequencyRead (Index N = 01 bis 32)	UINT		•		
518 + N*24	FrequencyBandNTolerance (Index N = 01 bis 32)	UINT				•
1206 + N*24	FrequencyBandNToleranceRead (Index N = 01 bis 32)	UINT		•		
526 + N*24	FrequencyBandNUpperFrequency (Index N = 01 bis 32)	UINT				•
1214 + N*24	FrequencyBandNUpperFrequencyRead (Index N = 01 bis 32)	UINT		•		

4.5 Funktionsmodell 254 - Bus Controller

Dieses Funktionsmodell kann nur mit einem CANIO Bus Controller verwendet werden. Es beinhaltet die gleiche Funktionalität wie [Funktionsmodell 2 - Langsamer Master](#).

Die Unterschiede sind:

- Die zyklischen Register sind am Bus etwas anders Angeordnet.
- Da die Datenpunkte AnalogInputToggleBit01-04 nicht konsistent zu den Datenpunkten "[AnalogInput](#)" auf [Seite 32](#) übertragen werden können sind diese in diesem Funktionsmodell nicht verfügbar. Der Anwender muss auf die Wertänderung der Datenpunkten "[AnalogInput](#)" achten um festzustellen, ob ein neuer Wert verfügbar ist.

Register	Offset ¹⁾	Name	Datentyp	Lesen		Schreiben	
				Zyklisch	Azyklisch	Zyklisch	Azyklisch
Allgemeine Register							
2 + N*2	2 + N*4	ActSpeed0N (Index N = 1 bis 4)	UINT			•	
1310		AutogainDelay01	UINT				•
526		AutogainDelay01Read	UINT		•		
0	2	Control01	UINT			•	
514		SensorConfig01	UINT				•
		SensorConfig01Read					
0	2	Status01	UINT	•			
AnalogInput - Funktionen							
2 * N	2 * N*4	AnalogInput0N (Index N = 1 bis 4)	INT	•			
1330		AnalogInputConfig01	UINT				•
570		AnalogInputConfig01Read	UINT		•		
2	22	AnalogInputControlByte01	UINT			•	
1298		AnalogInputScale01	UINT				•
546		AnalogInputScale01Read	UINT		•		
1310 + N*4		SamplesAnalogInput0N (Index N = 1 bis 4)	UINT				•
526 + N*4		SamplesAnalogInput0NRead (Index N = 1 bis 4)	UINT		•		
Kennwerte							
828 + N*8		CrestFactorRaw0N (Index N = 1 bis 4)	UDINT		•		
572 + N*8		Iso10816_0N (Index N = 1 bis 4)	UDINT		•		
796 + N*8		KurtosisRaw0N (Index N = 1 bis 4)	DINT		•		
732 + N*8		PeakHighFrequency0N (Index N = 1 bis 4)	UDINT		•		
924 + N*8		PeakRaw0N (Index N = 1 bis 4)	UDINT		•		
668 + N*8		RmsAccEnvelope0N (Index N = 1 bis 4)	UDINT		•		
604 + N*8		RmsAccRaw0N (Index N = 1 bis 4)	UDINT		•		
764 + N*8		RmsHighFrequency0N (Index N = 1 bis 4)	UDINT		•		
982 + N*8		RmsRaw0N (Index N = 1 bis 4)	UDINT		•		
700 + N*8		RmsVelEnvelope0N (Index N = 1 bis 4)	UDINT		•		
636 + N*8		RmsVelRaw0N (Index N = 1 bis 4)	UDINT		•		
860 + N*8		SkewnessRaw0N (Index N = 1 bis 4)	DINT		•		
Kennwerte Minimal- und Maximalwerte							
2690		MinMaxCounter01	UINT		•		
3588 + N*8		CrestFactorRawMax0N (Index N = 1 bis 4)	UDINT		•		
2948 + N*8		CrestFactorRawMin0N (Index N = 1 bis 4)	UDINT		•		
3332 + N*8		Iso10816Max0N (Index N = 1 bis 4)	UDINT		•		
2692 + N*8		Iso10816Min0N (Index N = 1 bis 4)	UDINT		•		
3556 + N*8		KurtosisRawMax0N (Index N = 1 bis 4)	DINT		•		
2916 + N*8		KurtosisRawMin0N (Index N = 1 bis 4)	DINT		•		
3492 + N*8		PeakHighFrequencyMax0N (Index N = 1 bis 4)	UDINT		•		
2852 + N*8		PeakHighFrequencyMin0N (Index N = 1 bis 4)	UDINT		•		
3684 + N*8		PeakRawMax0N (Index N = 1 bis 4)	UDINT		•		
3044 + N*8		PeakRawMin0N (Index N = 1 bis 4)	UDINT		•		
3428 + N*8		RmsAccEnvelopeMax0N (Index N = 1 bis 4)	UDINT		•		
2788 + N*8		RmsAccEnvelopeMin0N (Index N = 1 bis 4)	UDINT		•		
3364 + N*8		RmsAccRawMax0N (Index N = 1 bis 4)	UDINT		•		
2724 + N*8		RmsAccRawMin0N (Index N = 1 bis 4)	UDINT		•		
3524 + N*8		RmsHighFrequencyMax0N (Index N = 1 bis 4)	UDINT		•		
2884 + N*8		RmsHighFrequencyMin0N (Index N = 1 bis 4)	UDINT		•		
3652 + N*8		RmsRawMax0N (Index N = 1 bis 4)	UDINT		•		
3012 + N*8		RmsRawMin0N (Index N = 1 bis 4)	UDINT		•		
3460 + N*8		RmsVelEnvelopeMax0N (Index N = 1 bis 4)	UDINT		•		
2820 + N*8		RmsVelEnvelopeMin0N (Index N = 1 bis 4)	UDINT		•		
3396 + N*8		RmsVelRawMax0N (Index N = 1 bis 4)	UDINT		•		
2756 + N*8		RmsVelRawMin0N (Index N = 1 bis 4)	UDINT		•		
3620 + N*8		SkewnessRawMax0N (Index N = 1 bis 4)	DINT		•		
2980 + N*8		SkewnessRawMin0N (Index N = 1 bis 4)	DINT		•		
Frequenzkonfiguration							
1302		HighFrequencyConfig01	UINT				•
550		HighFrequencyConfig01Read	UINT		•		

Register	Offset ¹⁾	Name	Datentyp	Lesen		Schreiben	
				Zyklisch	Azyklisch	Zyklisch	Azyklisch
1306		MaxFrequencyEnvelope01	UINT				•
558		MaxFrequencyEnvelope01Read	UINT		•		
526		MaxFrequencyRaw01	UINT				•
554		MaxFrequencyRaw01Read	UINT		•		
522		MinFrequencyEnvelope01	UINT				•
566		MinFrequencyEnvelope01Read	UINT		•		
518		MinFrequencyRaw01	UINT				•
562		MinFrequencyRaw01Read	UINT		•		
Frequenzbänder							
3716 + N*8		FrequencyBandMaxN (Index N = 01 bis 32)	UDINT		•		
3076 + N*8		FrequencyBandMinN (Index N = 01 bis 32)	UDINT		•		
956 + N*8		FrequencyBandN (Index N = 01 bis 32)	UDINT		•		
506 + N*24		FrequencyBandNConfig (Index N = 01 bis 32)	UINT				•
1194 + N*24		FrequencyBandNConfigRead (Index N = 01 bis 32)	UINT		•		
514 + N*24		FrequencyBandNDmgFreq60rpm (Index N = 01 bis 32)	UINT				•
1202 + N*24		FrequencyBandNDmgFreq60rpmRead (Index N = 01 bis 32)	UINT		•		
522 + N*24		FrequencyBandNLowerFrequency (Index N = 01 bis 32)	UINT				•
1210 + N*24		FrequencyBandNLowerFrequencyRead (Index N = 01 bis 32)	UINT		•		
518 + N*24		FrequencyBandNTolerance (Index N = 01 bis 32)	UINT				•
1206 + N*24		FrequencyBandNToleranceRead (Index N = 01 bis 32)	UINT		•		
526 + N*24		FrequencyBand0NUpperFrequency (Index N = 01 bis 32)	UINT				•
1214 + N*24		FrequencyBandNUpperFrequencyRead (Index N = 01 bis 32)	UINT		•		

1) Der Offset gibt an, wo das Register im CAN-Objekt angeordnet ist.

4.5.1 Verwendung des Moduls am Bus Controller

Das Funktionsmodell 254 "Bus Controller" wird defaultmäßig nur von nicht konfigurierbaren Bus Controllern verwendet. Alle anderen Bus Controller können, abhängig vom verwendeten Feldbus, andere Register und Funktionen verwenden.

Für Detailinformationen siehe X20 Anwenderhandbuch (ab Version 3.50), Abschnitt "Zusätzliche Informationen - Verwendung von I/O-Modulen am Bus Controller".

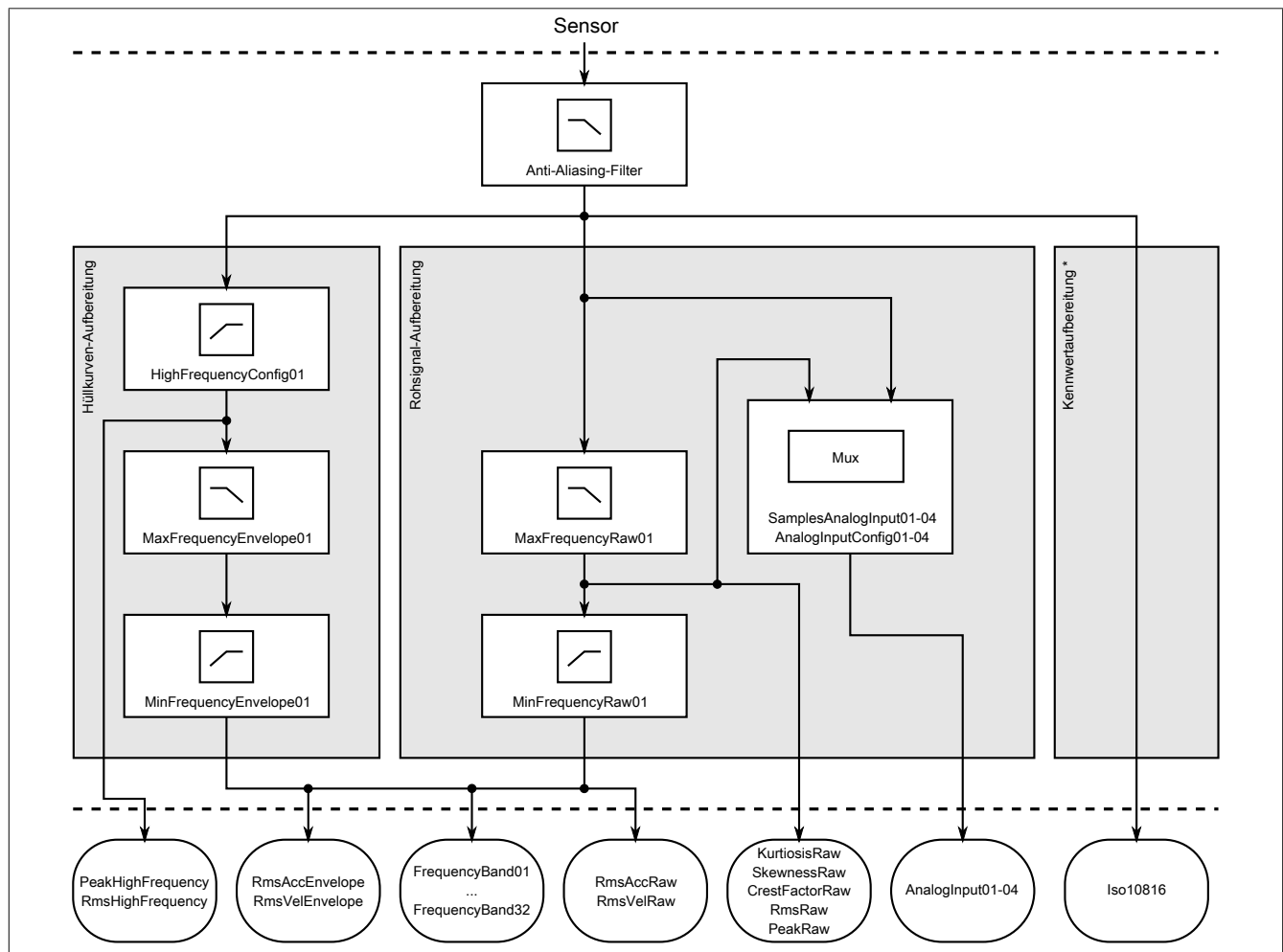
4.5.2 CAN-I/O Bus Controller

Das Modul belegt an CAN-I/O 2 analoge logische Steckplätze.

4.6 Allgemeine Informationen

4.6.1 Signalgenerierung

Folgende Signale und Kennwerte werden aus dem Eingangssignal des Beschleunigungssensors berechnet:



* Aufbereitung nach ISO 10816-3

4.6.2 Filterkonfiguration

Das Modul besitzt verschiedene konfigurierbare Filter.

Für das gesamte Modul gibt es einen einstellbaren Hochpassfilter, welcher über Register **"HighFrequencyConfig"** auf Seite 57 konfiguriert werden kann. Mögliche Einstellungen sind 500 Hz, 1 kHz und 2 kHz. Dieser Hochpass betrifft alle HighFrequency- und Hüllkurvenkennwerte aller Modulkanaäle.

Zusätzlich gibt es pro Kanal noch 2 einstellbare Tiefpassfilter.

- Filterung des Rohsignals. Dieser Filter wird über Register **"MaxFrequencyRaw"** auf Seite 58 konfiguriert. Mögliche Einstellungen sind 200 Hz, 500 Hz, 1 kHz, 2 kHz, 5 kHz und 10 kHz.
- Filterung des Hüllkurvensignals. Dieser Filter wird über Register **"MaxFrequencyEnvelope"** auf Seite 57 konfiguriert. Mögliche Einstellungen sind 200 Hz, 500 Hz, 1 kHz und 2 kHz.

Die Tiefpassfilter beeinflussen alle berechneten Kennwerte des jeweiligen Signals, d. h. des Roh- oder Hüllkurvensignals. Mit ihrer Hilfe kann die Frequenzauflösung in der FFT erhöht werden. Bei der Kennwertberechnung im AnalogInput kann hingegen ausgewählt werden, ob die Kennwerte aus dem direkten Eingangssignal oder dem tiefpassgefilterten Rohsignal berechnet werden sollen.

4.6.3 Frequenzbänder

Es ist möglich bis zu 32 Frequenzbänder individuell zu konfigurieren, in denen der Effektivwert (RMS) oder das Rauschen eines Quadranten berechnet wird.

Parameter	Einstellungen		
Enable	Aus RMS Rauschen		
Kanal	1 2 3 4		
Quelle	Rohsignal Beschleunigung Rohsignal Geschwindigkeit Hüllkurvensignal Beschleunigung Hüllkurvensignal Geschwindigkeit		
Berechnung der Oberwellen (nur bei RMS)	Ja Nein		
Drehzahlabhängig (nur bei RMS)	Ein	Auswahl des Datenpunktes für die Geschwindigkeit (" ActSpeed " auf Seite 27)	[1/100 Hz]
		Normierte Schadfrequenz bei 60 rpm	[1/100]
		± Breite des Frequenzbandes (Toleranzbandes)	[1/100 Hz]
	Aus	Untere Frequenz	[1/4 Hz]
		Obere Frequenz	[1/4 Hz]
Quadrant (nur bei Rauschen)	1. Quadrant 2. Quadrant 3. Quadrant 4. Quadrant		

4.6.4 Automatische Berechnung des Geschwindigkeitssignals

Das Modul kann aus dem Signal des Beschleunigungssensors das Geschwindigkeitssignal berechnen. Diese Berechnung ist per Default deaktiviert, weil dies die Genauigkeit des Beschleunigungssignals verringern kann.

Begründung

Durch die Umrechnung von Beschleunigung in Geschwindigkeit werden niederfrequente Anteile sehr groß. Dadurch wird das [Autogain](#) um einige Stufen herabgesetzt, wodurch in weiterer Folge die Genauigkeit sinkt.

Ist die Berechnung nicht aktiviert, so wird für alle aus dem Geschwindigkeitsspektrum berechneten Kennwerte 0 ausgegeben. Der Kennwert "[Iso10816](#)" auf Seite 43 ist davon nicht betroffen.

4.6.5 Autogain, AutogainDelay und Überlauf

Das Modul besitzt eine automatische, dynamische Anpassung (Autogain) an das Messsignal, um dieses mit höchster Genauigkeit messen zu können. Diese Anpassung besteht aus mehreren Stufen. Mit jeder Stufe wird das Eingangssignal etwas mehr verstärkt. Wenn das Signal sehr lange sehr klein war und plötzlich ein Schlag auftritt kann es vorkommen, dass es bei einigen berechneten Kennwerten zu einem Überlauf kommt. Dies wird dadurch signalisiert, dass das jeweilige Overflowbit des Kanals gesetzt wird (Overflow01-04 im Register "[Status](#)" auf Seite 30) und die betroffenen Kennwerte vom Modul auf ihr Maximum gesetzt werden.

Beim [Funktionsmodell 0 - Standard](#) gibt es noch die zusätzlichen Register "[OverflowCharacteristicValues](#)" auf Seite 38 und "[OverflowFrequencyBands](#)" auf Seite 39. Diese werden automatisch vom Automation Runtime generiert und zeigen direkt den Überlaufstatus der einzelnen Kennwerte und Frequenzbänder an.

Tritt ein Überlauf auf oder wird eine interne Schwelle überschritten, so wird das Autogain für die nächsten Messungen um eine Stufe verringert. Tritt für eine gewisse Anzahl an Messungen (einstellbar durch Register "[AutogainDelay](#)" auf Seite 27) kein Überlauf auf oder bleibt das Signal unter der internen Schwelle, so wird das Autogain wieder um eine Stufe erhöht.

Bei häufigen Überläufen kann eine Erhöhung der Konfiguration AutogainDelay Abhilfe verschaffen.

4.6.6 Begriffsdefinition: Abtastrate und Abtastfrequenz

In diesem Dokument werden die Begriffe Abtastrate und Abtastfrequenz verwendet. Im Anschluss erfolgt eine Begriffsdefinition:

Begriff	Definition
Abtastrate	Abtastung eines analogen Signals pro Zeiteinheit. In der Regel wird als Zeiteinheit 1 s verwendet. Beispiel: 100 Abtastungen pro Sekunde
Abtastfrequenz	Abtastung eines analogen Signals bezogen auf 1 s. Die Angabe erfolgt in Hertz [Hz]. Beispiele: - Die Abtastung eines analogen Signals 1x pro Sekunde entspricht einer Abtastfrequenz von 1 Hz - Die Abtastung eines analogen Signals 1x pro ms entspricht einer Abtastfrequenz von 1 kHz

4.7 Allgemeine Register

4.7.1 ActSpeed

Name:

ActSpeed01 bis ActSpeed04

Register für die aktuelle Geschwindigkeit zum Berechnen der Frequenzbänder01 bis 32, sofern diese geschwindigkeitsabhängig konfiguriert wurden.

Die aktuelle Geschwindigkeit muss in 1/100 Hz angegeben werden. Im [Funktionsmodell 0 - Standard](#) übernimmt dies das Automation Runtime.

Sollten die 4 verschiedenen Geschwindigkeitsdatenpunkte nicht ausreichend sein, z. B. bei mehreren verschiedenen Übersetzungen, so kann optional das Drehzahlverhältnis in die normierte Schadfrequenz des Frequenzbandes (Register "[FrequencyBandDmgFreq60rpm](#)" auf [Seite 66](#)) mit eingerechnet werden.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Werte	Einheit
REAL	0 bis 655,35 ¹⁾	in 1 Hz

1) Größere Werte werden vom Treiber auf 655,35 reduziert.

Kennwert im allen anderen Funktionsmodellen

Format	Werte	Einheit
UINT	0 bis 65.535	in 1/100 Hz

4.7.2 AutogainDelay

Name:

AutogainDelay01

Mit Hilfe dieses Registers kann für alle 4 Kanäle die Verzögerung für das Autogain eingestellt werden.

Damit auch kleinere Signale genau berechnet werden können, kann das Autogain in Stufen erhöht werden. Dies geschieht, wenn für die in diesem Register konfigurierte Anzahl von Messzyklen kein Überlauf stattgefunden hat und immer alle Bedingungen für die nächste Stufe erfüllt waren. Tritt jedoch ein Überlauf auf, so wird das Autogain sofort um eine Stufe verringert.

Die Einheit der Verzögerung des Autogains wird in Messzyklen (300 ms) angegeben.

Datentyp	Werte	Information
UINT	1 bis 65.535	Anzahl Messzyklen; Bus Controller Default: 50

4.7.3 AutogainDelayRead

Name:

AutogainDelay01Read

Register zum Auslesen der aktuellen "[AutogainDelay](#)" auf [Seite 27](#) Konfiguration.

Datentyp	Werte
UINT	0 bis 65.535

4.7.4 Control

Name:

Control01

Allgemeines Steuerregister für das Modul.

Datentyp	Werte
UINT	Siehe Bitstruktur

Bitstruktur:

Bit	Beschreibung	Wert	Information
0	RequestBufferLock01 bzw. RequestDataLock01	0	Daten sollen nicht eingefroren werden
		1	Daten sollen eingefroren werden
1	MinMaxUpdate01	x	Mit jeder Flanke werden die azyklischen Minimal- und Maximalwerte aktualisiert
2 - 15	Reserviert	0	

RequestBufferLock01 bzw. RequestDataLock01

"Funktionsmodell 0 - Standard" und "1 - Schneller Master"

Mit Hilfe von RequestBufferLock01 können alle Puffer des Moduls eingefroren werden. Wird das Bit auf 1 gesetzt, so werden mit Beginn der nächsten Messung alle Puffer (Rohdaten und FFT) eingefroren. Bevor ein Puffer hochgeladen werden kann, müssen die darin enthaltenen Daten zwingend eingefroren werden. Die Kennwerte, welche zu den eingefrorenen Puffern gehören, werden im Kennwert Flatstream übertragen, sobald BufferLockValid01 = 1 wird.

Information:

Da die Messung durchgängig ist werden die Parameter, welche zu den Puffern gehören, nur 1x übertragen.

"Funktionsmodell 2 - Langsamer Master" und "254 - Bus Controller"

Mit Hilfe von RequestDataLock01 können alle Parameter des Moduls eingefroren werden. Wird das Bit auf 1 gesetzt, so werden mit Beginn der nächsten Messung alle Messdaten konsistent zueinander gehalten und eingefroren. Nachdem die Daten am Modul eingefroren sind können alle berechneten Kennwert azyklisch vom Modul ausgelesen werden.

Die Daten des Moduls sind erst dann eingefroren, wenn das Bit BufferLockValid01 bzw. DataLockValid01 im Register "Status01" auf Seite 30 gesetzt ist.

Nachdem die Daten abgeholt wurden kann das Bit RequestBufferLock01 bzw. RequestDataLock01 wieder auf 0 zurückgesetzt werden. Die Daten am Modul sind nicht mehr eingefroren, sobald BufferLockValid01 bzw. DataLockValid01 im Register "Status01" auf Seite 30 wieder 0 ist.

Ein erneutes Einfrieren wird vom Modul nur dann zugelassen wenn die Puffer des Kanals mit der größten Pufferlänge wieder gefüllt sind. Die Pufferlänge hängt von den Einstellungen der Register "MaxFrequency-Raw" auf Seite 58 und "MaxFrequencyEnvelope" auf Seite 57 ab.

MinMaxUpdate01

Mit Hilfe einer Flanke von MinMaxUpdate01 werden alle azyklischen Minimal- und Maximalwerte aktualisiert. Danach startet intern ein neuer Zyklus für die Generierung der Minimal- und Maximalwerte, welche mit der nächsten Flanke wieder auf die azyklischen Register kopiert werden. Nachdem eine Flanke angelegt wurde, können im nächsten X2X-Zyklus die aktuellen Minimal- und Maximalwerte azyklisch ausgelesen werden. Das Register "MinMaxCounter" auf Seite 53 gibt dabei an, wie viele Messzyklen bei der Minimum- und Maximumbildung ausgewertet wurden. Die Minimal- und Maximalwerte selbst sind nur gültig, wenn der Zähler nicht 0 ist.

4.7.5 SensorConfig

Name:

SensorConfig01

Mit diesem Register kann die IEPE-Sensorversorgung der einzelnen Kanäle aktiviert bzw. deaktiviert werden.

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	0

Bitstruktur:

Bit	Beschreibung	Wert	Information
0	Kanal 1: IEPE-Versorgung	0	Aus (Bus Controller Default)
		1	Ein
...		...	
3	Kanal 4: IEPE-Versorgung	0	Aus (Bus Controller Default)
		1	Ein
4 - 7	Reserviert	0	
8	Kanal 1: EnableVelocityCalculation	0	Keine Berechnung (Bus Controller Default)
		1	Berechnung aktiviert
...		...	
11	Kanal 4: EnableVelocityCalculation	0	Keine Berechnung (Bus Controller Default)
		1	Berechnung aktiviert
12 - 13	Reserviert	0	
14	Pufferlänge	0	8192 Messwerte (Bus Controller Default)
		1	65535 Messwerte
15	Auswahl des Funktionsmodells	0	Funktionsmodell 2 - Langsamer Master und Funktionsmodell 254 - Bus Controller (Bus Controller Default)
		1	Funktionsmodell 1 - Schneller Master

EnableVelocityCalculation

Mit Hilfe dieses Bits kann die Berechnung aller auf dem Geschwindigkeitsspektrum berechneten Kennwerte aktiviert werden.

Ist die Berechnung nicht aktiviert, so wird für alle auf dem Geschwindigkeitsspektrum berechneten Kennwerte 0 ausgegeben.

Um die Genauigkeit der auf dem Beschleunigungsspektrum basierenden Kennwerte zu erhöhen wird empfohlen, dieses Bit nur zu aktivieren, wenn die Geschwindigkeitssignale unbedingt benötigt werden.

4.7.6 SensorConfigRead

Name:

SensorConfig01Read

Register zum Auslesen der aktuellen "SensorConfig" auf Seite 29 Konfiguration.

Datentyp	Werte
UINT	0 bis 65.535

4.7.7 Status

Name:
Status01

Allgemeines Statusregister für das Modul.

Datentyp	Werte
UINT	Siehe Bitstruktur

Bitstruktur:

Bit	Beschreibung	Wert	Information
0	Kanal 1: BrokenWire01	0	Kein Fehler
		1	Drahtbruch
...		...	
3	Kanal 4: BrokenWire04	0	Kein Fehler
		1	Drahtbruch
4	BufferLockValid01 bzw. DataLockValid01 ¹⁾	0	Daten nicht eingefroren
		1	Daten sind eingefroren, konsistent und gültig
5	Kanal 1: Overflow01	0	Kein Fehler
		1	Überlauf eines oder mehreren Kennwerten
...		...	
8	Kanal 4: Overflow04	0	Kein Fehler
		1	Überlauf eines oder mehreren Kennwerten
9	Kanal 1: AnalogInputToggleBit01 ²⁾	0	Wechselt nicht
		1	Wechselt
...		...	
12	Kanal 4: AnalogInputToggleBit04 ²⁾	0	Wechselt nicht
		1	Wechselt
13 - 15	Reserviert	-	

1) Bestätigung von RequestBufferLock01 bzw. RequestDataLock01 aus Register "Control" auf Seite 28

2) Wechselt mit jeder neuen Berechnung bzw. jedem neuen Eingangswert von Register "AnalogInput0x" auf Seite 32

4.8 AnalogInput-Funktionen

Die 4 Beschleunigungssensoreingänge des Moduls können zusätzlich direkt als analoge Eingänge mit diversen Sonderfunktionen verwendet werden (siehe Register ["AnalogInput" auf Seite 32](#)).

Durch die Konfiguration kann die Auflösung der analogen Eingänge eingestellt werden (["AnalogInputScale" auf Seite 35](#)). Je kleiner der Maximalwert, desto größer die Auflösung des Registers und umgekehrt. Beim Überschreiten des Maximalwertes wird das Register auf sein jeweiliges Maximum beschränkt (positiv bzw. negativ).

Mit Hilfe eines Toggle Bits (AnalogInputToggleBit01-04) wird signalisiert, wann ein neuer Wert übertragen wurde.

Zur Auswahl stehen folgende Funktionen:

- Normale AnalogInput-Funktion
- Kennwertberechnung im kontinuierlichen Modus mit Enable (Continuous Mode)
- Kennwertberechnung im Triggermodus (Single Shot)

4.8.1 Messwerte

Es werden immer die letzten 8 Messwerte vor dem X2X Zyklus gemittelt und am Bus übertragen. Hierbei wird das direkte Eingangssignal (Rohsignal, maximal 10 kHz) mit einer Abtastfrequenz von 25,781 kHz verwendet und ist nicht Mittelwertbereinigt. Der Wert wird entsprechend der Konfiguration skaliert. (siehe Register ["AnalogInputScale" auf Seite 35](#))

4.8.2 Kennwertberechnung im AnalogInput

Folgende Kennwerte können direkt im Register ["AnalogInput" auf Seite 32](#) angezeigt werden. Dabei ist auf die eingestellte Skalierung zu achten.

- Mittelwert
- Spitzenwert (Betrag)
- Effektivwert
- Crest-Faktor

Für die Berechnung des eingestellten Kennwertes stehen 2 Signale zur Auswahl:

- Eingangssignal auf 10 kHz gefiltert, mit einer Abtastfrequenz von 25,781 kHz und ohne Mittelwertbereinigung.
- Auf Maximalfrequenz gefiltertes Rohsignal mit einer Abtastfrequenz abhängig von der Konfiguration ["Max-FrequencyRaw" auf Seite 58](#) und mit einer Mittelwertbereinigung der letzten 8192 Abtastungen.

Ebenso kann mit Hilfe der Register ["SamplesAnalogInput" auf Seite 36](#) eingestellt werden, über wie viele Abtastwerte der jeweilige Kennwert berechnet werden soll. Die Zeit zwischen 2 Abtastungen ist dabei abhängig von der maximalen Frequenz.

Es stehen 2 Modi zur Verfügung:

- ["Kontinuierlicher Modus mit Enable \(Continuous Mode\)" auf Seite 32](#)
- ["Trigger Modus \(Single Shot\)" auf Seite 32](#)

4.8.2.1 Kontinuierlicher Modus mit Enable (Continuous Mode)

Dieser Modus bietet folgende Vorteile:

- Bei richtiger Parametrierung kann nichts übersehen werden.
- Mit Hilfe des Enables kann nach einem Ereignis die Messung im Modul gestartet werden bzw. können Ereignisse ausgeblendet werden.
- Mit jedem neuen Wert ändert das Toggle-Bit seine Wertigkeit.

Bei der Konfiguration muss auf Folgendes geachtet werden:

- Um eine lückenlose Messung garantieren zu können muss die Abtastzeit (Anzahl der Abtastungen * Abtastrate) größer als der X2X Link Zyklus sein (siehe Register "[SamplesAnalogInput](#)" auf Seite 36).
- Wird eine kleinere Abtastzeit als der X2X Link Zyklus konfiguriert, so wird immer die letzte vollständige Messung übertragen.

Information:

In diesem Modus gehen Werte verloren. Die gemessenen Werte können nicht am Bus übertragen werden, da pro X2X Link Zyklus mehrere Werte berechnet werden.

4.8.2.2 Trigger Modus (Single Shot)

Dieser Modus bietet folgende Vorteile:

- Es wird immer nur genau eine Messung ausgeführt.
- Dadurch, dass der Trigger Flankensensitiv ist kann in jedem X2X Link Zyklus neu getriggert werden.
- Mit jedem neuen Wert ändert das Toggle-Bit seine Wertigkeit.

Bei der Konfiguration muss auf Folgendes geachtet werden:

- Ein erneuter Trigger während einer laufenden Messung wird ignoriert. Durch die Laufzeit am Bus kann aber schon wieder erneut getriggert werden, bevor das Toggle-Bit gewechselt hat.
- Wird eine kleinere Abtastzeit als der X2X Link Zykluszeit konfiguriert, so wird immer die erste vollständige Messung übertragen.
- Beim Trigger Modus gehen zwangsweise Werte verloren, da die Aufnahme der Messwerte azyklisch zum X2X Link erfolgt und nicht dauerhaft synchronisiert werden kann.
- Durch den Trigger wird eine temporäre Synchronisation zum X2X Link erreicht.

4.8.3 AnalogInput

Name:

AnalogInput01 bis AnalogInput04

Je nach Konfiguration enthalten diese Register

- den aktuellen Eingangswert des dazugehörigen Eingangs auf die letzten 8 Abtastungen gemittelt,
- oder den zu berechnenden Kennwert über die konfigurierte Anzahl an Abtastungen.

Die Skalierung des Wertes im Register hängt von der Konfiguration des Registers "[AnalogInputScale](#)" auf Seite 35 ab.

Sollte, durch die Skalierung bedingt, der Wert größer als der Wertebereich eines INT werden, so wird dieser auf das Minimum bzw. Maximum eines INT begrenzt. Das Overflow Bit des dazugehörigen Kanals wird dabei nicht gesetzt.

Information:

Im Funktionsmodell 0 - Standard wird der AnalogInput automatisch, mit Beachtung von SensitivitySensor und AnalogInputScale, vom Automation Runtime auf mg bzw. dimensionslos (Crest-Faktor) skaliert. Sollte der Wert des AnalogInput ohne Sensorskalierung den Wertebereich von AnalogInputScale bezogen auf 100 mV/g überschreiten, so wird das dazugehörige AnalogInputOverflow Bit gesetzt.

Datentyp	Werte
INT	-32768 bis 32767

4.8.4 AnalogInputConfig

Name:

AnalogInputConfig01

Register zur Konfiguration der Kennwertberechnung in "AnalogInput" auf Seite 32. Wird nur benötigt, wenn "SamplesAnalogInput" auf Seite 36 des jeweiligen Kanals größer als 0 ist.

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	0

Bitstruktur:

Bit	Beschreibung	Wert	Beschreibung
0 - 1	Zu berechnender Wert in AnalogInput01	0	Mittelwert (Bus Controller Default)
		1	Spitzenwert
		2	Effektivwert
		3	Crest-Faktor
2 - 3	Zu berechnender Wert in AnalogInput02	x	Für mögliche Werte siehe Bit 0-1
4 - 5	Zu berechnender Wert in AnalogInput03	x	Für mögliche Werte siehe Bit 0-1
6 - 7	Zu berechnender Wert in AnalogInput04	x	Für mögliche Werte siehe Bit 0-1
8	Trigger Modus AnalogInput01	0	Kontinuierlich mit Enable (Bus Controller Default)
		1	Einmalig mit Trigger
...	...	...	...
11	Trigger Modus AnalogInput04	0	Kontinuierlich mit Enable (Bus Controller Default)
		1	Einmalig mit Trigger
12	Signalquelle für Kennwertberechnung AnalogInput01	0	Rohsignal auf konfigurierte Maximalfrequenz (siehe "MaxFrequencyRaw" auf Seite 58) gefiltert mit Mittelwertbereinigung. ¹⁾ (Bus Controller Default)
		1	Rohsignal auf 10 kHz gefiltert ohne Mittelwertbereinigung
...	...	...	...
15	Signalquelle für Kennwertberechnung AnalogInput04	0	Rohsignal auf konfigurierte Maximalfrequenz (siehe "MaxFrequencyRaw" auf Seite 58) gefiltert mit Mittelwertbereinigung. ¹⁾ (Bus Controller Default)
		1	Rohsignal auf 10 kHz gefiltert ohne Mittelwertbereinigung

1) Für die Mittelwertbereinigung wird der Mittelwert der letzten 8192 Abtastungen verwendet.

4.8.5 AnalogInputConfigRead

Name:

AnalogInputConfig01Read

Register zum Auslesen der aktuellen AnalogInputConfig01 Konfiguration.

Datentyp	Werte
UINT	0 bis 65.535

4.8.6 AnalogInputControlByte

Name:

AnalogInputControlByte01

Das Steuerregister für "AnalogInput" auf Seite 32 ist nur dann funktionsfähig, wenn das jeweils dazugehörige Konfigurationsregister "SamplesAnalogInput" auf Seite 36 größer als 0 ist.

Die Konfiguration in "AnalogInputConfig" auf Seite 33 des jeweiligen Kanals entscheidet dabei, ob das jeweilige Bit ein Enable- oder ein Triggerbit ist.

Datentyp	Werte
UINT	Siehe Bitstruktur

Bitstruktur:

Bit	Beschreibung	Wert	Information
0	AnalogInputControl01	x	Siehe Funktion des Bits
...		...	
3	AnalogInputControl04	x	Siehe Funktion des Bits
4 - 15	Reserviert	0	

Funktion des Bits

Funktion im kontinuierlichen Modus:

Bit zum Starten der kontinuierlichen Kennwertberechnung von "AnalogInput".

0 = Keine Berechnung

1 = Kennwerte kontinuierlich berechnen

Funktion im single-shot Modus:

Bit zum Starten einer neuen Kennwertberechnung von "AnalogInput".

0, 1 Jede Flanke startet eine neue Kennwertberechnung, sofern die alte bereits abgeschlossen wurde.

Kennwertberechnung

Der in "AnalogInputConfig" auf Seite 33 konfigurierte Kennwert des jeweiligen Kanals wird berechnet. Dabei wird die in "SamplesAnalogInput" auf Seite 36 konfigurierte Anzahl an Abtastungen verwendet. Der berechnete Wert wird in "AnalogInput" auf Seite 32 mit der in "AnalogInputScale" auf Seite 35 konfigurierten Skalierung des jeweiligen Kanals angezeigt. AnalogInputToggleBit01-04 ändert dabei mit jeder neuen Berechnung seinen Wert.

4.8.7 AnalogInputSamples

Name:

AnalogInputSamples01 bis AnalogInputSamples04

Ist Bit 15 des dazugehörigen "SamplesAnalogInput" auf Seite 36 Register gleich 1, so wird dieses Register zur zyklischen Vorgabe der Anzahl an Abtastungen zur Kennwertberechnung verwendet.

Information:

Wird das Register geändert, während eine Messung läuft, so wird die aktuelle Messung verworfen (AnalogInputToggleBit0X wechselt nicht). Darauf ist vor allem im kontinuierlichen Modus mit Enable zu achten.

Datentyp	Werte	Information
UINT	0	Ungültig
	1 bis 8191	Kennwertberechnung des jeweiligen Kanals im dazugehörigen AnalogInput aktiv
	> 8191	Ungültig

Die Zeit zwischen 2 Abtastungen ist abhängig von Register "MaxFrequencyRaw" auf Seite 58:

Maximale Frequenz	Abtastzeit (Zeit zwischen 2 Abtastungen)
10000 Hz	38,79 µs
5000 Hz	77,58 µs
2000 Hz	193,94 µs
1000 Hz	387,88 µs
500 Hz	775,76 µs
200 Hz	1939,39 µs

4.8.8 AnalogInputScale

Name:

AnalogInputScale01

Mit diesem Register kann die Skalierung der 4 Analogeingänge ("[AnalogInput](#)" auf Seite 32) festgelegt werden. Ist der tatsächliche Wert größer als der im Register konfigurierte Wert, wird das jeweilige Register für den Analogeingang ("[AnalogInput](#)" auf Seite 32) auf das positive Maximum begrenzt (32767).

Ist zum Beispiel für AnalogInput04 ± 128 konfiguriert, so stellen die 16 Bit einen Wertebereich von ± 128 g bzw. einheitenlos, wenn der Crest-Faktor berechnet wurde, dar.

Information:

Der eingestellte Skalierwert bezieht sich immer auf einen 100 mV/g Sensor. Wird ein Sensor mit einer anderen Auflösung verwendet, muss dies bei der Konfiguration berücksichtigt werden.

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	0x8888

Bitstruktur:

Bit	Beschreibung	Wert	Information
0 - 3	Skalierung AnalogInput01	0	Ungültig
		1	± 1
		2	± 2
		3	± 4
		4	± 8
		5	± 16
		6	± 32
		7	± 64
		8	± 128 (Bus Controller Default)
		9 bis 15	Ungültig
4 - 7	Skalierung AnalogInput02	x	Für mögliche Werte siehe AnalogInput01
8 - 11	Skalierung AnalogInput03	x	Für mögliche Werte siehe AnalogInput01
12 - 15	Skalierung AnalogInput04	x	Für mögliche Werte siehe AnalogInput01

4.8.9 AnalogInputScaleRead

Name:

AnalogInputScale01Read

Mit diesem Register kann die Skalierung der Analogeingänge ("[AnalogInput](#)" auf Seite 32) ausgelesen werden.

Datentyp	Werte
UINT	0 bis 65.535

4.8.10 SamplesAnalogInput

Name:

SamplesAnalogInput01 bis SamplesAnalogInput04

Ist das dazugehörige SamplesAnalogInput Register gleich 0, liefern die Register "AnalogInput" auf Seite 32 den aktuellen Eingangswert des Analogeingangs.

Ist das SamplesAnalogInput Register größer 0, so wird der in Register "AnalogInputConfig" auf Seite 33 konfigurierte Kennwert des jeweiligen Kanals berechnet. Dabei wird die in diesem Register konfigurierte Anzahl an Abtastungen verwendet und in dem dazugehörigen AnalogInput mit der konfigurierten Skalierung dargestellt.

Ist Bit 15 dieses Registers gleich 1, so wird ebenfalls der in "AnalogInputConfig01" konfigurierte Kennwert des jeweiligen Kanals berechnet. Dabei wird allerdings die im Register "AnalogInputSamples" auf Seite 34 zyklisch vorgegebene Abtastanzahl verwendet und in dem dazugehörigen AnalogInput mit der konfigurierten Skalierung dargestellt.

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	0

Bitstruktur

Bit	Bedeutung	Werte	Information
0 - 14		0	Keine Kennwertberechnung des jeweiligen Kanals im dazugehörigen AnalogInput (Bus Controller Default)
		1 bis 8191	Kennwertberechnung des jeweiligen Kanals im dazugehörigen AnalogInput aktiv
		> 8191	Ungültig
15		0	Vorgabe der Abtastungen erfolgt per Konfiguration über dieses Register
		1	Vorgabe der Abtastungen erfolgt zyklisch über das Register "AnalogInputSamples" auf Seite 34 (Bus Controller Default)

Die Zeit zwischen 2 Abtastungen ist abhängig von Register "MaxFrequencyRaw" auf Seite 58:

Maximale Frequenz	Abtastzeit (Zeit zwischen 2 Abtastungen)
10000 Hz	38,79 µs
5000 Hz	77,58 µs
2000 Hz	193,94 µs
1000 Hz	387,88 µs
500 Hz	775,76 µs
200 Hz	1939,39 µs

4.8.11 SamplesAnalogInputRead

Name:

SamplesAnalogInput01Read bis SamplesAnalogInput04Read

Register zum Auslesen der aktuellen "SamplesAnalogInput" auf Seite 36 Konfiguration.

Datentyp	Werte
UINT	0 bis 65.535

4.9 Automation Runtime Unterstützung

Im [Funktionsmodell 0 - Standard](#) werden einige vom Modul berechnete Kennwerte zusätzlich vom Automation Runtime für den Anwender aufbereitet.

Information:

Für eine fehlerfreie Auswertung ist auf die maximale Zykluszeit zu achten.
Die berechneten Kennwerte des Moduls werden über den Flatstream alle 300 ms vom Modul gestreamt.
Wenn die gestreamten Daten nicht bis zur nächsten Übertragung abgeholt werden, gehen die gemessenen Kennwerte verloren.

Weitere Features durch die Automation Runtime Unterstützung:

- **"ActSpeed"**: Das Modul erwartet sich auf diesen Datenpunkten immer einen Wert in 0,01 Hz Auflösung. Durch die Automation Runtime Unterstützung kann der Anwender im "Standard" Funktionsmodell die aktuelle Geschwindigkeit direkt in Hz angeben.
- **"AnalogInput"**: Der Analogeingang wird automatisch auf die Sensorauflösung und mit dem eingestellten AnalogInputScale skaliert. Anschließend wird er in mg dem Anwender zur Verfügung gestellt. Diese Skalierung gilt nicht für den **"Crest-Faktor"** auf Seite 46, da dieser dimensionslos ist.
- **Kennwerte und Frequenzbänder**: Alle vom Modul berechneten Kennwerte und Frequenzbänder werden flach aufgelegt und können direkt in der I/O-Zuordnung verbunden werden. Sie sind bereits alle auf die richtige Sensorauflösung skaliert und werden je nach Kennwert in mg, mm/s oder dimensionslos (Kurtosis, CrestFactor, Skewness und Vdi3832) angezeigt.
- **Zusätzliche Kennwerte**: Neben den vom Modul berechneten Kennwerten werden durch das Automation Runtime noch folgende Kennwerte automatisch zur Verfügung gestellt:
 - **Vdi3832KtRaw01-04** Benötigt PeakRawRef und RmsRawRef als Referenzwerte und gibt die bei der Berechnung verwendeten Referenzwerte auf PeakRawRefCalculated und RmsRawRefCalculated wieder aus.
 - **CrestFactorHighFrequency01-04** Verhältnis des betragsmäßigen Maximums zum Effektivwert (**"Crest-Faktor"** auf Seite 46) des hochpassgefilterten Eingangssignals. (**"PeakHighFrequency"** auf Seite 42 und **"RmsHighFrequency"** auf Seite 44)
 - **Vdi3832KtHighFrequency01-04** Benötigt PeakHighFrequencyRef und RmsHighFrequencyRef als Referenzwerte und gibt die bei der Berechnung verwendeten Referenzwerte auf PeakHighFrequencyRefCalculated und RmsHighFrequencyRefCalculated wieder aus.

4.9.1 DataConsistentWithLockedBuffers

Name:

DataConsistentWithLockedBuffers01

Werden die Datenpuffer am Modul zum Hochladen eingefroren, so wird mit diesem Bit der Zeitpunkt signalisiert, bei welchem alle Kennwert und Frequenzbänder konsistent zu den eingefrorenen Puffern am Modul sind.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Werte
BOOL	0 oder 1

4.9.2 DataToggleBit

Name:

DataToggleBit01

Dieses Bit ändert seine Wertigkeit immer dann, wenn neue Kennwert vom Modul hochgeladen und aktualisiert wurden (ca. alle 300 ms).

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Werte
BOOL	0 oder 1

4.9.3 OverflowAnalogInput

Name:

OverflowAnalogInput01 bis OverflowAnalogInput04

Zeigt an, ob am Eingang ein Signal anliegt, welches größer als das konfigurierte "AnalogInputScale" auf Seite 35 ist.

Information:

Immer auf einen 100 mV/g Sensor bezogen.

Dieses Register ist nur im Funktionsmodell 0 - Standard in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Werte
BOOL	0 oder 1

4.9.4 OverflowCharacteristicValues

Name:

OverflowCharacteristicValues01 bis OverflowCharacteristicValues04

In diesem Register gibt es für jeden Kennwert des entsprechenden Kanals ein Bit, welches einen Überlauf anzeigt.

Dieses Register ist nur im Funktionsmodell 0 - Standard in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Werte
UINT	Siehe Bitstruktur

Bitstruktur:

Bit	Beschreibung	Wert	Information
0	PeakHighFrequency	0	Kein Fehler
		1	Überlauf
1	RmsHighFrequency	0	Kein Fehler
		1	Überlauf
2	CrestFactorHighFrequency	0	Kein Fehler
		1	Überlauf
3	Vdi3832KtHighFrequency	0	Kein Fehler
		1	Überlauf
4	RmsAccEnvelope	0	Kein Fehler
		1	Überlauf
5	RmsVelEnvelope	0	Kein Fehler
		1	Überlauf
6	RmsAccRaw	0	Kein Fehler
		1	Überlauf
7	RmsVelRaw	0	Kein Fehler
		1	Überlauf
8	PeakRaw	0	Kein Fehler
		1	Überlauf
9	CrestFactorRaw	0	Kein Fehler
		1	Überlauf
10	SkewnessRaw	0	Kein Fehler
		1	Überlauf
11	KurtosisRaw	0	Kein Fehler
		1	Überlauf
12	Vdi3832KtRaw	0	Kein Fehler
		1	Überlauf
13	Iso10816	0	Kein Fehler
		1	Überlauf
14	RmsRaw	0	Kein Fehler
		1	Überlauf
15	Reserviert	0	

4.9.5 OverflowFrequencyBands

Name:

OverflowFrequencyBands01

In diesem Register gibt es für jedes Frequenzband ein Bit, welches einen Überlauf anzeigt.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Werte
UDINT	Siehe Bitstruktur

Bitstruktur:

Bit	Beschreibung	Wert	Information
0	FrequencyBand01	0	Kein Fehler
		1	Überlauf
...		...	
31	FrequencyBand32	0	Kein Fehler
		1	Überlauf

4.9.6 PeakHighFrequencyRef

Name:

PeakHighFrequencyRef01 bis PeakHighFrequencyRef04

Der von der Applikation vorgegebene Referenzwert (Gutzustand) des Betragsmaximums des hochpassgefilterten Eingangssignals, welcher für die Berechnung des Vdi3832 K(t) des hochpassgefilterten Eingangssignals verwendet wird.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.9.7 PeakHighFrequencyRefCalculated

Name:

PeakHighFrequencyRefCalculated01 bis PeakHighFrequencyRefCalculated04

Der vom Modul verwendete Referenzwert (Gutzustand) des Betragsmaximums des hochpassgefilterten Eingangssignals, welcher für die letzte Berechnung des Vdi3832 K(t) des Spitzenwertes verwendet wurde.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.9.8 PeakRawRef

Name:

PeakRawRef01 bis PeakRawRef04

Der von der Applikation vorgegebene Referenzwert (Gutzustand) des Betragsmaximums des Rohsignals, welcher für die Berechnung des Vdi3832 K(t) des Rohsignals verwendet wird.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.9.9 PeakRawRefCalculated

Name:

PeakRawRefCalculated01 bis PeakRawRefCalculated04

Der vom Modul verwendete Referenzwert (Gutzustand) des Betragsmaximums des Rohsignals, welcher für die letzte Berechnung des Vdi3832 K(t) des Rohsignals verwendet wurde.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.9.10 RmsHighFrequencyRef

Name:

RmsHighFrequencyRef01 bis RmsHighFrequencyRef04

Der von der Applikation vorgegebene Referenzwert (Gutzustand) des Effektivwertes des hochpassgefilterten Eingangssignals, welcher für die Berechnung des Vdi3832 K(t) des hochpassgefilterten Eingangssignals verwendet wird.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.9.11 RmsHighFrequencyRefCalculated

Name:

RmsHighFrequencyRefCalculated01 bis RmsHighFrequencyRefCalculated04

Der vom Modul verwendete Referenzwert (Gutzustand) des Effektivwertes des hochpassgefilterten Eingangssignals, welcher für die letzte Berechnung des Vdi3832 K(t) des hochpassgefilterten Eingangssignals verwendet wurde.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.9.12 RmsRawRef

Name:

RmsRawRef01 bis RmsRawRef04

Der von der Applikation vorgegebene Referenzwert (Gutzustand) des Effektivwertes des Rohsignals, welcher für die Berechnung des Vdi3832 K(t) des Rohsignals verwendet wird.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.9.13 RmsRawRefCalculated

Name:

RmsRawRefCalculated01 bis RmsRawRefCalculated04

Der vom Modul verwendete Referenzwert (Gutzustand) des Effektivwertes des Rohsignals, welcher für die letzte Berechnung des Vdi3832 K(t) des Rohsignals verwendet wurde.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.9.14 SensitivitySensor

Name:

SensitivitySensor01 bis SensitivitySensor04

Das Modul berechnet die Kennwerte immer auf der Basis eines 100 mV/g Sensors am Eingang. Bei Verwendung eines anderen Sensor kann man in diesen Registern für jeden Kanal die Sensorauflösung in mV/g angeben. Alle zyklischen Kennwerte werden dann automatisch vom Automation Runtime auf die richtige Auflösung skaliert. Wird dieser Parameter verändert, so ist die nächste Messung, signalisiert mit "[DataToggleBit](#)" auf Seite 37, ungültig.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Datentyp	Information
REAL	Wertebereich abhängig von der verwendeten Sensorempfindlichkeit

4.10 Kennwerte

Für alle azyklischen Kennwert-Modulregister gilt:

- Diese Register sind nur im [Funktionsmodell 2 - Langsamer Master](#) und [Funktionsmodell 254 - Bus Controller](#) verfügbar.
- Mit Hilfe von RequestDataLock01 können alle berechneten Kennwerte eingefroren werden und es ist somit möglich alle Register konsistent zueinander auszulesen.

Es können folgende Kennwerte pro Kanal vom X20CM4810 ausgelesen werden:

Kennwerte	Beschreibung
PeakHighFrequency	Betragsmäßiges Maximum des hochpassgefilterten Eingangssignals.
CrestFactorHighFrequency ¹⁾	Verhältnis des betragsmäßigen Maximums zum Effektivwert des hochpassgefilterten Eingangssignals.
Vdi3832KtHighFrequency ¹⁾	Verhältnis der Referenzwerte zu den aktuell gemessenen Werten des hochpassgefilterten Eingangssignals nach der VDI 3832 Richtlinie.
PeakRaw	Spitzenwert (Betrag) des Eingangssignals bis zur konfigurierten maximalen Frequenz.
CrestFactorRaw	Verhältnis des betragsmäßigen Maximums zum Effektivwert des Eingangssignals bis zur konfigurierten maximalen Frequenz.
SkewnessRaw	Schiefe, drittes statistisches Moment (Skewness) des Eingangssignals bis zur konfigurierten maximalen Frequenz.
KurtosisRaw	Wölbung, viertes statistisches Moment (Kurtosis) des Eingangssignals bis zur konfigurierten maximalen Frequenz.
Vdi3832KtRaw ¹⁾	Verhältnis der Referenzwerte zu den aktuell gemessenen Werten des Eingangssignals nach der VDI 3832 Richtlinie.
RmsHighFrequency	Effektivwert des hochpassgefilterten Eingangssignals.
RmsRaw	Effektivwert des Eingangssignals bis zur konfigurierten maximalen Frequenz.
RmsAccRaw	Effektivwert der Beschleunigung des Eingangssignals von der konfigurierten minimalen Frequenz bis zur konfigurierten maximalen Frequenz.
RmsVelRaw	Effektivwert der Geschwindigkeit des Eingangssignals von der konfigurierten minimalen Frequenz bis zur konfigurierten maximalen Frequenz. ²⁾
Iso10816	Effektivwert der Geschwindigkeit im Frequenzbereich 10 Hz bis 1 kHz nach ISO 10816.
RmsAccEnvelope	Effektivwert der Beschleunigung der Hüllkurve des Eingangssignals von der konfigurierten minimalen Frequenz bis zur konfigurierten maximalen Frequenz.
RmsVelEnvelope	Effektivwert der Geschwindigkeit der Hüllkurve des Eingangssignals von der konfigurierten minimalen Frequenz bis zur konfigurierten maximalen Frequenz. ²⁾

1) Nur im [Funktionsmodell 0 - Standard](#)

2) Wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "[SensorConfig](#)" auf [Seite 29](#)) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

4.10.1 Betragsmaximalwert

Der Betragsmaximalwert wird oft auch als Spitzenwert oder Peakwert bezeichnet.

Der Spitzenwert eines Körperschallsignals gibt den maximalen Betrag einzelner Stöße an, die aus dem Grundrauschen herausragen. Verschiedene Schädigungen bewirken starke Stoßanregungen, die sich im Spitzenwert niederschlagen.

4.10.1.1 PeakHighFrequency

Name:

PeakHighFrequency01 bis PeakHighFrequency04

Register zum Auslesen des Betragsmaximums des hochpassgefilterten Eingangssignals des jeweiligen Kanals.

PeakHighFrequency wird gebildet aus dem hochpassgefilterten Eingangssignal der Schwingbeschleunigung im Frequenzbereich zwischen dem im Register "[HighFrequencyConfig](#)" auf Seite 57 eingestellten Wert und 10 kHz.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	1/65,536	mg	256000,0

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	1/65536	g	16777215

4.10.1.2 PeakRaw

Name:

PeakRaw01 bis PeakRaw04

Register zum Auslesen des Betragsmaximums des Rohsignals des jeweiligen Kanals.

PeakRaw wird gebildet aus dem Rohsignal der Schwingbeschleunigung bis zu der in Register "[MaxFrequencyRaw](#)" auf Seite 58 konfigurierten maximalen Frequenz.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	1/65,536	mg	256000,0

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	1/65536	g	16777215

4.10.2 Effektivwert

Der Effektivwert wird auch quadratischer Mittelwert oder Root Mean Square (RMS) genannt. Er berücksichtigt neben der Amplitude auch noch den Energiegehalt der Schwingung und ist für viele Beurteilungskennwerte der mathematische Hintergrund.

Wird der Effektivwert über die Schwinggeschwindigkeit berechnet, so spricht man oft auch von der Schwingschnelle. Beim Effektivwert werden alle Schwingungsbeiträge aufsummiert - die hohen Schwingungsamplituden einer Unwucht genauso, wie niedrige Schwingungspegel eines gerade beginnenden Lagerschadens.

Wird der Effektivwert breitbandig gemessen können durch die Mittelung Änderungen einzelner Schwingungsbeiträge überdeckt werden. Eine frühzeitige und zuverlässige Detektion von Schäden, z. B. von Wälzlagerdefekten, ist nur bedingt möglich.

4.10.2.1 Iso10816

Name:

Iso10816_01 bis Iso10816_04

Register zum Auslesen des Effektivwertes nach ISO 10816 des jeweiligen Kanals.

Iso10816 wird gebildet aus dem Rohsignal der Schwinggeschwindigkeit im Frequenzbereich von 10 Hz bis 1 kHz.

Dieser Breitbandkennwert wird oft zur Beurteilung des Maschinenzustandes herangezogen, da für diesen Kennwert in der Norm Beurteilungsgrenzen angegeben werden. Diese hängen von der Art der Maschinen und deren Aufstellungsart (starr oder elastisch) ab. Die Kennwerthöhen für eine Vorwarnung oder einer Warnung werden gemäß den definierten Klassen angegeben.

10 - 1000 Hz bei $n > 600 \text{ min}^{-1}$ (2 - 1000 Hz bei $600 \text{ min}^{-1} > n > 120 \text{ min}^{-1}$)	Effektivwert der Schwinggeschwindigkeit $V_{\text{eff, rms}}$ in [mm/s]	> 11,0				
		7,1 - 11,0				
		4,5- 7,1				
		3,5 - 4,5				
		2,8 - 3,5				
		2,3 - 2,8				
		1,4 - 2,3				
		0 - 1,4				
Unterbau		starr	elastisch	starr	elastisch	
Maschinentyp		Mittelgroße Maschinen 15 kW < P < 300 kW		Große Maschinen 300 kW < P < 50 MW		
		Motoren 160 mm < H < 315 mm		Motoren 315 mm < H		
Gruppe		Gruppe 2		Gruppe 1		

Zone D: Gefahr eines Maschinenschadens

Zone C: Begrenzt betriebsfähig

Zone B: Dauerbetrieb ohne Einschränkung

Zone A: Neu in Betrieb gesetzte Maschine

Abbildung 1: Beurteilungsschema ISO

Ist der Zustand der Maschine in Ordnung, dann ist der Kennwert gering. Tritt ein Schaden auf, steigt dieser Wert sehr stark an. Durch die Beurteilung des Effektivwertes im Bereich der Schwinggeschwindigkeit werden vor allem untere Frequenzanteile, wie z. B. die Antriebsdrehzahl und somit Unwucht, Fehlauseinrichtung, usw. hervorgehoben und gehen in den Kennwert stärker ein.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	0,001	mm/s	16777,21

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	1	µm/s	16777215

4.10.2.2 RmsAccEnvelope

Name:

RmsAccEnvelope01 bis RmsAccEnvelope04

Register zum Auslesen des Effektivwertes der Hüllkurvenbeschleunigung des jeweiligen Kanals.

RmsAccEnvelope wird gebildet aus dem Hüllkurvensignal der Schwingbeschleunigung von der konfigurierten minimalen Frequenz ("[MinFrequencyEnvelope](#)" auf Seite 59) bis zur konfigurierten maximalen Frequenz ("[MaxFrequencyEnvelope](#)" auf Seite 57).

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	1	mg	16777215

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	0,001	g	16777215

4.10.2.3 RmsAccRaw

Name:

RmsAccRaw01 bis RmsAccRaw04

Register zum Auslesen des Beschleunigungs-Effektivwertes des jeweiligen Kanals.

RmsAccRaw wird gebildet aus dem Rohsignal der Schwingbeschleunigung von der konfigurierten minimalen Frequenz ("[MinFrequencyRaw](#)" auf Seite 60) bis zur konfigurierten maximalen Frequenz ("[MaxFrequencyRaw](#)" auf Seite 58).

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	1	mg	16777215

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	0,001	g	16777215

4.10.2.4 RmsHighFrequency

Name:

RmsHighFrequency01 bis RmsHighFrequency04

Register zum Auslesen des Hochpass-Effektivwertes des jeweiligen Kanals.

RmsHighFrequency wird gebildet aus dem hochpassgefilterten Eingangssignal der Schwingbeschleunigung im Frequenzbereich zwischen dem im Register "[HighFrequencyConfig](#)" auf Seite 57 eingestellten Wert und 10 kHz.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	1/65,536	mg	256000,0

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	1/65536	g	16777215

4.10.2.5 RmsRaw

Name:

RmsRaw01 bis RmsRaw04

Register zum Auslesen des Rohsignal-Effektivwertes des jeweiligen Kanals.

RmsRaw wird gebildet aus dem Rohsignal der Schwingbeschleunigung bis zu der in Register "[MaxFrequencyRaw](#)" auf Seite 58 konfigurierten maximalen Frequenz.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	1/65,536	mg	256000,0

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	1/65536	g	16777215

4.10.2.6 RmsVelEnvelope

Name:

RmsVelEnvelope01 bis RmsVelEnvelope04

Register zum Auslesen des Effektivwertes der Hüllkurvengeschwindigkeit des jeweiligen Kanals.

RmsVelEnvelope wird gebildet aus dem Hüllkurvensignal der Schwingbeschleunigung von der konfigurierten minimalen Frequenz ("[MinFrequencyEnvelope](#)" auf Seite 59) bis zur konfigurierten maximalen Frequenz ("[MaxFrequencyEnvelope](#)" auf Seite 57).

Dieser Kennwert wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "[Sensor-Config](#)" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	0,001	mm/s	16777,21

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	1	µm/s	16777215

4.10.2.7 RmsVelRaw

Name:

RmsVelRaw01 bis RmsVelRaw04

Register zum Auslesen des Geschwindigkeit-Effektivwertes des jeweiligen Kanals.

RmsVelRaw wird gebildet aus dem Rohsignal der Schwinggeschwindigkeit von der konfigurierten minimalen Frequenz ("[MinFrequencyRaw](#)" auf Seite 60) bis zur konfigurierten maximalen Frequenz ("[MaxFrequencyRaw](#)" auf Seite 58).

Dieser Kennwert wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "[Sensor-Config](#)" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung	Einheit	Wert bei Überlauf
REAL	0,001	mm/s	16777,21

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Einheit	Wert bei Überlauf
UDINT	1	µm/s	16777215

4.10.3 Crest-Faktor

Der Crest-Faktor ist definiert als Quotient aus Spitzenwert und Effektivwert. Bei einer sinusförmigen Schwingung beträgt dieser Faktor $\sqrt{2}$. Dieser Wert ist aus der Elektrotechnik als Scheitelfaktor bekannt.

Bei einem ordnungsgemäß laufenden Lager hat der Crest-Faktor ebenfalls in etwa den Wert $\sqrt{2}$. Verschlechtert sich der Lagerzustand, treten einzelne Stöße auf die den Spitzenwert und damit auch den Crest-Faktor anheben. Kommt allerdings zu einzeln ausgeprägten Defekten noch globaler Verschleiß dazu, hebt dies den Effektivwert ebenfalls an. Der Crest-Faktor kann also im schlimmsten Fall, trotz ansteigender Schädigung, unverändert bleiben oder sogar wieder sinken.

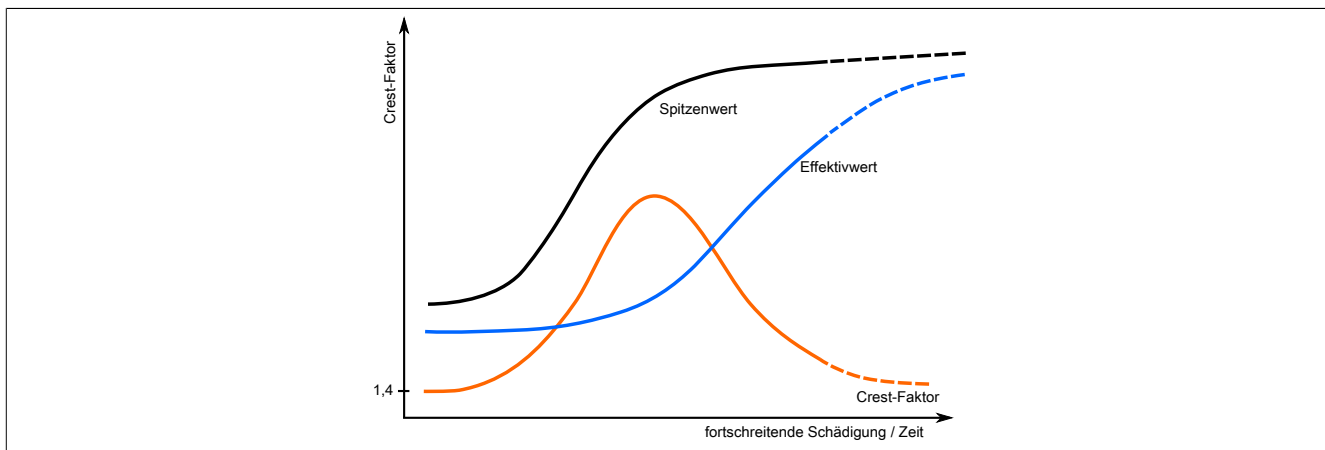


Abbildung 2: Zusammenhang Effektivwert - Spitzenwert

Information:

Wenn eine Crest-Trendaufzeichnung gemacht wird, dann sollen auch der Spitzen- und Effektivwert aufgezeichnet werden

4.10.3.1 CrestFactorHighFrequency

Name:

CrestFactorHighFrequency01 bis CrestFactorHighFrequency04

Register zum Auslesen des Hochpass-Verhältnis des betragsmäßigen Maximums ("[PeakHighFrequency](#)" auf Seite 42) zum Effektivwert ("[RmsHighFrequency](#)" auf Seite 44) des jeweiligen Kanals.

CrestFactorHighFrequency wird gebildet aus dem hochpassgefilterten Eingangssignal und der Schwingbeschleunigung im Frequenzbereich zwischen dem im Register "[HighFrequencyConfig](#)" auf Seite 57 eingestellten Wert und 10 kHz.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Format	Auflösung
REAL	1

4.10.3.2 CrestFactorRaw

Name:

CrestFactorRaw01 bis CrestFactorRaw04

Register zum Auslesen des Rohwert-Verhältnisses des betragsmäßigen Maximums zum Effektivwert des jeweiligen Kanals.

CrestFactor Raw wird gebildet aus dem Rohsignal der Schwingbeschleunigung bis zu der in Register "[MaxFrequencyRaw](#)" auf [Seite 58](#) konfigurierten maximalen Frequenz.

Da im Modul eine Division durch den Effektivwert ("[RmsRaw](#)" auf [Seite 45](#)) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung
REAL	1

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Wert bei Überlauf
UDINT	0,001	16777215

4.10.4 K(t)-Wert

Der K(t) ist in der VDI 3832 Richtlinie beschrieben und berechnet sich aus dem Effektivwert und dem Betragsmaximum des Zeitsignals der Schwingbeschleunigung. Beim Zeitsignal wird dabei das volle zur Verfügung stehende Frequenzspektrum für die Berechnung verwendet.

Dieses Verhältnis wird mit den Referenzwerten in Bezug gesetzt. Die Referenzwerte sollten kurz nach der Einlaufzeit vom Anwender gemessen werden. Diese Werte können als "Anlage in Ordnung" eingestuft werden und sind damit die Ausgangswerte.

Mit fortschreitender Abnutzung sinkt der K(t)-Wert. Damit kann er in 3 Klassen eingeteilt werden:

- Ungeschädigt
- Beginnender Schaden
- Ausgeprägter Schaden

Der Vorteil des K(t)-Werts ist, dass er sich auch bei einem ausgeprägten, fortschreitenden Maschinenschaden nicht mehr stark verändert.

$$K(t) = \frac{a_{rms}(0) \cdot a_p(0)}{a_{rms}(t) \cdot a_p(t)}$$

Dabei entspricht

Formelzeichen	Bedeutung	Kennwert im Modul
$a_{rms}(0)$	Effektivwerte des Referenzwertes	RmsHighFrequencyRef RmsRawRef
$a_p(0)$	Betragsmaximalwert des Referenzwertes	PeakHighFrequencyRef PeakRawRef
$a_{rms}(t)$	Aktueller Effektivwert	RmsHighFrequency RmsRaw
$a_p(t)$	Aktueller Betragsmaximalwert	PeakHighFrequency PeakRaw

Beispiel

Möglicher Verlauf des (K/t) Kennwertes

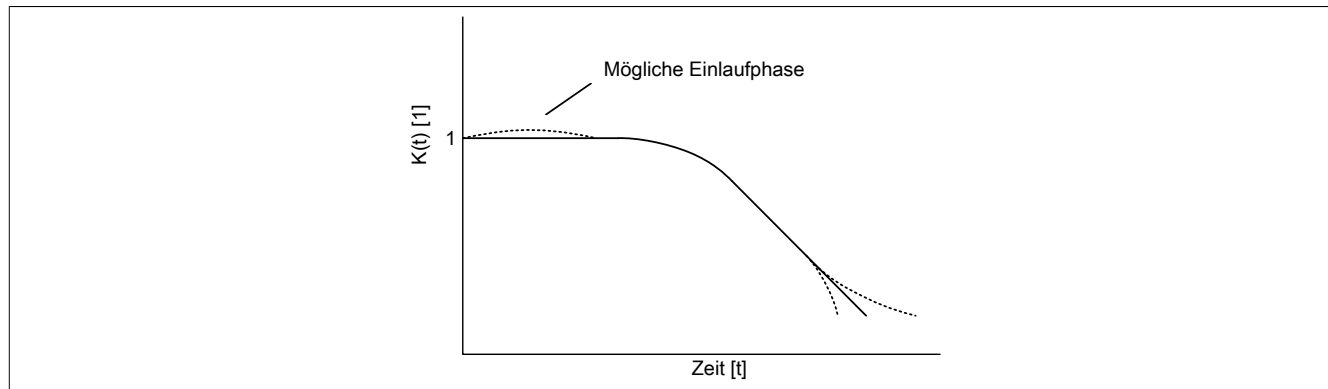


Abbildung 3: K(t)-Wert Verlauf

4.10.4.1 Vdi3832KtHighFrequency

Name:

Vdi3832KtHighFrequency01 bis Vdi3832KtHighFrequency04

Register zum Auslesen des Hochpass-K(t)-Werts nach der VDI 3832 Richtlinie des jeweiligen Kanals.

Vdi3832KtHighFrequency wird gebildet aus dem Spitzenwert ("[PeakHighFrequency](#)" auf Seite 42) und Effektivwert ("[RmsHighFrequency](#)" auf Seite 44) des hochpassgefilterten Eingangssignal und der Schwingbeschleunigung im Frequenzbereich zwischen dem im Register "[HighFrequencyConfig](#)" auf Seite 57 eingestellten Wert und 10 kHz.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Format	Auflösung
REAL	1

4.10.4.2 Vdi3832KtRaw

Name:

Vdi3832KtRaw01 bis Vdi3832KtRaw04

Register zum Auslesen des Rohwert-K(t)-Werts nach der VDI 3832 Richtlinie des jeweiligen Kanals.

Vdi3832KtRaw wird gebildet aus dem Rohsignal der Schwingbeschleunigung bis zu der in Register "[MaxFrequencyRaw](#)" auf Seite 58 konfigurierten maximalen Frequenz.

Dieses Register ist nur im [Funktionsmodell 0 - Standard](#) in der Automation Studio I/O-Zuordnung verfügbar.

Format	Auflösung
REAL	1

4.10.5 Kurtosis-Faktor

Der Kurtosis-Faktor ist ein guter Kennwert um die Spitzenhaltigkeit eines Signals zu beurteilen. Kurtosis (Wölbung, viertes statistisches Moment) ist definiert als das Verhältnis zweier statistischer Kennwerte/Verfahren.

Der Kurtosiswert ist eine Art gewichteter [Crest-Faktor](#). Die Signalspitzen werden auf Grund der vierten Potenz im Integral höher bewertet als der "Rauschwert" (auch "Teppichwert" genannt). Weiters wird nicht nur der Effektivwert herangezogen, sondern der gesamte Signalverlauf. Dadurch steigt der Aussagewert dieses Kennwertes.

Information:

Im Modul wird der Kurtosis-Faktor auf Null normiert.

Ein Kurtosisfaktor kleiner als 2 ist typisch für ein Maschine in gutem Zustand.

4.10.5.1 KurtosisRaw

Name:

KurtosisRaw01 bis KurtosisRaw04

Register zum Auslesen des Kurtosis-Faktor des jeweiligen Kanals.

KurtosisRaw wird gebildet aus dem Rohsignal der Schwingbeschleunigung bis zu der in Register "[MaxFrequencyRaw](#)" auf Seite 58 konfigurierten maximalen Frequenz.

Da im Modul eine Division durch den Effektivwert ("[RmsRaw](#)" auf Seite 45) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung
REAL	1

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Wert bei Überlauf
DINT	0,001	8388607

4.10.6 Skewness-Faktor

Der Skewness-Faktor (Schiefe, drittes statistisches Moment) gibt an wie unsymmetrisch ein Signal, bezogen auf dessen Normalverteilung, ist. Je niedriger der Skewness, desto gleichmäßiger verteilt ist das Signal. Ein Signal mit einem hohen Skewness-Faktor hat viele große Amplituden im Auswertebereich.

Ein symmetrisch verteiltes Signal hat einen Skewness-Faktor von 0. Abhängig von der Richtung der Verschiebung kann der Skewness-Faktor positiv oder negativ sein. Eine erhebliche Verschiebung tritt etwa bei einem Skewness-Faktor von ± 1 auf.

Ein großer Kurtosis-Faktor kombiniert mit einem großen Skewness-Faktor deutet auf eine elektrostatische Entladung hin.

4.10.6.1 SkewnessRaw

Name:

SkewnessRaw01 bis SkewnessRaw04

Register zum Auslesen des Skewness-Faktors des jeweiligen Kanals.

SkewnessRaw wird gebildet aus dem Rohsignal der Schwingbeschleunigung bis zu der in Register "[MaxFrequencyRaw](#)" auf Seite 58 konfigurierten maximalen Frequenz.

Da im Modul eine Division durch den Effektivwert ("[RmsRaw](#)" auf Seite 45) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung
REAL	1

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung	Wert bei Überlauf
DINT	0,001	8388607

4.10.7 FrequencyBand

Name:

FrequencyBand01 bis FrequencyBand32

Register zum Auslesen des jeweiligen Frequenzbandes.

Ist das Frequenzband auf ein Geschwindigkeitssignal parametrisiert, so wird dieser Wert nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "[SensorConfig](#)" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Für weitere Information siehe "[Frequenzbänder](#)" auf Seite 26.

Kennwert im [Funktionsmodell 0 - Standard](#)

Format	Auflösung ¹⁾	Einheit ¹⁾	Wert bei Überlauf
REAL	1	mg	16777215
	1/65,536	mg	256000
	0,001	mm/s	16777,21
	1/65536	mm/s	256

1) Je nach Konfiguration

Kennwert im allen anderen Funktionsmodellen

Format	Auflösung ¹⁾	Einheit ¹⁾	Wert bei Überlauf
24-Bit unsigned	1	mg	16777215
	1/65536	g	256000
	1	µm/s	16777,21
	1/65536	mm/s	256

1) Je nach Konfiguration

4.11 Kennwerte (Minimal und Maximal)

Die Kennwerte des Moduls werden alle 300 ms neu berechnet. Um keine Werte zu verlieren ist es notwendig diese Daten schnell genug abzuholen. Ist das nicht möglich, so können im [Funktionsmodell 2 - Langsamer Master](#) und [Funktionsmodell 254 - Bus Controller](#) die Kennwerte am Modul mit Hilfe des Datenpunktes RequestDataLock01 eingefroren und anschließend azyklisch ausgelesen werden. Allerdings gehen dadurch Messungen verloren.

Um keine Messungen zu verlieren wurde deshalb eine spezielle Funktion im Modul implementiert, welche von allen im Modul berechneten Kennwerten die Minimal- und Maximalwerte aufnimmt. Mit jeder Flanke des Bits MinMayUpdate01 in Registers ["Control" auf Seite 28](#) kann eine neue Messung gestartet werden, wodurch die Minimal- und Maximalwerte wieder neu initialisiert werden. Gleichzeitig werden die aktuellen Daten auf die azyklischen Register kopiert.

Anschließend kann mit Hilfe des azyklischen Registers ["MinMaxCounter" auf Seite 53](#) die Anzahl der erfassten Messungen ausgelesen werden. Mittels der azyklischen Minimal- und Maximalregister werden die jeweiligen Werte ausgelesen.

Beispiel

Für Iso10816 sind es die Register Iso10816Min01-04 und Iso10816Max01-04.

Information:

- Wenn die Daten mit RequestDataLock01 eingefroren werden, werden keine Messwerte mehr erfasst. Dies betrifft nur [Funktionsmodell 2 - Langsamer Master](#) und [Funktionsmodell 254 - Bus Controller](#), da beim Standard Funktionsmodell die Kennwerte nicht eingefroren werden.
- Bei einem Überlauf eines Kennwertes oder Drahtbruch werden keine neuen Werte aufgenommen.
- Die Minimal- und Maximalregister werden jeweils mit dem Minimum bzw. Maximum des jeweiligen Datentyps initialisiert. Wenn für die gesamte Messung kein gültiger Wert auf dem Kennwert liegt, bleibt der Initialisierungswert erhalten (z. B. bei Überlauf, Drahtbruch oder Daten eingefroren).
- Sind die Daten am Modul eingefroren (DataLockValid01 = 1), so werden keine neuen Werte in die Minimumbildung aufgenommen, die Messzyklen werden aber trotzdem weiterhin mitgezählt.

Die Register werden nur mit einer Flanke von ["MinMaxUpdate01" auf Seite 28](#) aktualisiert und sind nur gültig, wenn ["MinMaxCounter" auf Seite 53](#) ungleich 0 ist. Das Register ["MinMaxCounter" auf Seite 53](#) gibt dabei die erfassten Messzyklen für die Minimum- bzw. Maximumbildung an.

4.11.1 CrestFactorRawMax

Name:

CrestFactorRawMax01 bis CrestFactorRawMax04

Maximalwert des "[CrestFactorRaw](#)" auf Seite 47 Kennwertes des jeweiligen Kanals in 1/1000.

Da im Modul eine Division durch den Effektivwert ("[RmsRaw](#)" auf Seite 45) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.2 CrestFactorRawMin

Name:

CrestFactorRawMin01 bis CrestFactorRawMin04

Minimalwert des "[CrestFactorRaw](#)" auf Seite 47 Kennwertes des jeweiligen Kanals in 1/1000.

Da im Modul eine Division durch den Effektivwert ("[RmsRaw](#)" auf Seite 45) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.3 Iso10816Max

Name:

Iso10816Max01 bis Iso10816Max04

Maximalwert des "[Iso10816](#)" auf Seite 43 Kennwertes des jeweiligen Kanals in 1/1000 mm/s.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.4 Iso10816Min

Name:

Iso10816Min01 bis Iso10816Min04

Minimalwert des "[Iso10816](#)" auf Seite 43 Kennwertes des jeweiligen Kanals in 1/1000 mm/s.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.5 KurtosisRawMax

Name:

KurtosisRawMax01 bis KurtosisRawMax04

Maximalwert des "[KurtosisRaw](#)" auf Seite 49 Kennwertes des jeweiligen Kanals in 1/1000.

Da im Modul eine Division durch den Effektivwert ("[RmsRaw](#)" auf Seite 45) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
DINT	-2.147.483.648 bis 2.147.483.647

4.11.6 KurtosisRawMin

Name:

KurtosisRawMin01 bis KurtosisRawMin04

Minimalwert des "[KurtosisRaw](#)" auf Seite 49 Kennwertes des jeweiligen Kanals in 1/1000.

Da im Modul eine Division durch den Effektivwert ("[RmsRaw](#)" auf Seite 45) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
DINT	-2.147.483.648 bis 2.147.483.647

4.11.7 MinMaxCounter

Name:

MinMaxCounter01

Dieses Register gibt an wie viele Messungen bei der letzten Minimum- und Maximumbildung erfasst wurden. Es wird nur mit einer Flanke von MinMaxUpdate01 aktualisiert. Wird nach 65535 Messungen das MinMaxUpdate01 Bit nicht getoggelt, so wird der Zähler auf 65535 begrenzt. Die Minimum- und Maximumbildung läuft aber trotzdem weiter. Alle azyklischen Minimal- und Maximalwerte sind nur gültig, wenn "[MinMaxCounter](#)" auf Seite 53 ungleich 0 ist.

Datentyp	Werte
UINT	0 bis 65.535

4.11.8 PeakHighFrequencyMax

Name:

PeakHighFrequencyMax01 bis PeakHighFrequencyMax04

Maximalwert des "[PeakHighFrequency](#)" auf Seite 42 Kennwertes des jeweiligen Kanals in 1/65536 g.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.9 PeakHighFrequencyMin

Name:

PeakHighFrequencyMin01 bis PeakHighFrequencyMin04

Minimalwert des "[PeakHighFrequency](#)" auf Seite 42 Kennwertes des jeweiligen Kanals in 1/65536 g.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.10 PeakRawMax

Name:

PeakRawMax01 bis PeakRawMax04

Maximalwert des "[PeakRaw](#)" auf Seite 42 Kennwertes des jeweiligen Kanals in 1/65536 g.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.11 PeakRawMin

Name:

PeakRawMin01 bis PeakRawMin04

Minimalwert des "PeakRaw" auf Seite 42 Kennwertes des jeweiligen Kanals in 1/65536 g.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.12 RmsAccEnvelopeMax

Name:

RmsAccEnvelopeMax01 bis RmsAccEnvelopeMax04

Maximalwert des "RmsAccEnvelope" auf Seite 44 Kennwertes des jeweiligen Kanals in 1/1000 g.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.13 RmsAccEnvelopeMin

Name:

RmsAccEnvelopeMin01 bis RmsAccEnvelopeMin04

Minimalwert des "RmsAccEnvelope" auf Seite 44 Kennwertes des jeweiligen Kanals in 1/1000 g.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.14 RmsAccRawMax

Name:

RmsAccRawMax01 bis RmsAccRawMax04

Maximalwert des "RmsAccRaw" auf Seite 44 Kennwertes des jeweiligen Kanals in 1/1000 g.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.15 RmsAccRawMin

Name:

RmsAccRawMin01 bis RmsAccRawMin04

Minimalwert des "RmsAccRaw" auf Seite 44 Kennwertes des jeweiligen Kanals in 1/1000 g.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.16 RmsHighFrequencyMax

Name:

RmsHighFrequencyMax01 bis RmsHighFrequencyMax04

Maximalwert des "RmsHighFrequency" auf Seite 44 Kennwertes des jeweiligen Kanals in 1/65536 g.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.17 RmsHighFrequencyMin

Name:

RmsHighFrequencyMin01 bis RmsHighFrequencyMin04

Minimalwert des "[PeakHighFrequency](#)" auf [Seite 42](#) Kennwertes des jeweiligen Kanals in 1/65536 g.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf [Seite 51](#) für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.18 RmsRawMax

Name:

RmsRawMax01 bis RmsRawMax04

Maximalwert des "[RmsRaw](#)" auf [Seite 45](#) Kennwertes des jeweiligen Kanals in 1/65536 g.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf [Seite 51](#) für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.19 RmsRawMin

Name:

RmsRawMin01 bis RmsRawMin04

Minimalwert des "[RmsRaw](#)" auf [Seite 45](#) Kennwertes des jeweiligen Kanals in 1/65536 g.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf [Seite 51](#) für weitere Informationen.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.20 RmsVelEnvelopeMax

Name:

RmsVelEnvelopeMax01 bis RmsVelEnvelopeMax04

Maximalwert des "[RmsVelEnvelope](#)" auf [Seite 45](#) Kennwertes des jeweiligen Kanals in 1/1000 mm/s.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf [Seite 51](#) für weitere Informationen.

Dieser Wert wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "[SensorConfig](#)" auf [Seite 29](#)) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.21 RmsVelEnvelopeMin

Name:

RmsVelEnvelopeMin01 bis RmsVelEnvelopeMin04

Minimalwert des "[RmsVelEnvelope](#)" auf [Seite 45](#) Kennwertes des jeweiligen Kanals in 1/1000 mm/s.

Siehe auch "[Kennwerte \(Minimal und Maximal\)](#)" auf [Seite 51](#) für weitere Informationen.

Dieser Wert wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "[SensorConfig](#)" auf [Seite 29](#)) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.22 RmsVelRawMin

Name:

RmsVelRawMin01 bis RmsVelRawMin04

Minimalwert des "RmsVelRaw" auf Seite 45 Kennwertes des jeweiligen Kanals in 1/1000 mm/s.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Dieser Wert wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.23 RmsVelRawMax

Name:

RmsVelRawMax01 bis RmsVelRawMax04

Maximalwert des "RmsVelRaw" auf Seite 45 Kennwertes des jeweiligen Kanals in 1/1000 mm/s.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Dieser Wert wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.11.24 SkewnessRawMax

Name:

SkewnessRawMax01 bis SkewnessRawMax04

Maximalwert des "SkewnessRaw" auf Seite 50 Kennwertes des jeweiligen Kanals in 1/1000.

Da im Modul eine Division durch den Effektivwert ("RmsRaw" auf Seite 45) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
DINT	-2.147.483.648 bis 2.147.483.647

4.11.25 SkewnessRawMin

Name:

SkewnessRawMin01 bis SkewnessRawMin04

Minimalwert des "SkewnessRaw" auf Seite 50 Kennwertes des jeweiligen Kanals in 1/1000.

Da im Modul eine Division durch den Effektivwert ("RmsRaw" auf Seite 45) erfolgt kann es zu einem Überlauf kommen, wenn dieser sehr klein ist. Um dies zu vermeiden wird dieser Wert modulintern auf mindestens 1 mg begrenzt.

Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 für weitere Informationen.

Datentyp	Werte
DINT	-2.147.483.648 bis 2.147.483.647

4.12 Frequenzband-Konfigurationregister

4.12.1 HighFrequencyConfig

Name:

HighFrequencyConfig01

Register zum Einstellen des Hochpasses für das Hüllkurvensignal und den Kennwerten "[PeakHighFrequency](#)" auf Seite 42 und "[RmsHighFrequency](#)" auf Seite 44. Die Einstellung betrifft das gesamte Modul.

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	1

Bitstruktur:

Bit	Beschreibung	Werte	Information
0 - 3	Hochpasskonfiguration für das gesamte Modul	0	2000 Hz
		1	1000 Hz (Bus Controller Default)
		2	500 Hz
		3 bis 15	Ungültig
4 - 15	Reserviert	-	

4.12.2 HighFrequencyConfigRead

Name:

HighFrequencyConfig01Read

Register zum Auslesen des Hochpasses für das Hüllkurvensignal und den Kennwerten "[PeakHighFrequency](#)" auf Seite 42 und "[RmsHighFrequency](#)" auf Seite 44.

Datentyp	Werte
UINT	0 bis 65.535

4.12.3 MaxFrequencyEnvelope

Name:

MaxFrequencyEnvelope01

Register zum Einstellen der maximalen Frequenz des Kanals. Durch verringern der maximalen Frequenz kann die Frequenzauflösung im Spektrum erhöht werden.

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	0

Bitstruktur:

Bit	Beschreibung	Wert	Information
0 - 3	Maximale Frequenz für Kanal 1	0	2000 Hz (Bus Controller Default)
		1	1000 Hz
		2	500 Hz
		3	200 Hz
		4 bis 15	Ungültig
4 - 7	Maximale Frequenz für Kanal 2	x	Für mögliche Werte siehe Kanal 1
8 - 11	Maximale Frequenz für Kanal 3	x	Für mögliche Werte siehe Kanal 1
12 - 15	Maximale Frequenz für Kanal 4	x	Für mögliche Werte siehe Kanal 1

Übersicht über die maximale Frequenz des Hüllkurvensignal

Maximale Frequenz	Abtastfrequenz	Messzeit	Frequenzauflösung im Frequenzspektrum
2000 Hz	5156 Hz	1,5888 s	0,6294 Hz
1000 Hz	2578 Hz	3,1775 s	0,3147 Hz
500 Hz	1289 Hz	6,3550 s	0,1574 Hz
200 Hz	516 Hz	15,8875 s	0,0629 Hz

Zu beachten für die Konfiguration der maximalen Frequenz

- Der Frequenzbereich muss größer als die Schadensfrequenz sein, soll aber ansonsten so klein wie möglich gehalten werden, um eine gute Auflösung zu erzielen.
- Bei Benutzung von Breitbandwerten wird während einer Messung nur der größte Spitzenwert zurückgegeben. Die Verwendung einer langen Messzeit bei niedriger Frequenz kann in manchen Anwendungen dazu führen, dass Messwerte übersehen werden.
- Die maximale Frequenz beeinflusst die Größe der Abtastrate und kann durch Register "[Analog Input](#)" auf Seite 36 eingestellt werden.

4.12.4 MaxFrequencyEnvelopeRead

Name:

MaxFrequencyEnvelope01Read

Register zum Auslesen der konfigurierten maximalen Frequenz für das Hüllkurvensignal der einzelnen Kanäle.

Datentyp	Werte
UINT	0 bis 65.535

4.12.5 MaxFrequencyRaw

Name:

MaxFrequencyRaw01

Register zum Einstellen der maximalen Frequenz des Kanals. Durch verringern der maximalen Frequenz kann die Frequenzauflösung im Spektrum erhöht werden.

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	8738

Bitstruktur:

Bit	Beschreibung	Wert	Information
0 - 3	Maximale Frequenz für Kanal 1	0	10000 Hz
		1	5000 Hz
		2	2000 Hz (Bus Controller Default)
		3	1000 Hz
		4	500 Hz
		5	200 Hz
		6 bis 15	Ungültig
4 - 7	Maximale Frequenz für Kanal 2	x	Für mögliche Werte siehe Kanal 1
8 - 11	Maximale Frequenz für Kanal 3	x	Für mögliche Werte siehe Kanal 1
12 - 15	Maximale Frequenz für Kanal 4	x	Für mögliche Werte siehe Kanal 1

Übersicht über die maximale Frequenz des Rohsignals

Maximale Frequenz	Abtastfrequenz	Messzeit	Frequenzauflösung im Frequenzspektrum
10000 Hz	25781 Hz	0,3178 s	3,1471 Hz
5000 Hz	12891 Hz	0,6355 s	1,5736 Hz
2000 Hz	5156 Hz	1,5888 s	0,6294 Hz
1000 Hz	2578 Hz	3,1775 s	0,3147 Hz
500 Hz	1289 Hz	6,3550 s	0,1574 Hz
200 Hz	516 Hz	15,8875 s	0,0629 Hz

Zu beachten für die Konfiguration der maximalen Frequenz

- Der Frequenzbereich muss größer als die Schadensfrequenz sein, soll aber ansonsten so klein wie möglich gehalten werden, um eine gute Auflösung zu erzielen.
- Bei Benutzung von Breitbandwerten wird während einer Messung nur der größte Spitzenwert zurückgegeben. Die Verwendung einer langen Messzeit bei niedriger Frequenz kann in manchen Anwendungen dazu führen, dass Messwerte übersehen werden.
- Die maximale Frequenz beeinflusst die Größe der Abtastrate und kann durch Register ["Analog Input"](#) auf [Seite 36](#) eingestellt werden.

4.12.6 MaxFrequencyRawRead

Name:

MaxFrequencyRaw01Read

Register zum Auslesen der konfigurierten maximalen Frequenz für das Rohsignal der einzelnen Kanäle.

Datentyp	Werte
UINT	0 bis 65.535

4.12.7 MinFrequencyEnvelope

Name:

MinFrequencyEnvelope01

Register zur Konfiguration der niedrigsten, noch auszuwertenden Frequenz des Hüllkurvensignals der einzelnen Kanäle.

Diese Konfiguration wird für die einzelnen Kanäle nur benötigt, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist.

Bit EnableVelocityCalculation = 0

Folgende minimal möglichen Werte, abhängig von der maximalen Frequenz ("MaxFrequencyEnvelope" auf Seite 57), werden verwendet. Die Tabelle zeigt die minimale MinFrequencyEnvelope in Abhängigkeit von MaxFrequencyEnvelope:

Maximale Frequenz	Frequenzauflösung im Frequenzspektrum	Minimale Frequenz
2000 Hz	0,6294 Hz	1,888 Hz
1000 Hz	0,3147 Hz	0,944 Hz
500 Hz	0,1574 Hz	0,472 Hz
200 Hz	0,0629 Hz	0,188 Hz

Bit EnableVelocityCalculation = 1

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	0

Bitstruktur:

Bit	Beschreibung	Wert	Information
0 - 3	Niedrigste Frequenz für Kanal 1	0	10 Hz (Bus Controller Default)
		1	5 Hz
		2	2 Hz
		3	1 Hz
		4	0,5 Hz
		5 bis 15	Ungültig
4 - 7	Niedrigste Frequenz für Kanal 2	x	Für mögliche Werte siehe Kanal 1
8 - 11	Niedrigste Frequenz für Kanal 3	x	Für mögliche Werte siehe Kanal 1
12 - 15	Niedrigste Frequenz für Kanal 4	x	Für mögliche Werte siehe Kanal 1

Information:

Wird für einen Kanal eine Frequenz konfiguriert, die kleiner ist als die minimale Frequenz, so wird auf diese begrenzt.

4.12.8 MinFrequencyEnvelopeRead

Name:

MinFrequencyEnvelope01Read

Register zum Auslesen der niedrigsten, noch auszuwertenden Frequenz des Hüllkurvensignals der einzelnen Kanäle.

Datentyp	Werte
UINT	Siehe Bitstruktur

Bitstruktur:

Bit	Beschreibung	Wert	Information
0 - 3	Niedrigste Frequenz für Kanal 1	0	10 Hz
		1	5 Hz
		2	2 Hz
		3	1 Hz
		4	0,5 Hz
		5 bis 14	Ungültig
		15	Auf niedrigste Frequenz begrenzt
4 - 7	Niedrigste Frequenz für Kanal 2	x	Für mögliche Werte siehe Kanal 1
8 - 11	Niedrigste Frequenz für Kanal 3	x	Für mögliche Werte siehe Kanal 1
12 - 15	Niedrigste Frequenz für Kanal 4	x	Für mögliche Werte siehe Kanal 1

4.12.9 MinFrequencyRaw

Name:

MinFrequencyRaw01

Register zur Konfiguration der niedrigsten, noch auszuwertenden Frequenz des Rohsignales der einzelnen Kanäle.

Diese Konfiguration wird für die einzelnen Kanäle nur benötigt, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "[SensorConfig](#)" auf Seite 29) des entsprechenden Kanals gesetzt ist.

Bit EnableVelocityCalculation = 0

Folgende minimal mögliche Werte, abhängig von der maximalen Frequenz ("[MaxFrequencyRaw](#)" auf Seite 58), werden verwendet. Die Tabelle zeigt die minimale MinFrequencyRaw in Abhängigkeit von MaxFrequencyRaw:

Maximale Frequenz	Frequenzauflösung im Frequenzspektrum	Minimale Frequenz
10000 Hz	3,1471 Hz	9,441 Hz
5000 Hz	1,5736 Hz	4,720 Hz
2000 Hz	0,6294 Hz	1,888 Hz
1000 Hz	0,3147 Hz	0,944 Hz
500 Hz	0,1574 Hz	0,472 Hz
200 Hz	0,0629 Hz	0,188 Hz

Bit EnableVelocityCalculation = 1

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	0

Bitstruktur:

Bit	Beschreibung	Wert	Information
0 - 3	Niedrigste Frequenz für Kanal 1	0	10 Hz (Bus Controller Default)
		1	5 Hz
		2	2 Hz
		3	1 Hz
		4	0,5 Hz
		5 bis 15	Ungültig
4 - 7	Niedrigste Frequenz für Kanal 2	x	Für mögliche Werte siehe Kanal 1
8 - 11	Niedrigste Frequenz für Kanal 3	x	Für mögliche Werte siehe Kanal 1
12 - 15	Niedrigste Frequenz für Kanal 4	x	Für mögliche Werte siehe Kanal 1

Information:

Wird für einen Kanal eine Frequenz konfiguriert, die kleiner ist als die minimale Frequenz, so wird auf diese begrenzt.

4.12.10 MinFrequencyRawRead

Name: MinFrequencyRaw01Read

Register zum Auslesen der niedrigsten, noch auszuwertenden Frequenz des Rohsignales der einzelnen Kanäle.

Datentyp	Werte
UINT	Siehe Bitstruktur

Bitstruktur:

Bit	Beschreibung	Wert	Information
0 - 3	Niedrigste Frequenz für Kanal 1	0	10 Hz
		1	5 Hz
		2	2 Hz
		3	1 Hz
		4	0,5 Hz
		5 bis 14	Ungültig
		15	Auf niedrigste Frequenz begrenzt
4 - 7	Niedrigste Frequenz für Kanal 2	x	Für mögliche Werte siehe Kanal 1
8 - 11	Niedrigste Frequenz für Kanal 3	x	Für mögliche Werte siehe Kanal 1
12 - 15	Niedrigste Frequenz für Kanal 4	x	Für mögliche Werte siehe Kanal 1

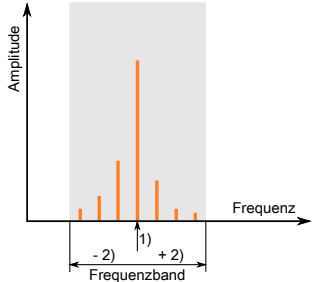
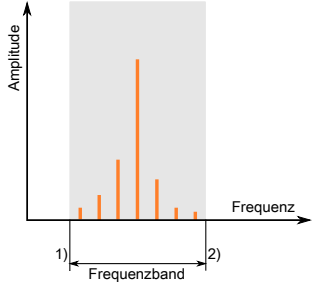
4.13 Frequenzbänder

Für eine frühzeitige Schadens- bzw. Problemerkennung ist oft eine Überwachung einzelner Frequenzbänder sinnvoll. Als Kennwert für diese Frequenzbänder kann ein selektiver Effektivwert eingesetzt werden.

Dabei kann ausgewählt werden zwischen

- "Breitband Effektivwert" auf Seite 62
- "Drehzahlabhängiger Effektivwert" auf Seite 63
- "Rauschen" auf Seite 63

Es können insgesamt bis zu 32 verschiedene Frequenzbänder definiert werden. Es ist darauf zu achten, dass abhängig von der Konfiguration das Format etwas variiert.

Drehzahlabhängiger Effektivwert	
	Frequenzbandbreite wird aus der Frequenz der Drehzahl $\pm$ der Frequenzabstand gebildet 1) Frequenz der Drehzahl: <code>FrequencyBandxxDmgFreq60Rpm * ActSpeed</code> 2) Frequenzabstand: <code>FrequencyBandTolerance</code>
Drehzahlunabhängiger Effektivwert	
	Frequenzbandbreite wird durch Angabe der unteren und oberen Frequenzbandgrenze gebildet. 1) Untere Frequenzbandgrenze: <code>FrequencyBandLowerFrequency</code> 2) Obere Frequenzbandgrenze: <code>FrequencyBandUpperFrequency</code>

Durch die Einschränkung des Frequenzbereiches können bestimmte Fehlerfälle gut identifiziert werden.

Als Beispiel sei hier Erkennung der Unwucht genannt. Diese zeigt sich deutlich in einem Anstieg der Frequenzlinie der Drehzahl. Mit der Bildung eines selektiven Kennwertes um die Drehzahlfrequenz kann die Zuordnung verbessert werden.

Auch für Effektivwerte gebildet aus der Hüllkurve können selektive Kennwerte gebildet werden.

Ein Außenringschaden zeigt sich verstärkt im Anwachsen der sogenannten Lagerschadensfrequenzen. Bei einem Außenringschaden ist dies das Anwachsen des Anteils der Außenringschadensfrequenz.

Diese Lagerschadensfrequenzen werden in der Regel von den Lagerherstellern zur Verfügung gestellt.

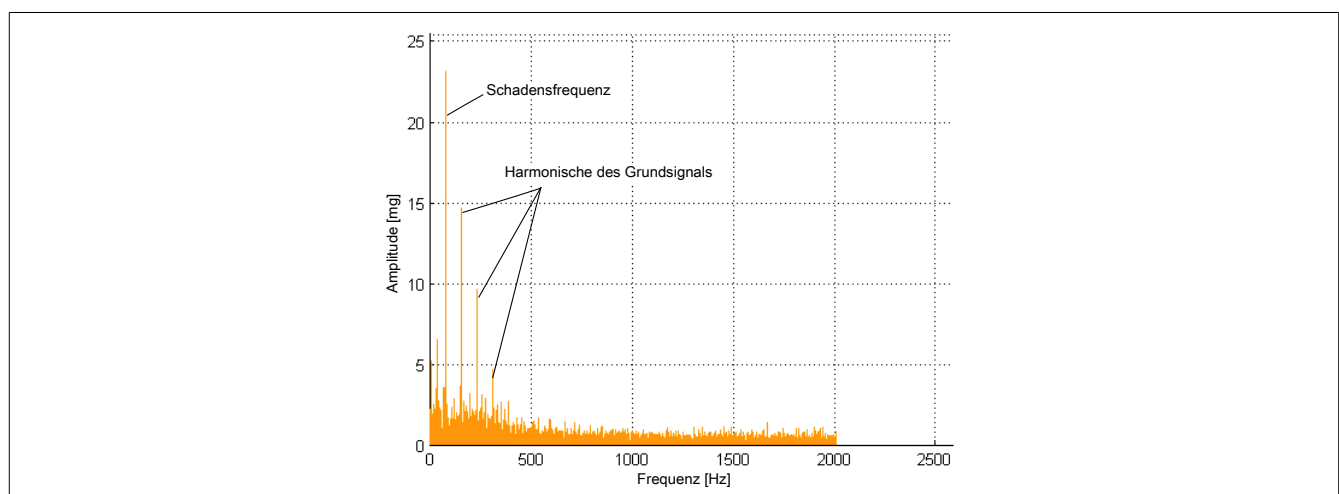


Abbildung 4: Selektiver Kennwert für einen Außenringschaden

Über die Parametrierung eines Kennwertes mit einem Frequenzbereich rund um die Schadensfrequenz kann damit ein Wälzlagerschaden am Außenring frühzeitig erkannt werden.

Werden Frequenzen außerhalb der Minimal- und Maximalfrequenz des Signals des ausgewählten Kanals eingegeben, so werden nur die Bereiche zwischen der Minimal- und Maximalfrequenz ausgewertet.

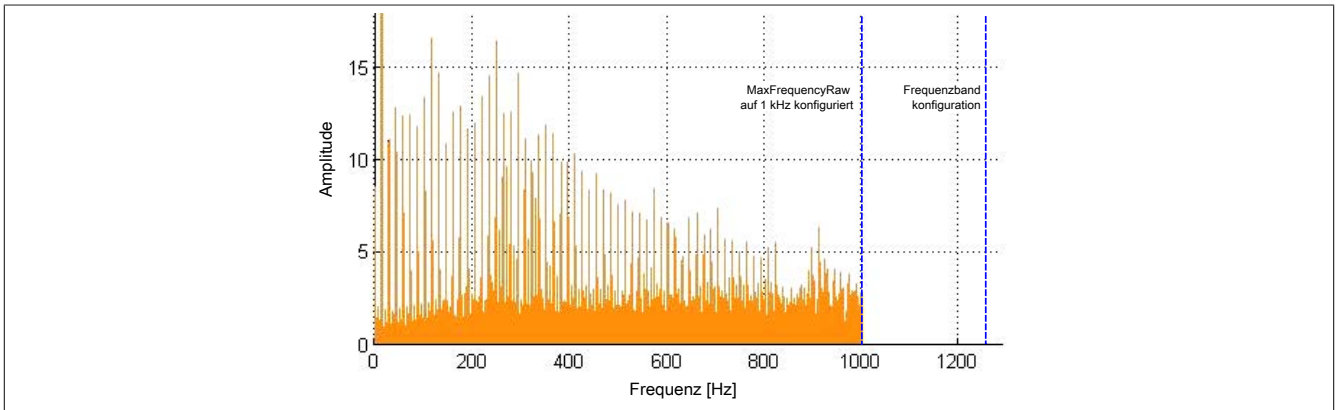


Abbildung 5: Begrenzung der Frequenzbandauswertung

Die beiden benachbarten Linien (Abtastungen) im Spektrum, welche bereits außerhalb des eingestellten Fensters liegen, d. h. einer oberhalb und einer unterhalb des Fensters, werden dabei, je nach Entfernung zum Fenster, noch anteilmäßig mitberechnet.

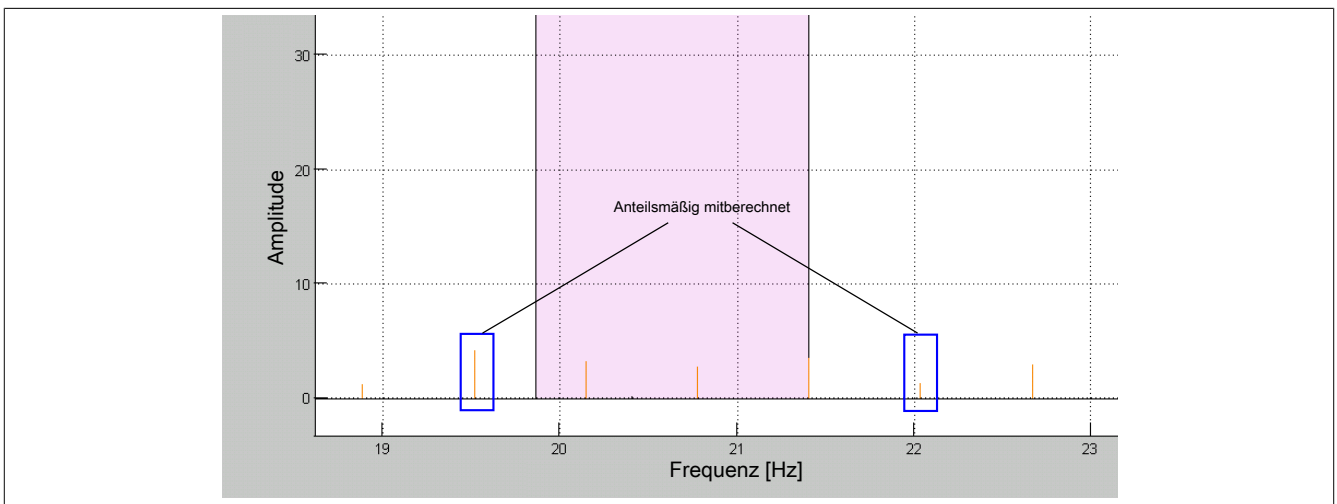


Abbildung 6: Anteilmäßige Mitberechnung seitlicher Linien

4.13.1 Breitband Effektivwert

In dieser Konfiguration wird der Effektivwert des eingestellten Signals und Kanals im Frequenzband berechnet. Die Berechnung erfolgt im Bereich der eingestellten Minimalfrequenz ("[FrequencyBandXXLowerFrequency](#)" auf Seite 66) bis zur eingestellten Maximalfrequenz ("[FrequencyBandXXUpperFrequency](#)" auf Seite 67). Die Minimal- und Maximalfrequenz kann dabei mit einer Genauigkeit von 0,25 Hz eingegeben werden.

Es kann für jedes Frequenzband jeder Kanal ausgewählt werden. Folgende Signale stehen pro Kanal zur Auswahl:

- Rohsignal Beschleunigung
- Rohsignal Geschwindigkeit. Ist gleich 0 wenn Geschwindigkeitsberechnung deaktiviert ist.
- Hüllkurvensignal Beschleunigung
- Hüllkurvensignal Geschwindigkeit. Ist gleich 0 wenn Geschwindigkeitsberechnung deaktiviert ist.

Die harmonischen Frequenzen (ganzzahlige Vielfache) des Fensters können ebenfalls mitberechnet werden. Hierbei wird einfach die Breite des Fensters beibehalten und die mittlere Frequenz des Fensters immer multipliziert (*1, *2, *3, etc.) bis die maximale Frequenz des eingestellten Signals und Kanals erreicht wird.

4.13.2 Drehzahlabhängiger Effektivwert

In dieser Konfiguration wird der Effektivwert in einem beweglichen Fenster berechnet. Dafür gibt es 4 Geschwindigkeitseingänge (siehe Register "[ActSpeed](#)" auf Seite 27). Für jedes der 32 Frequenzbänder kann jeweils eine der 4 Geschwindigkeiten ausgewählt werden. Zusätzlich muss die normierte Schadfrequenz bei 60 rpm ("[FrequencyBandXXDmgFreq60rpm](#)" auf Seite 66) und eine Toleranz ("[FrequencyBandXXTolerance](#)" auf Seite 66) konfiguriert werden. Diese können für jedes Frequenzband extra konfiguriert werden.

Das Fenster, in dem der Effektivwert berechnet wird, ergibt sich wie folgt:

Minimalfrequenz = (Geschwindigkeit * normierte Schadfrequenz bei 60 rpm) - Toleranz

Maximalfrequenz = (Geschwindigkeit * normierte Schadfrequenz bei 60 rpm) + Toleranz

Die normierte Schadfrequenz und Toleranz kann dabei mit einer Genauigkeit von 0,01 Hz eingegeben werden.

Folgende Signale stehen pro Kanal zur Auswahl:

- Rohsignal Beschleunigung
- Rohsignal Geschwindigkeit. Ist gleich 0 wenn Geschwindigkeitsberechnung deaktiviert ist.
- Hüllkurvensignal Beschleunigung
- Hüllkurvensignal Geschwindigkeit. Ist gleich 0 wenn Geschwindigkeitsberechnung deaktiviert ist.

Die harmonischen Frequenzen (ganzzahlige Vielfache) des Fensters können ebenfalls mitberechnet werden. Hierbei wird einfach die Breite des Fensters beibehalten und die mittlere Frequenz des Fensters immer multipliziert (*1, *2, *3, usw.) bis die maximale Frequenz des eingestellten Signals und Kanals erreicht wird.

Information:

Wird ein festes Frequenzband benötigt, bei dem man Minimalfrequenz ("[FrequencyBandXXLowerFrequency](#)" auf Seite 66) und Maximalfrequenz ("[FrequencyBandXXUpperFrequency](#)" auf Seite 67) genauer als 0,25 Hz einstellen muss, so kann dafür ein drehzahlabhängiges Frequenzband verwendet werden, bei dem die Drehzahl fest eingestellt ist.

4.13.3 Rauschen

In dieser Konfiguration wird im Frequenzband das Rauschen eines Quadranten des jeweiligen Signals des ausgewählten Kanals berechnet.

Dafür wird die konfigurierte Maximalfrequenz (Register "[MaxFrequencyEnvelope](#)" auf Seite 57 und "[MaxFrequencyRaw](#)" auf Seite 58) des Signals des ausgewählten Kanals durch 4 geteilt. Dadurch entstehen 4 Quadranten. Mit Hilfe der Konfiguration kann ausgewählt werden in welchem dieser 4 Quadranten das Rauschen ermittelt werden soll.

Folgende Signale stehen pro Kanal zur Auswahl:

- Rohsignal Beschleunigung
- Rohsignal Geschwindigkeit. Ist gleich 0 wenn Geschwindigkeitsberechnung deaktiviert ist.
- Hüllkurvensignal Beschleunigung
- Hüllkurvensignal Geschwindigkeit. Ist gleich 0 wenn Geschwindigkeitsberechnung deaktiviert ist.

Mit Hilfe dieser Konfiguration kann man z. B. Gleitvorgänge gut messen. Je mehr Reibung vorhanden ist, desto höher ist das Rauschen.

4.13.4 Konfiguration

Jedes der 32 Frequenzbänder kann auf einen der folgenden Kennwerte konfiguriert werden.

Kennwert	Beschreibung
FrequencyBandRmsAccEnvelope	Der Effektivwert gebildet aus dem Hüllkurvensignal der Schwingbeschleunigung in einem frei wählbaren Frequenzbereich.
FrequencyBandRmsVelEnvelope	Der Effektivwert gebildet aus dem Hüllkurvensignal der Schwinggeschwindigkeit in einem frei wählbaren Frequenzbereich. Dieses Frequenzband wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.
FrequencyBandRmsAccRaw	Der Effektivwert gebildet aus dem Rohsignal der Schwingbeschleunigung in einem frei wählbaren Frequenzbereich.
FrequencyBandRmsVelRaw	Der Effektivwert gebildet aus dem Rohsignal der Schwinggeschwindigkeit in einem frei wählbaren Frequenzbereich.
FrequencyBandNoiseAccEnvelope	Der Rauschwert gebildet aus dem Hüllkurvensignal der Schwingbeschleunigung in einem frei wählbaren Quadranten.
FrequencyBandNoiseVelEnvelope	Der Rauschwert gebildet aus dem Hüllkurvensignal der Schwinggeschwindigkeit in einem frei wählbaren Quadranten. Dieses Frequenzband wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.
FrequencyBandNoiseAccRaw	Der Rauschwert gebildet aus dem Rohsignal der Schwingbeschleunigung in einem frei wählbaren Quadranten.
FrequencyBandNoiseVelRaw	Der Rauschwert gebildet aus dem Rohsignal der Schwinggeschwindigkeit in einem frei wählbaren Quadranten. Dieses Frequenzband wird nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Berechnung der Gesamtbreite des Frequenzbandes

Die gemessene Schadfrequenz kann geringfügig von der erwarteten Frequenz abweichen. Es muss daher sichergestellt werden, dass das Frequenzband zur Überwachung der Schadfrequenz breit genug ist, um diese Abweichung auszugleichen.

Als maximal zulässige Abweichung wurde 1/2% der maximalen Antriebsfrequenz festgelegt. Da die Abweichung sowohl in positive als auch in negative Richtung auftreten kann, muss diese Abweichung doppelt eingerechnet werden. Daraus ergibt sich die Formel:

$$\text{Gesamtbreite des Frequenzbandes} = 2 * 1/2\% * \text{maximaler Antriebsfrequenz [Hz]}$$

Abhängig von der gewählten MaxFrequency sollten wenigstens 3 Linien innerhalb jedes Frequenzbandes konfiguriert werden.

Die ersten Linien links und rechts vom konfigurierten Frequenzband werden abhängig von der Entfernung zum Frequenzband prozentuell mitberechnet (siehe Abbildung in "Frequenzbänder" auf Seite 61).

Beispiel

Bei einer maximalen Antriebsfrequenz von 50 Hz soll ein Frequenzband bei 10 Hz eingestellt werden.
 $10 \text{ Hz} \pm (0,005 * 50 \text{ Hz}) = \text{mindestens } 9,75 \text{ Hz untere und } 10,25 \text{ Hz obere Frequenzbandgrenze}$

4.13.5 FrequencyBandMax

Name:

FrequencyBandMax01 bis FrequencyBandMax32

Maximalwert des jeweiligen Frequenzbandes in 1/1000 mm/s oder g je nach Konfiguration. Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 und "Frequenzbänder" auf Seite 26 für weitere Informationen.

Ist das Frequenzband auf ein Geschwindigkeitssignal parametrisiert, so wird dieser Wert nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.13.6 FrequencyBandMin

Name:

FrequencyBandMin01 bis FrequencyBandMin32

Minimalwert des jeweiligen Frequenzbandes in 1/1000 mm/s oder g je nach Konfiguration. Siehe auch "Kennwerte (Minimal und Maximal)" auf Seite 51 und "Frequenzbänder" auf Seite 26 für weitere Informationen.

Ist das Frequenzband auf ein Geschwindigkeitssignal parametrisiert, so wird dieser Wert nur berechnet, wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

Datentyp	Werte
UDINT	0 bis 4.294.967.295

4.13.7 FrequencyBandConfig

Name:

FrequencyBand01Config bis FrequencyBand32Config

Allgemeine Konfiguration der einzelnen Frequenzbänder.

Jedes Frequenzband kann auf jedem Kanal mit jedem der 4 Geschwindigkeitsdatenpunkte ("ActSpeed" auf Seite 27) berechnet werden.

Datentyp	Werte	Bus Controller Default
UINT	Siehe Bitstruktur	0

Bitstruktur:

Bit	Beschreibung	Wert	Information
0 - 3	Eingangskanal für die Berechnung des Frequenzbandes	0	Kanal 1 (Bus Controller Default)
		1	Kanal 2
		2	Kanal 3
		3	Kanal 4
		4 bis 15	Ungültig
4 - 5	Signals zur Berechnung des Frequenzbandes	0	Rohsignal Geschwindigkeit ¹⁾ (Bus Controller Default)
		1	Hüllkurvensignal Geschwindigkeit ¹⁾
		2	Rohsignal Beschleunigung
		3	Hüllkurvensignal Beschleunigung
6	Geschwindigkeitsabhängige Berechnung des Frequenzbandes des konfigurierten ActSpeed Datenpunktes	0	Aus Für die Berechnung wird "FrequencyBandLowerFrequency" auf Seite 66 und "FrequencyBandUpperFrequency" auf Seite 67 verwendet (Bus Controller Default)
		1	Ein Für die Berechnung wird "FrequencyBandDmgFreq60rpm" auf Seite 66 und "FrequencyBandTolerance" auf Seite 66 verwendet
7	Harmonische Frequenzen mitberechnen	0	Aus (Bus Controller Default)
		1	Ein Alle Harmonischen bis zur maximalen Frequenz werden immer mitberechnet. Die Harmonischen können auch Frequenzanteile enthalten, welche von Schäden an anderer Stelle verursacht werden. Dadurch kann es zu einer Fehlinterpretation der Messung kommen.
8 - 11	Verwendeter ActSpeed Datenpunkt für die Berechnung, wenn das Frequenzband geschwindigkeitsabhängig berechnet wird.	0	ActSpeed01 (Bus Controller Default)
		1	ActSpeed02
		2	ActSpeed03
		3	ActSpeed04
		4 bis 15	Ungültig
12	Berechnung des Rauschens anstatt des RMS aktivieren	0	Aus (Bus Controller Default)
		1	Ein Die Einstellungen für Geschwindigkeitsabhängig und Harmonische werden ignoriert.
13 - 14	Auswahl des Quadranten zur Berechnung des Rauschens.	0	1. Quadrant von MinFrequency bis 1/4 MaxFrequency ²⁾ (Bus Controller Default)
		1	2. Quadrant von 1/4 MaxFrequency bis 1/2 MaxFrequency ²⁾
		2	3. Quadrant von 1/2 MaxFrequency bis 3/4 MaxFrequency ²⁾
		3	4. Quadrant von 3/4 MaxFrequency bis MaxFrequency ²⁾
15	Reserviert	0	

1) Wird nur berechnet wenn das Bit EnableVelocityCalculation (Konfiguration im Register "SensorConfig" auf Seite 29) des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben.

2) Bezieht sich auf das jeweilige Signal (Rohsignal oder Hüllkurvensignal) des Kanals.

4.13.8 FrequencyBandConfigRead

Name:

FrequencyBand01ConfigRead bis FrequencyBand32ConfigRead

Register zum Auslesen der Konfiguration der einzelnen Frequenzbänder.

Datentyp	Werte
UINT	0 bis 65.535

4.13.9 FrequencyBandDmgFreq60rpm

Name:

FrequencyBand01DmgFreq60rpm bis FrequencyBand32DmgFreq60rpm

Normierte Schadfrequenz bei 60 rpm, wenn das Frequenzband drehzahlabhängig konfiguriert ist.

Diese wird im Modul mit dem konfigurierten Geschwindigkeitsdatenpunkt multipliziert, um das Frequenzband zu berechnen.

Die normierte Schadfrequenz muss in 1/100 angegeben werden.

Datentyp	Werte	Information
UINT	1 bis 65.535	Bus Controller Default: 0

4.13.10 FrequencyBandDmgFreq60rpmRead

Name:

FrequencyBand01DmgFreq60rpmRead bis FrequencyBand32DmgFreq60rpmRead

Register zum Auslesen der normierten Schadfrequenz bei 60 rpm der einzelnen Frequenzbänder.

Die normierte Schadfrequenz wird in 1/100 angegeben.

Datentyp	Werte
UINT	1 bis 65.535

4.13.11 FrequencyBandTolerance

Name:

FrequencyBand01Tolerance bis FrequencyBand32Tolerance

Ist das Frequenzband drehzahlabhängig konfiguriert, so kann mit Hilfe dieses Datenpunktes die Breite des Frequenzbandes angegeben werden.

Zu der berechneten Schadfrequenz aus [ActSpeed](#) und [FrequencyBandDmgFreq60rpm](#) wird die [FrequencyBandTolerance](#) einmal subtrahiert, um die untere Frequenz des Frequenzbandes zu erhalten, und einmal addiert um die obere Frequenz des Frequenzbandes zu erhalten.

Die Toleranz muss in 1/100 Hz angegeben werden.

Datentyp	Werte	Information
UINT	1 bis 65.535	Bus Controller Default: 0

4.13.12 FrequencyBandToleranceRead

Name:

FrequencyBand01ToleranceRead bis FrequencyBand32ToleranceRead

Register zum Auslesen der Toleranz der einzelnen Frequenzbänder.

Die Toleranz wird in 1/100 Hz angegeben.

Datentyp	Werte
UINT	1 bis 65.535

4.13.13 FrequencyBandLowerFrequency

Name:

FrequencyBand01LowerFrequency bis FrequencyBand32LowerFrequency

Minimale Frequenz für die Berechnung des Frequenzbandes, wenn das Frequenzband nicht drehzahlabhängig ist.

Die minimale Frequenz muss in 1/4 Hz angegeben werden.

Datentyp	Werte	Information
UINT	1 bis 40.000	Bus Controller Default: 0

4.13.14 FrequencyBandLowerFrequencyRead

Name:

FrequencyBand01LowerFrequencyRead bis FrequencyBand32LowerFrequencyRead

Register zum Auslesen der minimalen Frequenz der einzelnen Frequenzbänder.

Die minimale Frequenz wird in 1/4 Hz angegeben.

Datentyp	Werte
UINT	1 bis 40.000

4.13.15 FrequencyBandUpperFrequency

Name:

FrequencyBand01UpperFrequency bis FrequencyBand32UpperFrequency

Maximale Frequenz für die Berechnung des Frequenzbandes, wenn das Frequenzband nicht drehzahlabhängig ist.

Die maximale Frequenz muss in 1/4 Hz angegeben werden.

Datentyp	Werte	Information
UINT	1 bis 40.000	Bus Controller Default: 0

4.13.16 FrequencyBandUpperFrequencyRead

Name:

FrequencyBand01UpperFrequencyRead bis FrequencyBand32UpperFrequencyRead

Register zum Auslesen der maximalen Frequenz der einzelnen Frequenzbänder.

Die maximale Frequenz wird in 1/4 Hz angegeben.

Datentyp	Werte
UINT	1 bis 40.000

4.14 Flatstream

4.14.1 Kennwertübertragung via Flatstream

Im Funktionsmodell "Standard" übernimmt das Automation Runtime die Flatstream-Kommunikation für die Kennwertübertragung. Alle Kennwerte werden dem Anwender automatisch und bereits richtig skaliert zur Verfügung gestellt. Deshalb werden die Register für den Kennwerte-Flatstream im [Funktionsmodell 0 - Standard](#) nicht in der I/O-Zuordnung angezeigt.

Bei Verwendung des Flatstreams im [Funktionsmodell 0 - Standard](#) und [Funktionsmodell 1 - Schneller Master](#) sollte folgendes beachtet werden:

- Die maximale Buszykluszeit soll 10 ms nicht überschreiten.
- Der Taskzyklus ist entweder gleich schnell und synchron oder schneller als der Bus am Modul.

Buszykluszeit

Da die Übertragung der Kennwerte über den Flatstream erfolgt, soll eine maximale Buszykluszeit von 10 ms nicht überschritten werden, da ansonsten die vom Modul alle 300 ms berechneten Daten nicht vollständig am Bus übertragen werden können. Sollte die aktuelle Übertragung noch nicht abgeschlossen sein, wenn neue Kennwerte vorliegen, so werden diese verworfen. Um lückenlos messen zu können ist somit eine Zykluszeit von ≤ 10 ms erforderlich.

Taskzykluszeit

Es ist zu beachten, dass das Modul normalerweise jeden X2X Link-Zyklus die Werte im Flatstream ändert. Aus Gründen der Performanz und um sicherzustellen, dass im Task jeder Wert erfasst wird, soll der Taskzyklus so gewählt werden, dass er entweder gleich schnell und damit synchron oder schneller als der Bus am Modul ist. Dies gilt auch für Feldbusverbindungen zwischen CPU und Bus Controller. Ist dies aus diversen Gründen nicht möglich, so gibt es die Option das ForwardDelay des Flatstreams zu verändern (siehe ["Die Forward-Funktion beim X20CM4810"](#) auf Seite 70).

4.14.1.1 Register für den Kennwert-Flatstream

Folgende Register werden für die Kennwertübertragung benötigt:

["ParameterInputSequence01"](#) auf Seite 77

["ParameterRxByte01-13"](#) auf Seite 75

["ParameterOutputSequence01"](#) auf Seite 76

Daraus ergibt sich eine [InputMTU](#) von 13 und eine [OutputMTU](#) von 0 Byte für den Kennwert-Flatstream.

4.14.1.2 Aufbau des Kennwert-Streams

Jeder Kennwert, welcher im Stream vom Modul übertragen wird, hat eine Länge von 3 Byte und wird mit dem Highbyte zuerst übertragen. Für das genaue Format siehe ["Kennwerte" auf Seite 41](#). Der Stream ist insgesamt 240 Byte lang. Als erstes werden jeweils die gesamten Kennwerte von Kanal 1, danach Kanal 2, 3 und 4 übertragen. Anschließend befinden sich noch die 32 Frequenzbänder im Stream.

Die Kennwerte RmsVelRaw, RmsVelEnvelope sowie FrequencyBandxx werden nur übertragen, wenn sie auf Geschwindigkeit parametrisiert sind und das Bit EnableVelocityCalculation des entsprechenden Kanals gesetzt ist. Ansonsten wird 0 ausgegeben. EnableVelocityCalculation wird im Register ["SensorConfig" auf Seite 29](#) konfiguriert.

Byte Offset im Stream				Kennwert
Kanal 1	Kanal 2	Kanal 3	Kanal 4	
Parameter der Kanäle				
0	36	72	108	RmsAccRaw
3	39	75	111	RmsVelRaw
6	42	78	114	RmsAccEnvelope
9	45	81	117	RmsVelEnvelope
12	48	84	120	PeakHighFrequency
15	51	87	123	RmsHighFrequency
18	54	90	126	Iso10816
21	57	93	129	CrestFactorRaw
24	60	96	132	KurtosisRaw
27	63	99	135	PeakRaw
30	66	102	138	SkewnessRaw
33	69	105	141	RmsRaw
Frequenzband 1 - 32				
144				FrequencyBand01
:				:
237				FrequencyBand32

4.14.2 Pufferupload via Flatstream

Für die Funktionsweise des Flatstreams siehe ["Die Flatstream-Kommunikation" auf Seite 71](#).

Der Flatstream für den Pufferupload wird vom Modul im [Funktionsmodell 0 - Standard](#) und [Funktionsmodell 1 - Schneller Master](#) bereitgestellt.

Zum Hochladen der Puffer vom Modul steht die Bibliothek **AsIOVib** zur Verfügung. Für eine Beschreibung der Bibliothek siehe ["Automation Help - Programmierung - Bibliotheken - Direkter I/O Zugriff - AsIOVIB"](#).

Die Rohdatenpuffer (Rohsignal und FFT) vom Modul werden auf Anforderung (Puffer-Flatstream Tx-Richtung) vom Modul über den Puffer-Flatstream (Rx-Richtung) übertragen, jedoch nur, wenn die Puffer zuvor am Modul mit den Einstellungen RequestBufferLock01 = 1 und BufferLockValid01 = 1 eingefroren wurden. Es ist darauf zu achten, dass die Datenpunkte des Puffer-Flatstreams vom Modul mit jedem X2X Link-Zyklus aktualisiert werden. Nach Abschluss der Übertragung muss in der SPS der Puffer vom Modul aus dem Stream "zusammengebaut" werden.

4.14.2.1 Register für den Puffer-Flatstream

Folgende Register werden für den Pufferupload benötigt:

["BufferInputSequence01" auf Seite 77](#)

["BufferRxByte01-05" auf Seite 75](#)

["BufferOutputSequence01" auf Seite 76](#)

["BufferTxByte01-04" auf Seite 75](#)

Daraus ergibt sich eine [InputMTU](#) von 5 und eine [OutputMTU](#) von 4 Byte für den Puffer-Flatstream.

4.14.2.2 Ablauf des Pufferuploads

Bevor vom Modul ein Puffer hochgeladen werden kann, müssen die Puffer am Modul durch die Zuweisung RequestBufferLock01 = 1 eingefroren werden. Erst wenn BufferLockValid01 = 1 ist, sind die Puffer eingefroren und es kann mit dem Hochladen begonnen werden.

Wird ein Geschwindigkeitspuffer hochgeladen so beinhaltet dieser nur Werte, wenn das Bit EnableVelocityCalculation des entsprechenden Kanals gesetzt ist, ansonsten wird 0 ausgegeben. Bit EnableVelocityCalculation kann im Register ["SensorConfig" auf Seite 29](#) konfiguriert werden.

Als erstes muss über den Puffer-Flatstream (Tx-Richtung) ein Puffer vom Modul angefordert werden. Dabei bedeuten:

- BufferTxByte01: Wert 0x83 (Frameende und 3 Byte gültig)
- BufferTxByte02: Der gewünschte Puffer vom Modul
- BufferTxByte03: Highbyte der Anzahl der zu lesenden Werte (pro Wert 4 Byte)
- BufferTxByte04: Lowbyte der Anzahl der zu lesenden Werte (pro Wert 4 Byte)

Anschließend wird die Sequenz um 1 erhöht. Sobald die Sequenz vom Modul bestätigt wurde ist es wichtig in BufferTxByte die Anzahl der gültigen Bytes auf 0 zu stellen, da das Modul dies ansonsten als eine erneute Anfrage interpretiert. Im Modul können bis zu 2 Anfragen hintereinander zwischengespeichert werden.

Sobald das Modul die Anfrage erhalten hat beginnt es die geforderte Anzahl von Werten vom geforderten Puffer über den Puffer-Flatstream (Rx-Richtung) zu streamen. Die Werte werden immer im Festkommaformat 16.16 (1/65536) mit dem Highbyte zuerst übertragen. Als erster Wert wird ein Skalierungsfaktor übertragen mit dem alle anderen Werte anschließend multipliziert werden müssen. Bei Zeitsignalpuffern ist der erste Wert nach dem Skalierungsfaktor immer der älteste, bei FFT-Puffern ist der erste Wert nach dem Skalierungsfaktor immer bei 0 Hz. Die FFT-Puffer sind nur von der konfigurierten MinFrequency bis zur konfigurierten MaxFrequency des jeweiligen Signals (Roh bzw. Hüllkurve) und des jeweiligen Kanals gültig.

Für den Zeit- bzw. Frequenzabstand der einzelnen Werte im Stream siehe die Register "[MinFrequencyRaw](#)" auf Seite 60 bzw. "[MinFrequencyEnvelope](#)" auf Seite 59.

RequestBufferLock01 muss während des gesamten Hochladens auf 1 sein. Nachdem die gewünschten Puffer vom Modul hochgeladen wurden kann die Sperre wieder freigegeben werden. Im Modul werden anschließend die Puffer wieder mit aktuellen Werten gefüllt und es kann nach einiger Zeit wieder erneut eingefroren und hochgeladen werden (siehe dazu Register "[Control](#)" auf Seite 28).

Puffernummer (Dez)				Puffer	Maximale Anzahl der Werte (1 Wert = 4 Byte)
Kanal 1	Kanal 2	Kanal 3	Kanal 4		
9	11	13	15	Rohsignal (Filterwert: " MaxFrequencyRaw " auf Seite 58)	8193 bzw. 65535 ¹⁾
25	27	29	31	Hüllkurvensignal (Filterwert: " MinFrequencyEnvelope " auf Seite 59)	8193 bzw. 65535 ¹⁾
66	70	74	78	FFT Amplituden Spektrum Rohsignal Geschwindigkeit (Filterwert: " MaxFrequencyRaw " auf Seite 58)	4097 ¹⁾
67	71	75	79	FFT Amplituden Spektrum Rohsignal Beschleunigung (Filterwert: " MaxFrequencyRaw " auf Seite 58)	4097 ¹⁾
82	86	90	94	FFT Amplituden Spektrum Hüllkurvensignal Geschwindigkeit (Filterwert: " MinFrequencyEnvelope " auf Seite 59)	4097 ¹⁾
83	87	91	95	FFT Amplituden Spektrum Hüllkurvensignal Beschleunigung (Filterwert: " MinFrequencyEnvelope " auf Seite 59)	4097 ¹⁾

1) Der erste Wert im Puffer ist der Skalierungsfaktor.

Für spezielle Anwendungen kann es nötig sein, dass größere Puffer hochgeladen werden müssen.

Im Register "[SensorConfig](#)" auf Seite 29 kann mit Bit 14 eine Pufferlänge von 8193 oder 65535 Werten konfiguriert werden. Dadurch ist es möglich, das Rohsignal und das Hüllkurvensignal (Puffernummer 9, 11, 13, 15, 25, 27, 29 und 31) mit 65535 Werten incl. Skalierungsfaktor vom Modul hochzuladen. Die FFT-Puffer umfassen jedoch weiterhin 4097 Werte incl. Skalierungsfaktor und beziehen sich auf die letzten 8193 Werte der 65535 Werte des Roh- bzw. Hüllkurvensignals.

Nach dem Einfrieren der Puffer für den Upload (RequestBufferLock01) bis zum nächsten Einfrieren muss mindestens so lange gewartet werden bis der langsamste Puffer wieder gefüllt ist. Wird vor Ablauf dieser Zeit versucht den Puffer erneut einzufrieren, wird dies vom Modul bis nach dem Füllen des Puffers verhindert.

4.14.3 Die Forward-Funktion beim X20CM4810

Das ForwardDelay für den Puffer-Flatstream kann im [Funktionsmodell 0 - Standard](#) und [Funktionsmodell 1 - Schneller Master](#) in der I/O-Konfiguration des Moduls azyklisch konfiguriert werden (BufferForwardDelay01). Bei Verwendung des Funktionsmodells Schneller Master kann auch das ForwardDelay für den Parameter-Flatstream in der I/O-Konfiguration azyklisch konfiguriert werden (ParameterForwardDelay01).

Das Forwarding für den Parameter- und Puffer-Flatstream kann bei Bedarf azyklisch verstellt werden (siehe Register "[ParameterForward01](#)" auf Seite 95 und "[BufferForward01](#)" auf Seite 95). Es sollte jedoch erst verstellt werden nachdem das ForwardDelay für den jeweiligen Flatstream schon eingestellt wurde.

Bei Verwendung einer SG4 CPU kann das ForwardDelay für den Parameter-Flatstream nicht manuell eingestellt werden. Ab der Automation Runtime-Version J3.09 und J4.01 wird es automatisch eingestellt bzw. bei früheren Versionen ist das ForwardDelay 0.

4.14.4 Die Flatstream-Kommunikation

4.14.4.1 Einleitung

Für einige Module stellt B&R ein zusätzliches Kommunikationsverfahren bereit. Der "Flatstream" wurde für X2X und POWERLINK Netzwerke konzipiert und ermöglicht einen individuell angepassten Datentransfer. Obwohl das Verfahren nicht unmittelbar echtzeitfähig ist, kann die Übertragung effizienter gestaltet werden als bei der zyklischen Standardabfrage.

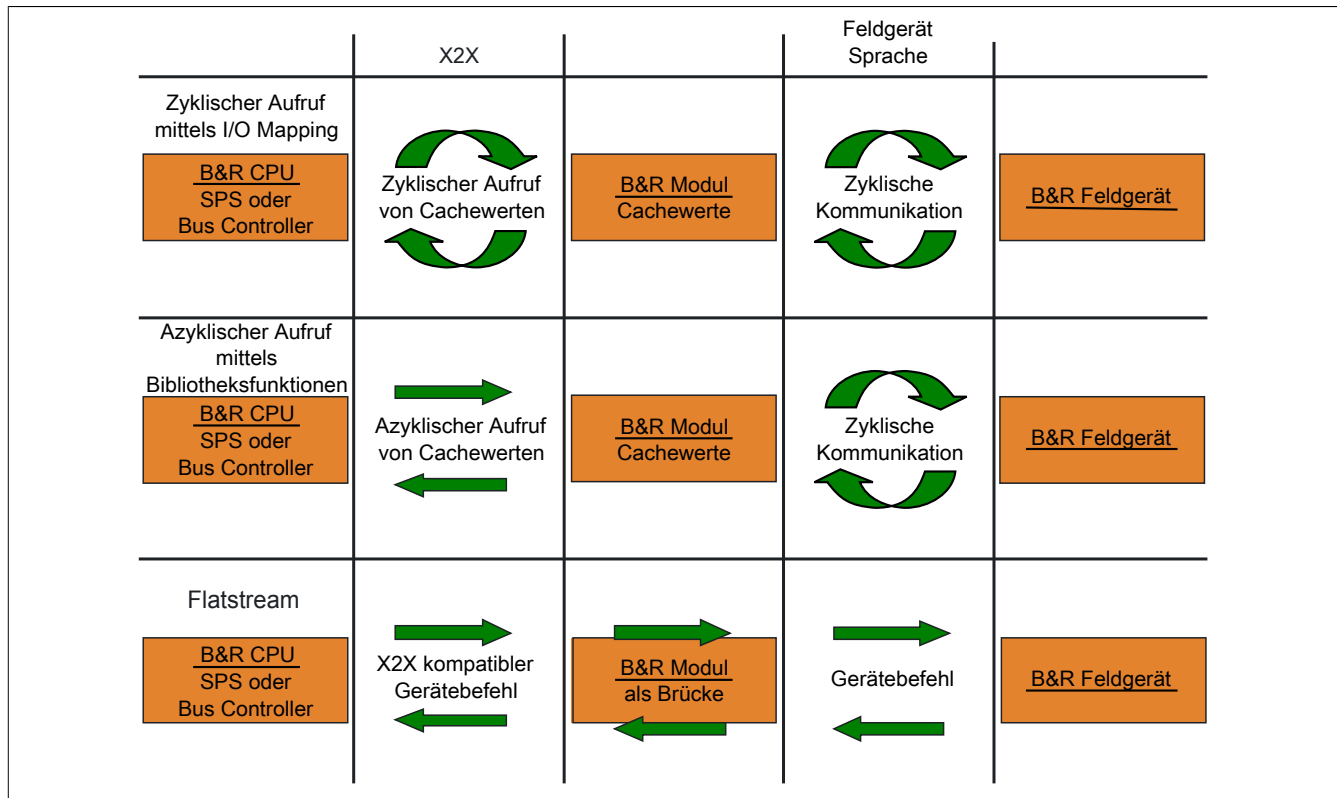


Abbildung 7: 3 Arten der Kommunikation

Durch den Flatstream wird die zyklische bzw. azyklische Abfrage ergänzt. Bei der Flatstream-Kommunikation fungiert das Modul als Bridge. Die Anfragen der CPU werden über das Modul direkt zum Feldgerät geleitet.

4.14.4.2 Nachricht, Segment, Sequenz, MTU

Die physikalischen Eigenschaften des Bussystems begrenzen die Datenmenge, die während eines Buszyklus übermittelt werden kann. Bei der Flatstream-Kommunikation werden alle Nachrichten als fortlaufender Datenstrom (engl. stream) betrachtet. Lange Datenströme müssen in mehrere Teile zerlegt und nacheinander versendet werden. Um zu verstehen wie der Empfänger die ursprüngliche Information wieder zusammensetzt, werden die Begriffe Nachricht, Segment, Sequenz und MTU unterschieden.

Nachricht

Eine Nachricht ist eine Mitteilung, die zwischen 2 Kommunikationspartnern ausgetauscht werden soll. Die Länge einer solchen Mitteilung wird durch das Flatstream-Verfahren nicht begrenzt. Es müssen allerdings modulspezifische Beschränkungen beachtet werden.

Segment (logische Gliederung einer Nachricht)

Ein Segment ist endlich groß und kann als Abschnitt der Nachricht verstanden werden. Die Anzahl der Segmente pro Nachricht ist beliebig. Damit der Empfänger die übertragenen Segmente wieder korrekt zusammensetzen kann, geht jedem Segment ein Byte mit Zusatzinformationen voraus. Das sogenannte Controlbyte enthält z. B. Informationen über die Länge eines Segments und ob das kommende Segment die Mitteilung vervollständigt. Auf diesem Weg wird der Empfänger in die Lage versetzt, den ankommenden Datenstrom korrekt zu interpretieren.

Sequenz (physikalisch notwendige Gliederung eines Segments)

Die maximale Größe einer Sequenz entspricht der Anzahl der aktivierten Rx- bzw. Tx-Bytes (später: "MTU"). Die sendende Station teilt das Sendearray in zulässige Sequenzen, die nacheinander in die MTU geschrieben, zum Empfänger übertragen und dort wieder aneinandergereiht werden. Der Empfänger legt die ankommenden Sequenzen in einem Empfangsarray ab und erhält somit ein Abbild des Datenstroms.

Bei der Flatstream-Kommunikation werden die abgesetzten Sequenzen gezählt. Erfolgreich übertragene Sequenzen müssen vom Empfänger bestätigt werden, um die Übertragung abzusichern.

MTU (Maximum Transmission Unit) - Physikalischer Transport

Die MTU des Flatstreams beschreibt die aktivierten USINT-Register für den Flatstream. Die Register können mindestens eine Sequenz aufnehmen und zum Empfänger übertragen. Für beide Kommunikationsrichtungen wird eine separate MTU vereinbart. Die OutputMTU definiert die Anzahl der Flatstream-Tx-Bytes und die InputMTU beschreibt die Anzahl der Flatstream-Rx-Bytes. Die MTUs werden zyklisch über den X2X Link transportiert, sodass die Auslastung mit jedem zusätzlich aktivierten USINT-Register steigt.

Eigenschaften

Flatstream-Nachrichten werden nicht zyklisch und nicht unmittelbar in Echtzeit übertragen. Zur Übertragung einer bestimmten Mitteilung werden individuell viele Buszyklen benötigt. Die Rx-/Tx-Register werden zwar zyklisch zwischen Sender und Empfänger ausgetauscht, aber erst weiterverarbeitet, wenn die Übernahme durch die Register "InputSequence" bzw. "OutputSequence" explizit angewiesen wird.

Verhalten im Fehlerfall (Kurzfassung)

Das Protokoll von X2X bzw. POWERLINK Netzwerken sieht vor, dass bei einer Störung die letzten gültigen Werte erhalten bleiben. Bei der herkömmlichen Kommunikation (zyklische/azyklische Abfrage) kann ein solcher Fehler in der Regel ignoriert werden.

Damit auch via Flatstream problemlos kommuniziert werden kann, müssen alle abgesetzten Sequenzen vom Empfänger bestätigt werden. Ohne die Nutzung des Forward verzögert sich die weitere Kommunikation um die Dauer der Störung.

Falls der Forward genutzt wird, erhält die Empfängerstation einen doppelt inkrementierten Sendezähler. Der Empfänger stoppt, das heißt, er schickt keine Bestätigungen mehr zurück. Anhand des SequenceAck erkennt die Sendestation, dass die Übertragung fehlerhaft war und alle betroffenen Sequenzen wiederholt werden müssen.

4.14.4.3 Prinzip des Flatstreams

Voraussetzung

Bevor der Flatstream genutzt werden kann, muss die jeweilige Kommunikationsrichtung synchronisiert sein, das heißt, beide Kommunikationspartner fragen zyklisch den SequenceCounter der Gegenstelle ab. Damit prüfen sie, ob neue Daten vorliegen, die übernommen werden müssen.

Kommunikation

Wenn ein Kommunikationspartner eine Nachricht an seine Gegenstelle senden will, sollte er zunächst ein Sendearray anlegen, das den Konventionen des Flatstreams entspricht. Auf diese Weise kann der Flatstream sehr effizient gestaltet werden, ohne wichtige Ressourcen zu blockieren.

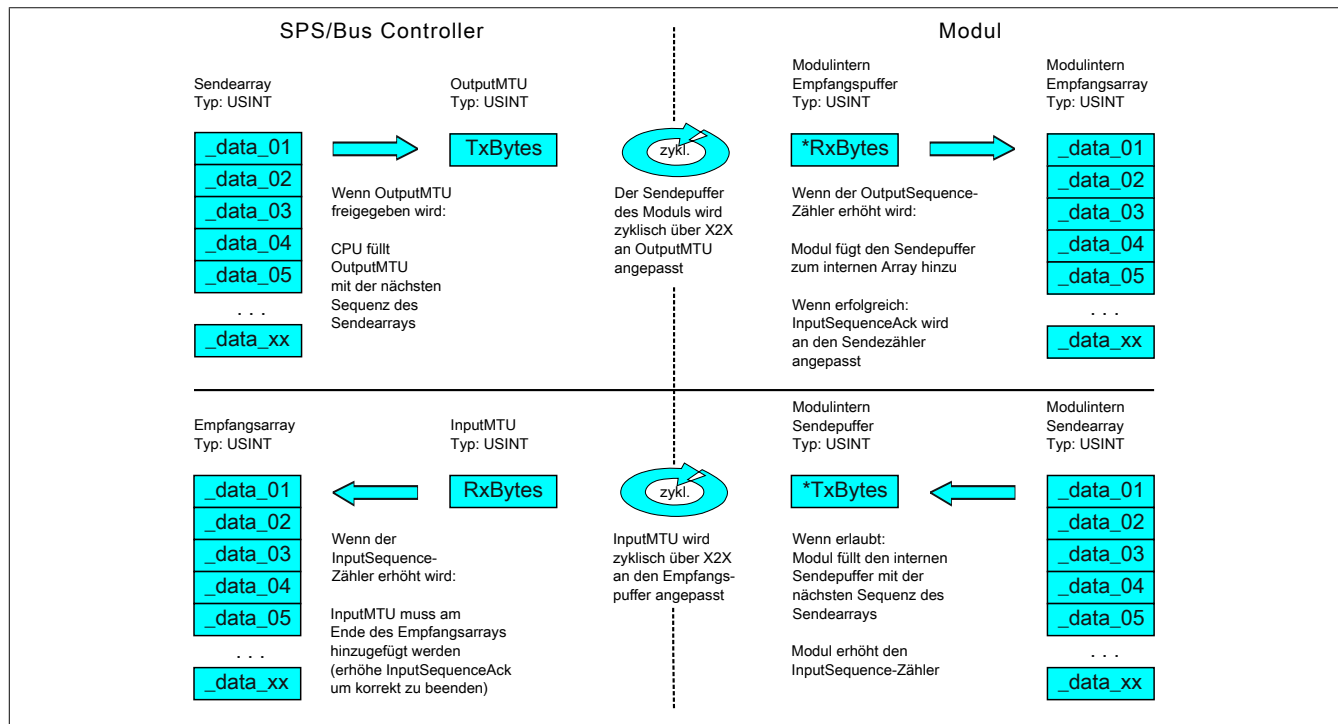


Abbildung 8: Kommunikation per Flatstream

Vorgehensweise

Als erstes wird die Nachricht in zulässige Segmente mit max. 63 Bytes aufgeteilt und die entsprechenden Controlbytes gebildet. Die Daten werden zu einem Datenstrom zusammengefügt, das heißt, je ein Controlbyte und das dazugehörige Segment im Wechsel. Dieser Datenstrom kann in das Sendearray geschrieben werden. Jedes Arrayelement ist dabei max. so groß, wie die freigegebene MTU, sodass ein Element einer Sequenz entspricht. Wenn das Array vollständig angelegt ist, prüft der Sender, ob die MTU neu befüllt werden darf. Danach kopiert er das erste Element des Arrays bzw. die erste Sequenz auf die Tx-Byte-Register. Die MTU wird zyklisch über den X2X Link zur Empfängerstation transportiert und auf den korrespondierenden Rx-Byte-Registern abgelegt. Als Signal, dass die Daten vom Empfänger übernommen werden sollen, erhöht der Sender seinen SequenceCounter. Wenn die Kommunikationsrichtung synchronisiert ist, erkennt die Gegenstelle den inkrementierten SequenceCounter. Die aktuelle Sequenz wird an das Empfangsarray angefügt und per SequenceAck bestätigt. Mit dieser Bestätigung wird dem Sender signalisiert, dass die MTU wieder neu befüllt werden kann.

Bei erfolgreicher Übertragung entsprechen die Daten im Empfangsarray exakt denen im Sendearray. Während der Übertragung muss die Empfangsstation die ankommenden Controlbytes erkennen und auswerten. Für jede Nachricht sollte ein separates Empfangsarray angelegt werden. Auf diese Weise kann der Empfänger vollständig übertragene Nachrichten sofort weiterverarbeiten.

4.14.4.4 Die Register für den Flatstream-Modus

Zur Konfiguration des Flatstreams sind 5 Register vorgesehen. Mit der Standardkonfiguration können geringe Datenmengen relativ einfach übermittelt werden.

Information:

Die CPU kommuniziert über die Register "OutputSequence" und "InputSequence" sowie den aktivierten Tx- bzw. RxBytes direkt mit dem Feldgerät. Deshalb benötigt der Anwender ausreichend Kenntnisse über das Kommunikationsprotokoll des Feldgerätes.

4.14.4.4.1 Konfiguration des Flatstreams

Um den Flatstream zu nutzen, muss der Programmablauf erweitert werden. Die Zykluszeit der Flatstream-Routinen muss auf ein Vielfaches des Buszyklus festgelegt werden. Die zusätzlichen ProgrammROUTINEN sollten in Cyclic #1 implementiert werden, um die Datenkonsistenz zu gewährleisten.

Bei der Minimalkonfiguration müssen die Register "InputMTU" und "OutputMTU" eingestellt werden. Alle anderen Register werden beim Start mit Standardwerten belegt und können sofort genutzt werden. Sie stellen zusätzliche Optionen bereit, um Daten kompakter zu übertragen bzw. den allgemeinen Ablauf hoch effizient zu gestalten.

Mit den Forward-Registern wird der Ablauf des Flatstream-Protokolls erweitert. Diese Funktion eignet sich, um die Datenrate des Flatstreams stark zu erhöhen, bedeutet aber erheblichen Mehraufwand bei der Erstellung des Programmablaufs.

4.14.4.4.1.1 Anzahl der aktivierten Tx- bzw. Rx-Bytes

Name:

OutputMTU

InputMTU

Diese Register definieren die Anzahl der aktivierten Tx- bzw. Rx-Bytes und somit auch die maximale Größe einer Sequenz. Der Anwender muss beachten, dass mehr freigegebene Bytes auch eine stärkere Belastung für das Bussystem bedeuten.

Information:

In der weiteren Beschreibung stehen die Bezeichnungen "OutputMTU" und "InputMTU" nicht für die hier erläuterten Register, sondern als Synonym für die momentan aktivierten Tx- bzw. Rx-Bytes.

Datentyp	Werte
USINT	Siehe modulspezifische Registerübersicht (theoretisch: 3 bis 27)

4.14.4.4.2 Bedienung des Flatstreams

Bei der Verwendung des Flatstreams ist die Kommunikationsrichtung von großer Bedeutung. Für das Senden von Daten an ein Modul (Output-Richtung) werden die Tx-Bytes genutzt. Für den Empfang von Daten eines Moduls (Input-Richtung) sind die Rx-Bytes vorgesehen.

Mit den Registern "OutputSequence" und "InputSequence" wird die Kommunikation gesteuert bzw. abgesichert, das heißt, der Sender gibt damit die Anweisung, Daten zu übernehmen und der Empfänger bestätigt eine erfolgreich übertragene Sequenz.

4.14.4.4.2.1 Format der Ein- und Ausgangsbytes

Name:

"Format des Flatstream" im Automation Studio

Bei einigen Modulen kann mit Hilfe dieser Funktion eingestellt werden, wie die Ein- und Ausgangsbytes des Flatstream (Tx- bzw. Rx-Bytes) übergeben werden.

- **gepackt:** Daten werden als ein Array übergeben
- **byteweise:** Daten werden als einzelne Bytes übergeben

4.14.4.4.2.2 Transport der Nutzdaten und der Controlbytes

Name:

TxByte1 bis TxByteN

RxByte1 bis RxByteN

(Die Größe der Zahl N ist je nach verwendetem Bus Controller Modell unterschiedlich.)

Die Tx- bzw. Rx-Bytes sind zyklische Register, die zum Transport der Nutzdaten und der notwendigen Controlbytes dienen. Die Anzahl aktiver Tx- bzw. Rx-Bytes ergibt sich aus der Konfiguration der Register "OutputMTU" bzw. "InputMTU".

Im Programmablauf des Anwenders können nur die Tx- bzw. Rx-Bytes der CPU genutzt werden. Innerhalb des Moduls gibt es die entsprechenden Gegenstücke, welche für den Anwender nicht zugänglich sind. Aus diesem Grund wurden die Bezeichnungen aus Sicht der CPU gewählt.

- "T" - "transmit" → CPU *sendet* Daten an das Modul
- "R" - "receive" → CPU *empfängt* Daten vom Modul

Datentyp	Werte
USINT	0 bis 255

4.14.4.4.2.3 Controlbytes

Neben den Nutzdaten übertragen die Tx- bzw. Rx-Bytes auch die sogenannten Controlbytes. Sie enthalten zusätzliche Informationen über den Datenstrom, damit der Empfänger die übertragenen Segmente wieder korrekt zur ursprünglichen Nachricht zusammensetzen kann.

Bitstruktur eines Controlbytes

Bit	Bezeichnung	Wert	Information
0 - 5	SegmentLength	0 - 63	Bytegröße des folgenden Segments (Standard: max. MTU-Größe - 1)
6	nextCBPos	0	Nächstes Controlbyte zu Beginn der nächsten MTU
		1	Nächstes Controlbyte direkt nach Ende des Segments
7	MessageEndBit	0	Nachricht wird nach dem folgenden Segment fortgesetzt
		1	Nachricht wird durch das folgende Segment beendet

SegmentLength

Die Segmentlänge kündigt dem Empfänger an, wie lang das kommende Segment ist. Wenn die eingestellte Segmentlänge für eine Nachricht nicht ausreicht, muss die Mitteilung auf mehrere Segmente verteilt werden. In diesen Fällen kann das tatsächliche Ende der Nachricht über Bit 7 (Controlbyte) erkannt werden.

Information:

Bei der Bestimmung der Segmentlänge wird das Controlbyte nicht mitgerechnet. Die Segmentlänge ergibt sich rein aus den Bytes der Nutzdaten.

nextCBPos

Mit diesem Bit wird angezeigt, an welcher Position das nächste Controlbyte zu erwarten ist. Diese Information ist vor allem bei Anwendung der Option "MultiSegmentMTU" wichtig.

Bei der Flatstream-Kommunikation mit MultiSegmentMTUs ist das nächste Controlbyte nicht mehr auf dem ersten Rx-Byte der darauffolgenden MTU zu erwarten, sondern wird direkt im Anschluss an das Segment übertragen.

MessageEndBit

Das "MessageEndBit" wird gesetzt, wenn das folgende Segment eine Nachricht abschließt. Die Mitteilung ist vollständig übertragen und kann weiterverarbeitet werden.

Information:

In Output-Richtung muss dieses Bit auch dann gesetzt werden, wenn ein einzelnes Segment ausreicht, um die vollständige Nachricht aufzunehmen. Das Modul verarbeitet eine Mitteilung intern nur, wenn diese Kennzeichnung vorgenommen wurde.

Die Größe einer übertragenen Mitteilung lässt sich berechnen, wenn alle Segmentlängen der Nachricht addiert werden.

Flatstream-Formel zur Berechnung der Nachrichtenlänge:

Nachricht [Byte] = Segmentlängen (aller CBs ohne ME) + Segmentlänge (des ersten CB mit ME)	CB	Controlbyte
	ME	MessageEndBit

4.14.4.2.4 Kommunikationsstatus der CPU

Name:

OutputSequence

Das Register "OutputSequence" enthält Informationen über den Kommunikationsstatus der CPU. Es wird von der CPU geschrieben und vom Modul gelesen.

Datentyp	Werte
USINT	Siehe Bitstruktur

Bitstruktur:

Bit	Bezeichnung	Wert	Information
0 - 2	OutputSequenceCounter	0 - 7	Zähler der in Output abgesetzten Sequenzen
3	OutputSyncBit	0	Output-Richtung deaktiviert (disable)
		1	Output-Richtung aktiviert (enable)
4 - 6	InputSequenceAck	0 - 7	Spiegel des InputSequenceCounters
7	InputSyncAck	0	Input-Richtung nicht bereit (disable)
		1	Input-Richtung bereit (enable)

OutputSequenceCounter

Der OutputSequenceCounter ist ein umlaufender Zähler der Sequenzen, die von der CPU abgeschickt wurden. Über den OutputSequenceCounter weist die CPU das Modul an, eine Sequenz zu übernehmen (zu diesem Zeitpunkt muss die Output-Richtung synchronisiert sein).

OutputSyncBit

Mit dem OutputSyncBit versucht die CPU den Output-Kanal zu synchronisieren.

InputSequenceAck

Der InputSequenceAck dient zur Bestätigung. Der Wert des InputSequenceCounters wird darin gespiegelt, wenn die CPU eine Sequenz erfolgreich empfangen hat.

InputSyncAck

Das Bit InputSyncAck bestätigt dem Modul die Synchronität des Input-Kanals. Die CPU zeigt damit an, dass sie bereit ist, Daten zu empfangen.

4.14.4.4.2.5 Kommunikationsstatus des Moduls

Name:

InputSequence

Das Register "InputSequence" enthält Informationen über den Kommunikationsstatus des Moduls. Es wird vom Modul geschrieben und sollte von der CPU nur gelesen werden.

Datentyp	Werte
USINT	Siehe Bitstruktur

Bitstruktur:

Bit	Bezeichnung	Wert	Information
0 - 2	InputSequenceCounter	0 - 7	Zähler der in Input abgesetzten Sequenzen
3	InputSyncBit	0	Nicht bereit (disable)
		1	Bereit (enable)
4 - 6	OutputSequenceAck	0 - 7	Spiegel des OutputSequenceCounters
7	OutputSyncAck	0	Nicht bereit (disable)
		1	Bereit (enable)

InputSequenceCounter

Der InputSequenceCounter ist ein umlaufender Zähler der Sequenzen, die vom Modul abgeschickt wurden. Über den InputSequenceCounter weist das Modul die CPU an, eine Sequenz zu übernehmen (zu diesem Zeitpunkt muss die Input-Richtung synchronisiert sein).

InputSyncBit

Mit dem InputSyncBit versucht das Modul den Input-Kanal zu synchronisieren.

OutputSequenceAck

Der OutputSequenceAck dient zur Bestätigung. Der Wert des OutputSequenceCounters wird darin gespiegelt, wenn das Modul eine Sequenz erfolgreich empfangen hat.

OutputSyncAck

Das Bit OutputSyncAck bestätigt der CPU die Synchronität des Output-Kanals. Das Modul zeigt damit an, dass es bereit ist, Daten zu empfangen.

4.14.4.4.2.6 Beziehung zwischen Output- und InputSequence

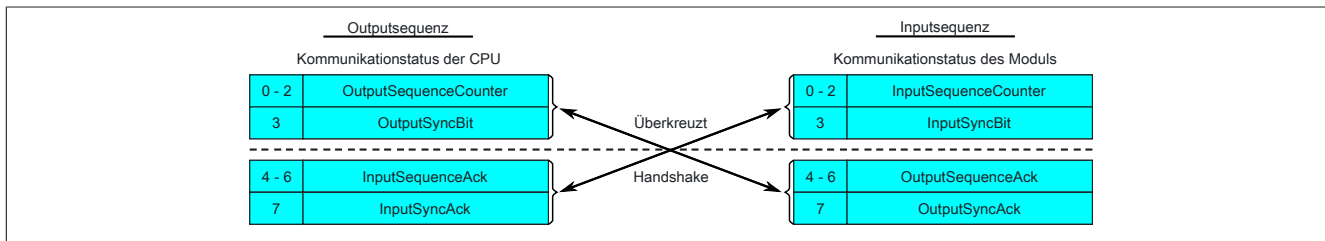


Abbildung 9: Zusammenhang zwischen Output- und InputSequence

Die Register "OutputSequence" und "InputSequence" sind logisch aus 2 Halb-Bytes aufgebaut. Über den Low-Teil wird der Gegenstelle signalisiert, ob ein Kanal geöffnet werden soll bzw. ob Daten übernommen werden können. Der High-Teil dient zur Bestätigung, wenn die geforderte Aktion erfolgreich ausgeführt wurde.

SyncBit und SyncAck

Wenn das SyncBit und das SyncAck einer Kommunikationsrichtung gesetzt sind, gilt der Kanal als "synchronisiert", das heißt, es können Nachrichten in diese Richtung versendet werden. Das Statusbit der Gegenstelle muss zyklisch überprüft werden. Falls das SyncAck zurückgesetzt wurde, muss das eigene SyncBit angepasst werden. Bevor neue Daten übertragen werden können, muss der Kanal resynchronisiert werden.

SequenceCounter und SequenceAck

Die Kommunikationspartner prüfen zyklisch, ob sich das Low-Nibble der Gegenstelle ändert. Wenn ein Kommunikationspartner eine neue Sequenz vollständig auf die MTU geschrieben hat, erhöht er seinen SequenceCounter. Daraufhin übernimmt der Empfänger die aktuelle Sequenz und bestätigt den erfolgreichen Empfang per SequenceAck. Auf diese Weise wird ein Handshake-Verfahren initiiert.

Information:

Bei einer Unterbrechung der Kommunikation werden Segmente von unvollständig übermittelten Mitteilungen verworfen. Alle fertig übertragenen Nachrichten werden bearbeitet.

4.14.4.4.3 Synchronisieren

Beim Synchronisieren wird ein Kommunikationskanal geöffnet. Es muss sichergestellt sein, dass ein Modul vorhanden und der aktuelle Wert des SequenceCounters beim Empfänger der Nachricht hinterlegt ist.

Der Flatstream bietet die Möglichkeit Vollduplex zu kommunizieren. Beide Kanäle/Kommunikationsrichtungen können separat betrachtet werden. Sie müssen unabhängig voneinander synchronisiert werden, sodass theoretisch auch simplex kommuniziert werden könnte.

Synchronisation der Output-Richtung (CPU als Sender)

Die korrespondierenden Synchronisationsbits (OutputSyncBit und OutputSyncAck) sind zurückgesetzt. Aus diesem Grund können momentan keine Nachrichten von der CPU an das Modul per Flatstream übertragen werden.

Algorithmus

1) CPU muss 000 in OutputSequenceCounter schreiben und OutputSyncBit zurücksetzen. CPU muss High-Nibble des Registers "InputSequence" zyklisch abfragen (Prüfung ob 000 in OutputSequenceAck und 0 in OutputSyncAck).
<i>Modul übernimmt den aktuellen Inhalt der InputMTU nicht, weil der Kanal noch nicht synchronisiert ist.</i> <i>Modul gleicht OutputSequenceAck und OutputSyncAck an die Werte des OutputSequenceCounters bzw. des OutputSyncBits an.</i>
2) Wenn die CPU die erwarteten Werte in OutputSequenceAck und OutputSyncAck registriert, darf sie den OutputSequenceCounter inkrementieren. Die CPU fragt das High-Nibble des Registers "OutputSequence" weiter zyklisch ab (Prüfung ob 001 in OutputSequenceAck und 0 in InputSyncAck).
<i>Modul übernimmt den aktuellen Inhalt der InputMTU nicht, weil der Kanal noch nicht synchronisiert ist.</i> <i>Modul gleicht OutputSequenceAck und OutputSyncAck an die Werte des OutputSequenceCounters bzw. des OutputSyncBits an.</i>
3) Wenn die CPU die erwarteten Werte in OutputSequenceAck und OutputSyncAck registriert, darf sie das OutputSyncBit setzen. Die CPU fragt das High-Nibble des Registers "OutputSequence" weiter zyklisch ab (Prüfung ob 001 in OutputSequenceAck und 1 in InputSyncAck).
Hinweis: Theoretisch könnten ab diesem Moment Daten übertragen werden. Es wird allerdings empfohlen, erst dann Daten zu übertragen, wenn die Output-Richtung vollständig synchronisiert ist.
<i>Modul setzt OutputSyncAck.</i>
Output-Richtung synchronisiert, CPU kann Daten an Modul senden.

Synchronisation der Input-Richtung (CPU als Empfänger)

Die korrespondierenden Synchronisationsbits (InputSyncBit und InputSyncAck) sind zurückgesetzt. Aus diesem Grund können momentan keine Nachrichten vom Modul an die CPU per Flatstream übertragen werden.

Algorithmus

<i>Modul schreibt 000 in InputSequenceCounter und setzt InputSyncBit zurück.</i> <i>Modul überwacht High-Nibble des Registers "OutputSequence" - erwartet 000 in InputSequenceAck bzw. 0 in InputSyncAck.</i>
1) CPU darf den aktuellen Inhalt der InputMTU nicht übernehmen, weil der Kanal noch nicht synchronisiert ist. CPU muss InputSequenceAck und InputSyncAck an die Werte des InputSequenceCounters bzw. des InputSyncBits angleichen.
<i>Wenn das Modul die erwarteten Werte in InputSequenceAck und in InputSyncAck registriert, inkrementiert es den InputSequenceCounter.</i> <i>Modul überwacht High-Nibble des Registers "OutputSequence" - erwartet 001 in InputSequenceAck bzw. 0 in InputSyncAck.</i>
2) CPU darf den aktuellen Inhalt der InputMTU nicht übernehmen, weil der Kanal noch nicht synchronisiert ist. CPU muss InputSequenceAck und InputSyncAck an die Werte des InputSequenceCounters bzw. des InputSyncBits angleichen.
<i>Wenn das Modul die erwarteten Werte in InputSequenceAck und in InputSyncAck registriert, setzt es das InputSyncBit.</i> <i>Modul überwacht High-Nibble des Registers "OutputSequence" - erwartet 1 in InputSyncAck.</i>
3) CPU darf InputSyncAck setzen.
Hinweis: Theoretisch könnten bereits in diesem Zyklus Daten übertragen werden. Es gilt: Wenn das InputSyncBit gesetzt ist und der InputSequenceCounter um 1 erhöht wurde, müssen die Informationen der aktivierten Rx-Bytes übernommen und bestätigt werden (siehe dazu auch Kommunikation in Input-Richtung).
Input-Richtung synchronisiert, Modul kann Daten an CPU senden.

4.14.4.4 Senden und Empfangen

Wenn ein Kanal synchronisiert ist, gilt die Gegenstelle als empfangsbereit und der Sender kann Nachrichten verschicken. Bevor der Sender Daten absetzen kann, legt er das sogenannte Sendearray an, um den Anforderungen des Flatstreams gerecht zu werden.

Die sendende Station muss für jedes erstellte Segment ein individuelles Controlbyte generieren. Ein solches Controlbyte enthält Informationen, wie der nächste Teil der übertragenen Daten zu verarbeiten ist. Die Position des nächsten Controlbytes im Datenstrom kann variieren. Aus diesem Grund muss zu jedem Zeitpunkt eindeutig definiert sein, wann ein neues Controlbyte übermittelt wird. Das erste Controlbyte befindet sich immer auf dem ersten Byte der ersten Sequenz. Alle weiteren Positionen werden rekursiv mitgeteilt.

Flatstream-Formel zur Berechnung der Position des nächsten Controlbytes:

$$\text{Position (nächstes Controlbyte)} = \text{aktuelle Position} + 1 + \text{Segmentlänge}$$

Beispiel

Es werden 3 unabhängige Nachrichten (7 Bytes, 2 Bytes, 9 Bytes) über eine 7-Byte breite MTU übermittelt. Die sonstige Konfiguration entspricht den Standardeinstellungen.

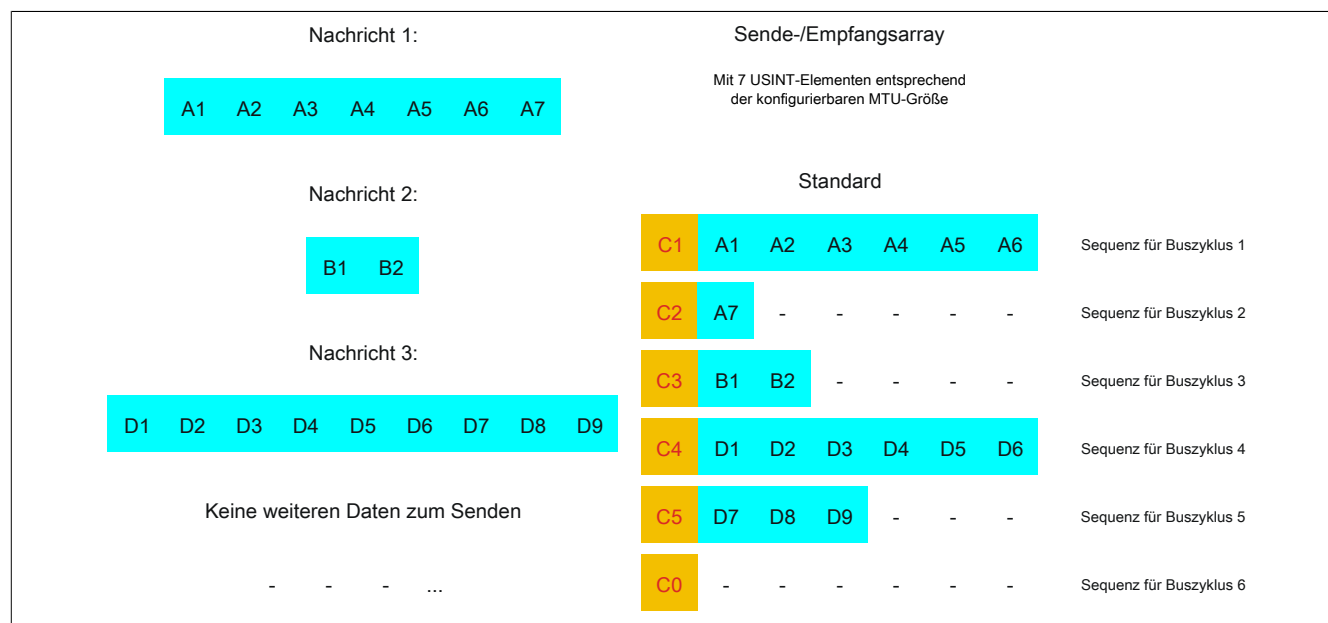


Abbildung 10: Sende-/Empfangsarray (Standard)

Zunächst müssen die Nachrichten in Segmente geteilt werden. Bei der Standardkonfiguration muss sichergestellt sein, dass jede Sequenz ein gesamtes Segment inklusive dem dazugehörigen Controlbyte aufnehmen kann. Die Sequenz ist auf die Größe der aktivierten MTU begrenzt, das heißt, ein Segment muss mindestens um 1 Byte kleiner sein als die aktivierte MTU.

MTU = 7 Bytes → max. Segmentlänge 6 Bytes

- Nachricht 1 (7 Bytes)
 - ⇒ erstes Segment = Controlbyte + 6 Datenbytes
 - ⇒ zweites Segment = Controlbyte + 1 Datenbyte
- Nachricht 2 (2 Bytes)
 - ⇒ erstes Segment = Controlbyte + 2 Datenbytes
- Nachricht 3 (9 Bytes)
 - ⇒ erstes Segment = Controlbyte + 6 Datenbytes
 - ⇒ zweites Segment = Controlbyte + 3 Datenbytes
- Keine weiteren Nachrichten
 - ⇒ C0-Controlbyte

Für jedes gebildete Segment muss ein spezifisches Controlbyte generiert werden. Außerdem wird das Controlbyte C0 generiert, um die Kommunikation auf Standby halten zu können.

C0 (Controlbyte0)			C1 (Controlbyte1)			C2 (Controlbyte2)		
- SegmentLength (0)	=	0	- SegmentLength (6)	=	6	- SegmentLength (1)	=	1
- nextCBPos (0)	=	0	- nextCBPos (0)	=	0	- nextCBPos (0)	=	0
- MessageEndBit (0)	=	0	- MessageEndBit (0)	=	0	- MessageEndBit (1)	=	128
Controlbyte	Σ	0	Controlbyte	Σ	6	Controlbyte	Σ	129

Tabelle 3: Flatstream-Ermittlung der Controlbytes für Beispiel mit Standardkonfiguration (Teil 1)

C3 (Controlbyte3)			C4 (Controlbyte4)			C5 (Controlbyte5)		
- SegmentLength (2)	=	2	- SegmentLength (6)	=	6	- SegmentLength (3)	=	3
- nextCBPos (0)	=	0	- nextCBPos (0)	=	0	- nextCBPos (0)	=	0
- MessageEndBit (1)	=	128	- MessageEndBit (0)	=	0	- MessageEndBit (1)	=	128
Controlbyte	Σ	130	Controlbyte	Σ	6	Controlbyte	Σ	131

Tabelle 4: Flatstream-Ermittlung der Controlbytes für Beispiel mit Standardkonfiguration (Teil 2)

4.14.4.4.5 Senden von Daten an ein Modul (Output)

Beim Senden muss das Sendearray im Programmablauf generiert werden. Danach wird es Sequenz für Sequenz über den Flatstream übertragen und vom Modul empfangen.

Information:

Obwohl alle B&R Module mit Flatstream-Kommunikation stets die kompakteste Übertragung in Output-Richtung unterstützen wird empfohlen die Übertragungsarrays für beide Kommunikationsrichtungen gleichermaßen zu gestalten.

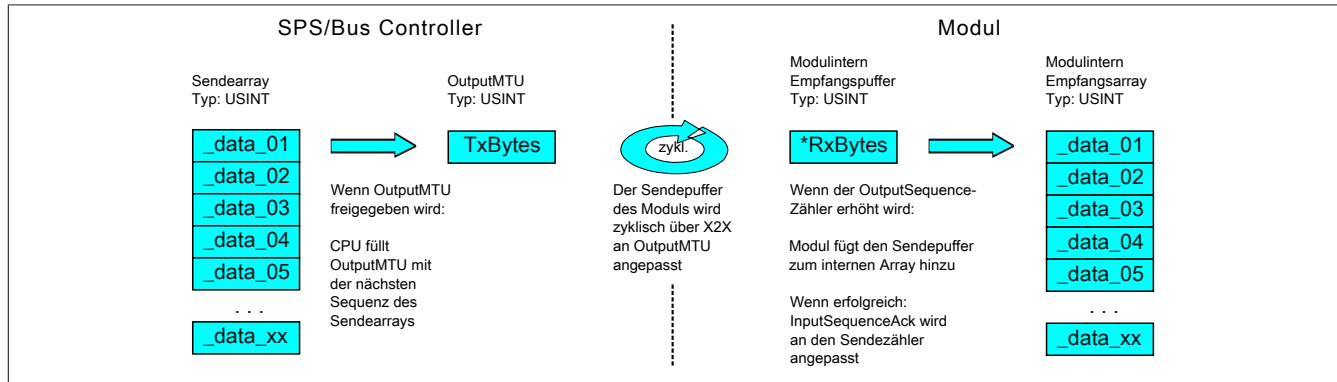


Abbildung 11: Kommunikation per Flatstream (Output)

Nachricht kleiner als OutputMTU

Die Länge der Nachricht sei zunächst kleiner als die OutputMTU. In diesem Fall würde eine Sequenz ausreichen, um die gesamte Nachricht und ein benötigtes Controlbyte zu übertragen.

Algorithmus

Zyklische Statusabfrage: - Modul überwacht OutputSequenceCounter
0) Zyklische Prüfungen: - CPU muss OutputSyncAck prüfen → falls OutputSyncAck = 0; OutputSyncBit zurücksetzen und Kanal resynchronisieren - CPU muss Freigabe der OutputMTU prüfen → falls OutputSequenceCounter > InputSequenceAck; MTU nicht freigegeben, weil letzte Sequenz noch nicht bestätigt
1) Vorbereitung (Sendearray anlegen): - CPU muss Nachricht auf zulässige Segmente aufteilen und entsprechende Controlbytes bilden - CPU muss Segmente und Controlbytes zu Sendearray zusammenfügen
2) Senden: - CPU überträgt das aktuelle Element des Sendearrays in die OutputMTU → OutputMTU wird zyklisch in den Sendepuffer des Moduls übertragen, aber noch nicht weiterverarbeitet - CPU muss OutputSequenceCounter erhöhen
Reaktion: - Modul übernimmt die Bytes des internen Empfangspuffers und fügt sie an das interne Empfangsarray an - Modul sendet Bestätigung; schreibt Wert des OutputSequenceCounters auf OutputSequenceAck
3) Abschluss: - CPU muss OutputSequenceAck überwachen → Eine Sequenz gilt erst dann als erfolgreich übertragen, wenn sie über das OutputSequenceAck bestätigt wurde. Um Übertragungsfehler auch bei der letzten Sequenz zu erkennen, muss sichergestellt werden, dass der Abschluss lange genug durchlaufen wird.
Hinweis: Für eine exakte Überwachung der Kommunikationszeiten sollten die Taskzyklen gezählt werden, die seit der letzten Erhöhung des OutputSequenceCounters vergangen sind. Auf diese Weise kann die Anzahl der Buszyklen abgeschätzt werden, die bislang zur Übertragung benötigt wurden. Übersteigt der Überwachungszähler eine vorgegebene Schwelle, kann die Sequenz als verloren betrachtet werden. (Das Verhältnis von Bus- und Taskzyklus kann vom Anwender beeinflusst werden, sodass der Schwellwert individuell zu ermitteln ist.) - Weitere Sequenzen dürfen erst nach erfolgreicher Abschlussprüfung im nächsten Buszyklus versendet werden.

Nachricht größer als OutputMTU

Das Sendearray, welches im Programmablauf erstellt werden muss, besteht aus mehreren Elementen. Der Anwender muss die Control- und Datenbytes korrekt anordnen und die Arrayelemente nacheinander übertragen. Der Übertragungsalgorithmus bleibt gleich und wird ab dem Punkt *zyklische Prüfungen* wiederholt durchlaufen.

Allgemeines Ablaufdiagramm

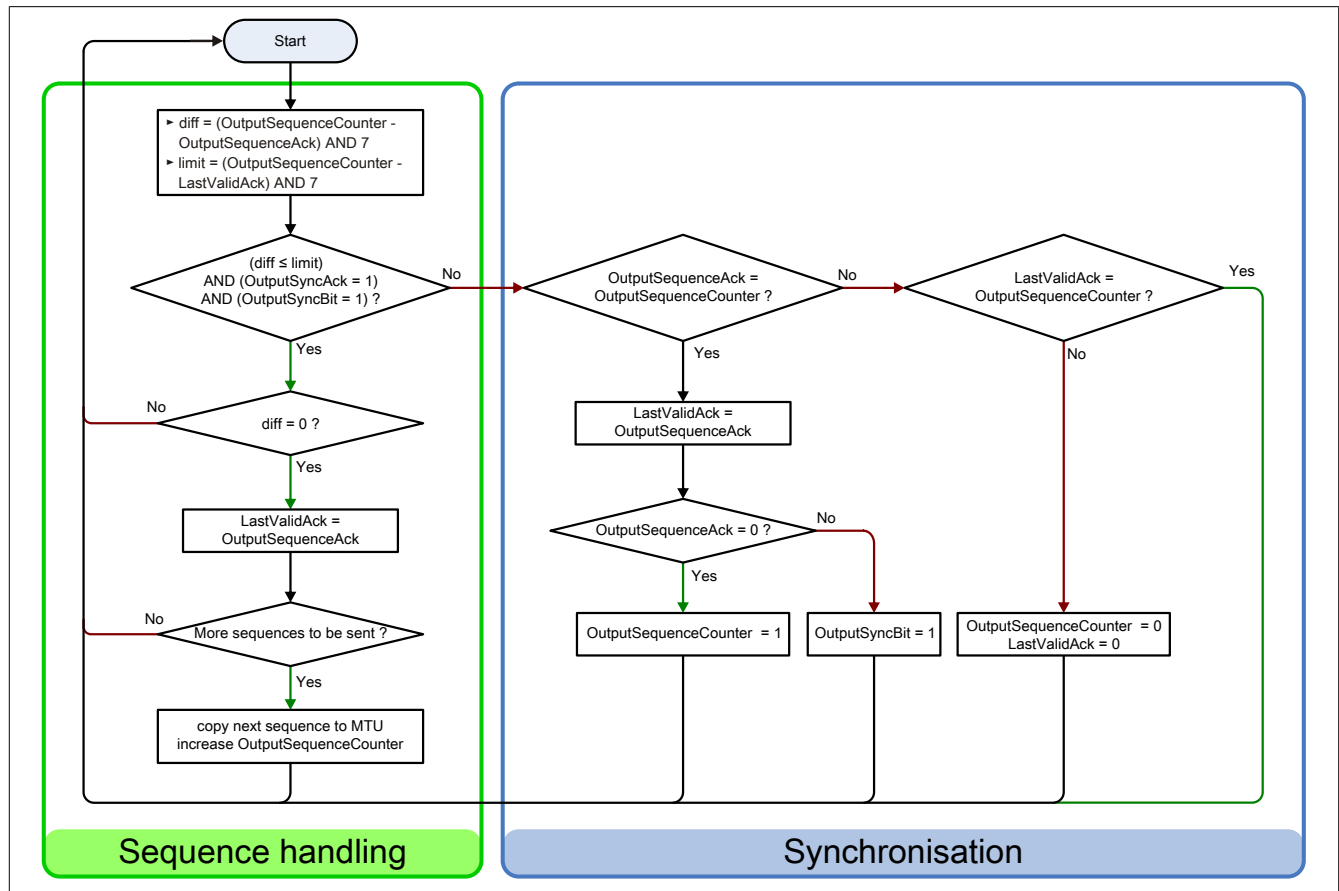


Abbildung 12: Ablaufdiagramm für Output-Richtung

4.14.4.4.6 Empfangen von Daten aus einem Modul (Input)

Beim Empfangen von Daten wird das Sendearray vom Modul generiert, über den Flatstream übertragen und muss auf dem Empfangsarray abgebildet werden. Die Struktur des ankommenden Datenstroms kann über das Modusregister eingestellt werden. Der Algorithmus zum Empfangen bleibt dabei aber unverändert.

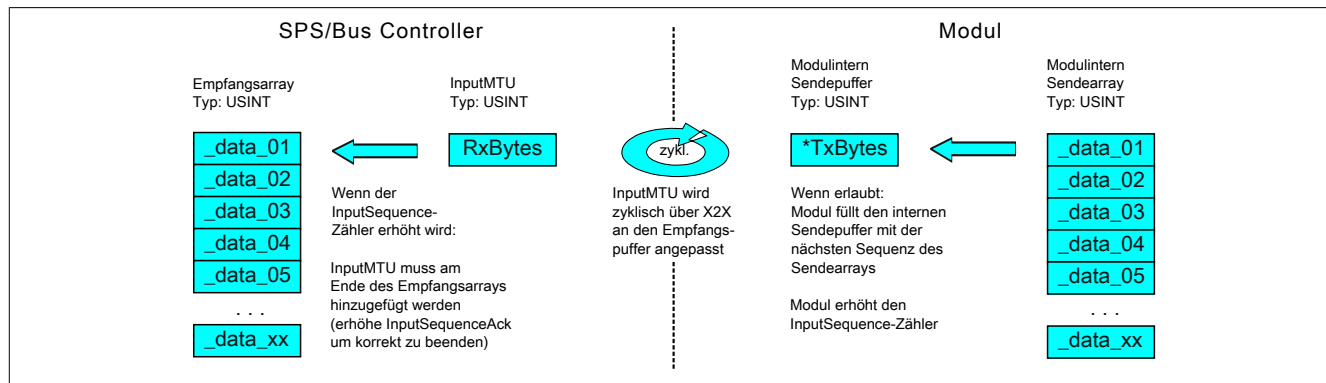


Abbildung 13: Kommunikation per Flatstream (Input)

Algorithmus

0) Zyklische Statusabfrage: - CPU muss InputSequenceCounter überwachen
Zyklische Prüfungen: - Modul prüft InputSyncAck - Modul prüft InputSequenceAck
Vorbereitung: - Modul bildet Segmente bzw. Controlbytes und legt Sendearray an
Aktion: - Modul überträgt das aktuelle Element des internen Sendearrays in den internen Sendepuffer - Modul erhöht InputSequenceCounter
1) Empfangen (sobald InputSequenceCounter erhöht): - CPU muss Daten aus InputMTU übernehmen und an das Ende des Empfangsarrays anfügen - CPU muss InputSequenceAck an InputSequenceCounter der aktuell verarbeiteten Sequenz angleichen
Abschluss: - Modul überwacht InputSequenceAck → Eine Sequenz gilt erst dann als erfolgreich übertragen, wenn sie über das InputSequenceAck bestätigt wurde. - Weitere Sequenzen werden erst nach erfolgreicher Abschlussprüfung im nächsten Buszyklus versendet.

Allgemeines Ablaufdiagramm

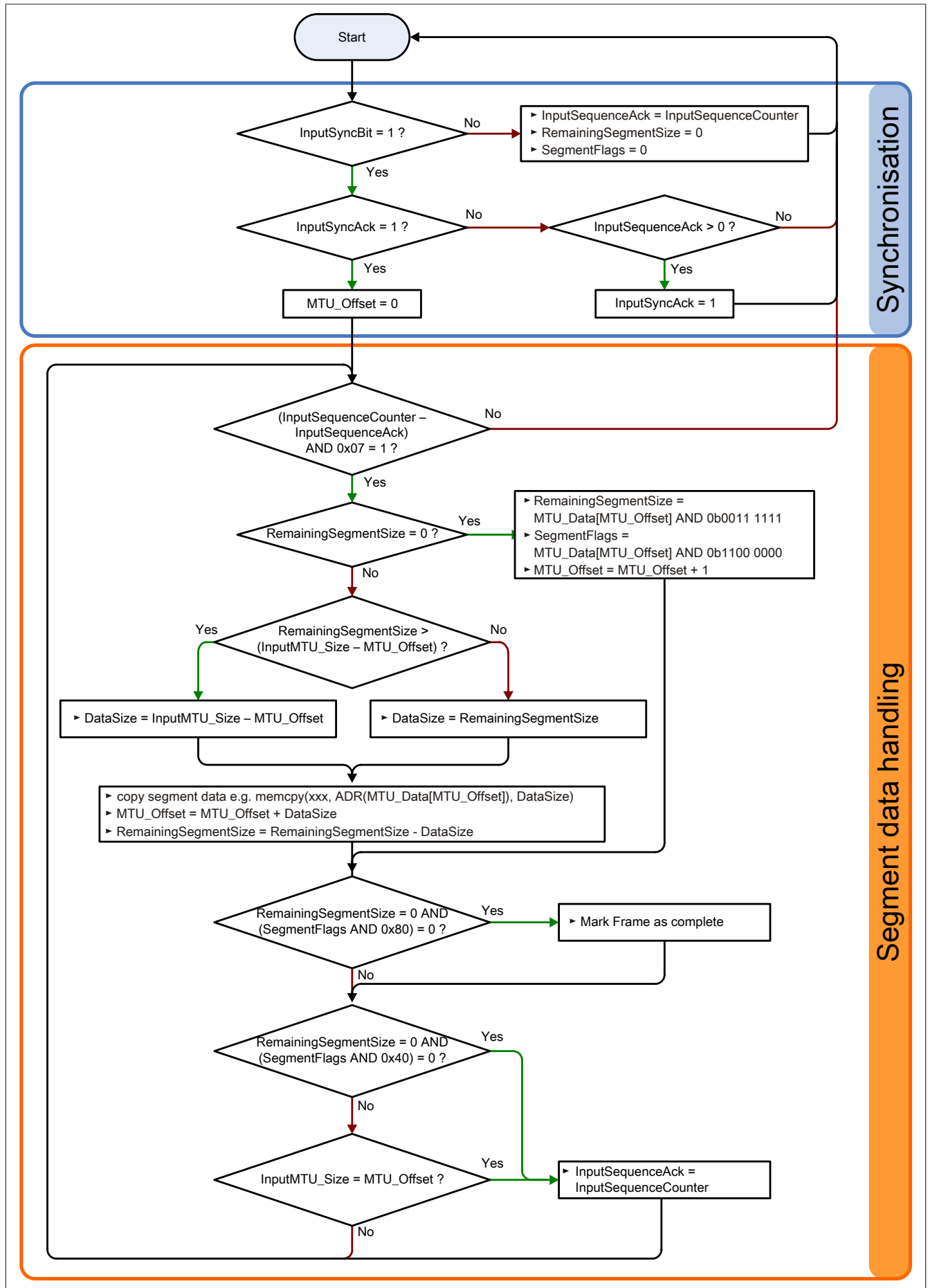


Abbildung 14: Ablaufdiagramm für Input-Richtung

4.14.4.4.7 Details

Es wird empfohlen die übertragenen Nachrichten in separate Empfangsarrays abzulegen

Nach der Übermittlung eines gesetzten MessageEndBits sollte das Folgesegment zum Empfangsarray hinzugefügt werden. Danach ist die Mitteilung vollständig und kann intern weiterverarbeitet werden. Für die nächste Nachricht sollte ein neues/separates Array angelegt werden.

Information:

Bei der Übertragung mit MultiSegmentMTUs können sich mehrere kurze Nachrichten in einer Sequenz befinden. Im Programmablauf muss sichergestellt sein, dass genügend Empfangsarrays verwaltet werden können. Das Acknowledge-Register darf erst nach Übernahme der gesamten Sequenz angepasst werden.

Wenn ein SequenceCounter um mehr als einen Zähler inkrementiert wird, liegt ein Fehler vor

Anmerkung: Beim Betrieb ohne Forward ist diese Situation sehr unwahrscheinlich.

In diesem Fall stoppt der Empfänger. Alle weiteren eintreffenden Sequenzen werden ignoriert, bis die Sendung mit dem korrekten SequenceCounter wiederholt wird. Durch diese Reaktion erhält der Sender keine Bestätigungen mehr für die abgesetzten Sequenzen. Über den SequenceAck der Gegenstelle kann der Sender die letzte erfolgreich übertragene Sequenz identifizieren und die Übertragung ab dieser Stelle fortsetzen.

Bestätigungen müssen auf Gültigkeit geprüft werden

Wenn der Empfänger eine Sequenz erfolgreich übernommen hat, muss sie bestätigt werden. Dazu übernimmt der Empfänger den mitgesendeten Wert des SequenceCounters und gleicht den SequenceAck daran an. Der Absender liest das SequenceAck und registriert die erfolgreiche Übermittlung. Falls dem Absender eine Sequenz bestätigt wird, die noch nicht abgesendet wurde, muss die Übertragung unterbrochen und der Kanal resynchronisiert werden. Die Synchronisationsbits werden zurückgesetzt und die aktuelle/unvollständige Nachricht wird verworfen. Sie muss nach der Resynchronisierung des Kanals erneut versendet werden.

4.14.4.4.8 Flatstream Modus

Name:

FlatstreamMode

In Input-Richtung wird das Sende-Array automatisch generiert. Dem Anwender werden über dieses Register 2 Optionen zur Verfügung gestellt, um eine kompaktere Anordnung beim eintreffenden Datenstrom zu erlauben. Nach der Aktivierung muss der Programmablauf zur Auswertung entsprechend angepasst werden.

Information:

Alle B&R Module, die den Flatstream-Modus anbieten, unterstützen in Output-Richtung die Optionen "große Segmente" und "MultiSegmentMTU". Nur für die Input-Richtung muss die kompakte Übertragung explizit erlaubt werden.

Bitstruktur:

Bit	Bezeichnung	Wert	Information
0	MultiSegmentMTU	0	Nicht erlaubt (Standard)
		1	Erlaubt
1	Große Segmente	0	Nicht erlaubt (Standard)
		1	Erlaubt
2 - 7	Reserviert		

Standard

Per Standard sind beide Optionen zur kompakten Übertragung in Input-Richtung deaktiviert.

1. Vom Modul werden nur Segmente gebildet, die mindestens ein Byte kleiner sind als die aktivierte MTU. Jede Sequenz beginnt mit einem Controlbyte, sodass der Datenstrom klar strukturiert ist und relativ einfach ausgewertet werden kann.
2. Weil die Länge einer Flatstream-Nachricht beliebig lang sein darf, füllt das letzte Segment der Mitteilung häufig nicht den gesamten Platz der MTU aus. Per Standard werden während eines solchen Übertragungszyklus die restlichen Bytes nicht verwendet.

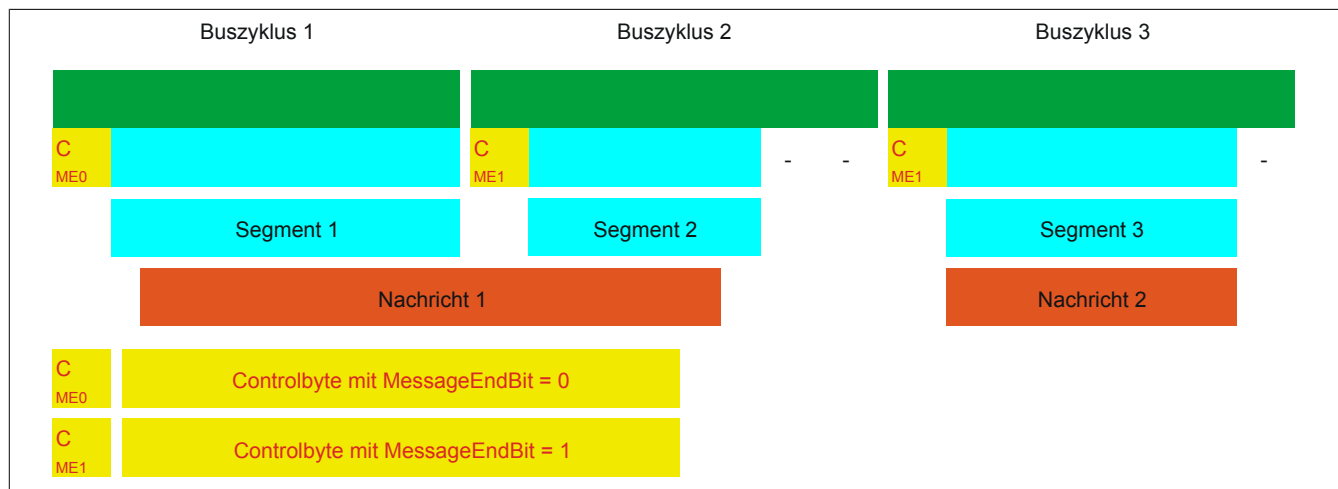


Abbildung 15: Anordnung von Nachrichten in der MTU (Standard)

MultiSegmentMTU erlaubt

Bei dieser Option wird die InputMTU vollständig befüllt (wenn genügend Daten anstehen). Die zuvor frei gebliebenen Rx-Bytes übertragen die nächsten Controlbytes bzw. deren Segmente. Auf diese Weise können die aktivierten Rx-Bytes effizienter genutzt werden.

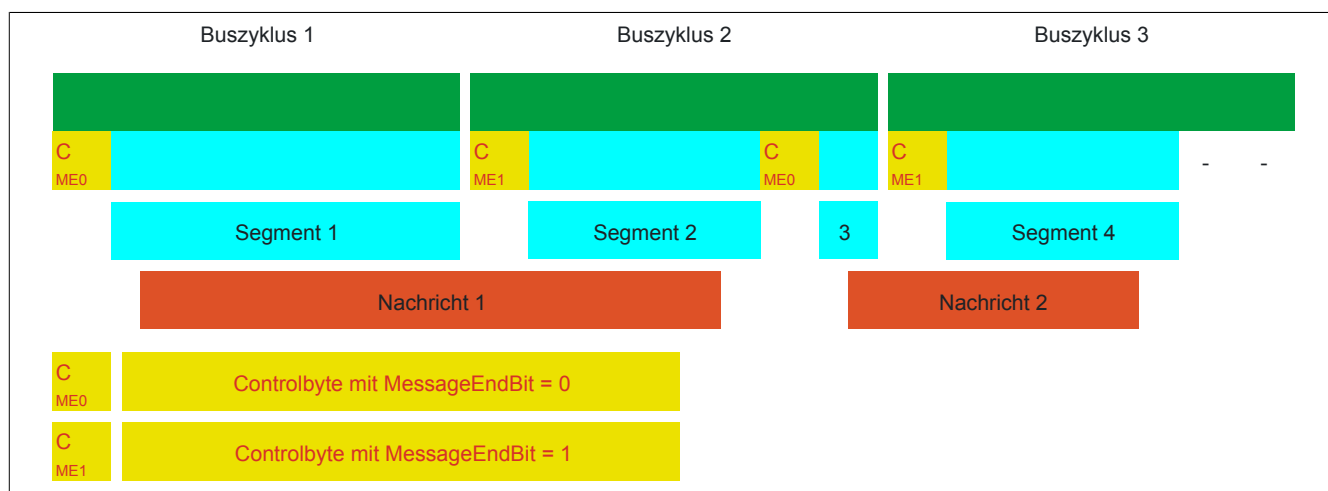


Abbildung 16: Anordnung von Nachrichten in der MTU (MultiSegmentMTU)

Große Segmente erlaubt

Bei der Übertragung sehr langer Mitteilungen bzw. bei der Aktivierung von nur wenigen Rx-Bytes müssen per Standard sehr viele Segmente gebildet werden. Das Bussystem wird stärker belastet als nötig, weil für jedes Segment ein zusätzliches Controlbyte erstellt und übertragen wird. Mit der Option "große Segmente" wird die Segmentlänge unabhängig von der InputMTU auf 63 Bytes begrenzt. Ein Segment darf sich über mehrere Sequenzen erstrecken, das heißt, es können auch reine Sequenzen ohne Controlbyte auftreten.

Information:

Die Möglichkeit eine Nachricht auf mehrere Segmente aufzuteilen bleibt erhalten, das heißt, wird diese Option genutzt und treten Nachrichten mit mehr als 63 Bytes auf, kann die Mitteilung weiterhin auf mehrere Segmente verteilt werden.

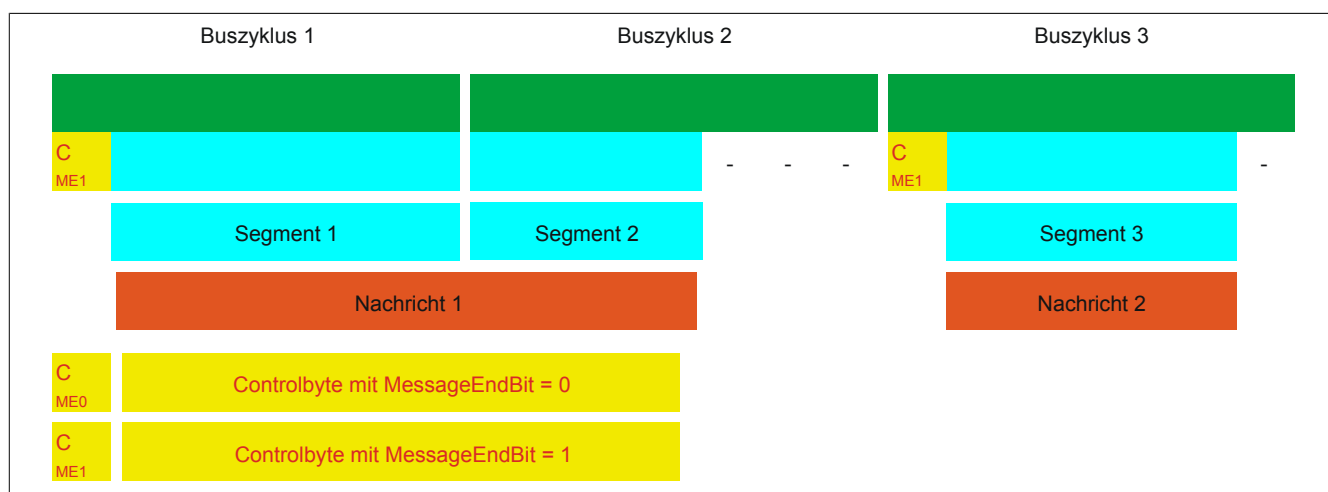


Abbildung 17: Anordnung von Nachrichten in der MTU (große Segmente)

Anwendung beider Optionen

Die beiden Optionen dürfen auch gleichzeitig angewendet werden.

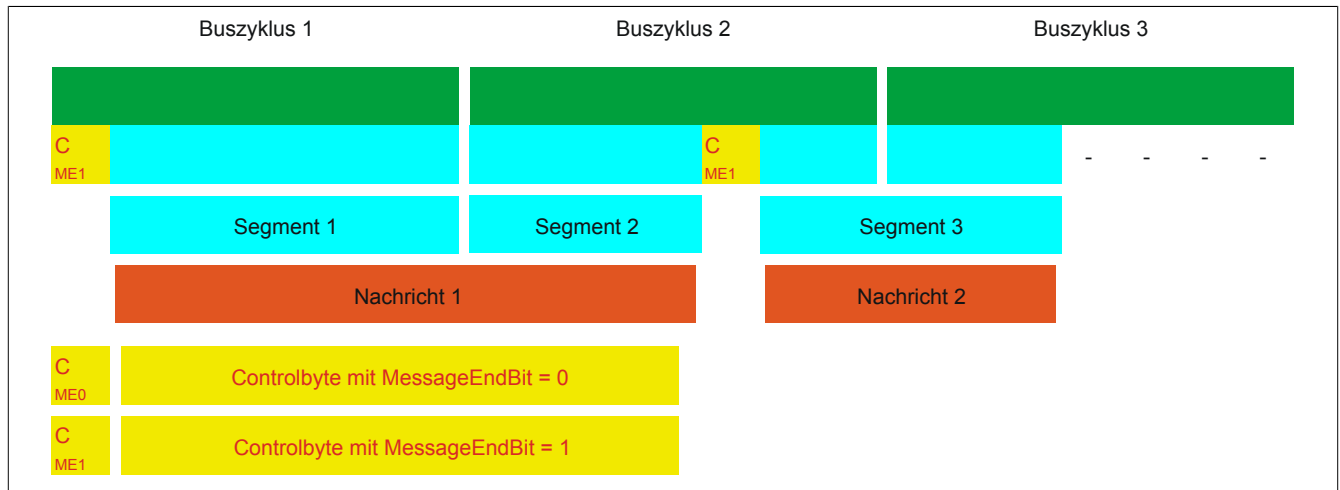


Abbildung 18: Anordnung von Nachrichten in der MTU (große Segmente und MultiSegmentMTU)

4.14.4.4.9 Anpassung des Flatstreams

Wenn die Strukturierung der Nachrichten verändert wurde, verändert sich auch die Anordnung der Daten im Sende-/Empfangsarray. Für das eingangs genannte Beispiel ergeben sich die folgenden Änderungen.

MultiSegmentMTU

Wenn MultiSegmentMTUs erlaubt sind, können "freie Stellen" in einer MTU genutzt werden. Diese "freien Stellen" entstehen, wenn das letzte Segment einer Nachricht nicht die gesamte MTU ausnutzt. MultiSegmentMTUs ermöglichen die Verwendung dieser Bits, um die folgenden Controlbytes bzw. Segmente zu übertragen. Im Programmablauf wird das "nextCBPos"-Bit innerhalb des Controlbytes gesetzt, damit der Empfänger das nächste Controlbyte korrekt identifizieren kann.

Beispiel

Es werden 3 unabhängige Nachrichten (7 Bytes, 2 Bytes, 9 Bytes) über eine 7-Byte breite MTU übermittelt. Die Konfiguration erlaubt die Übertragung von MultiSegmentMTUs.

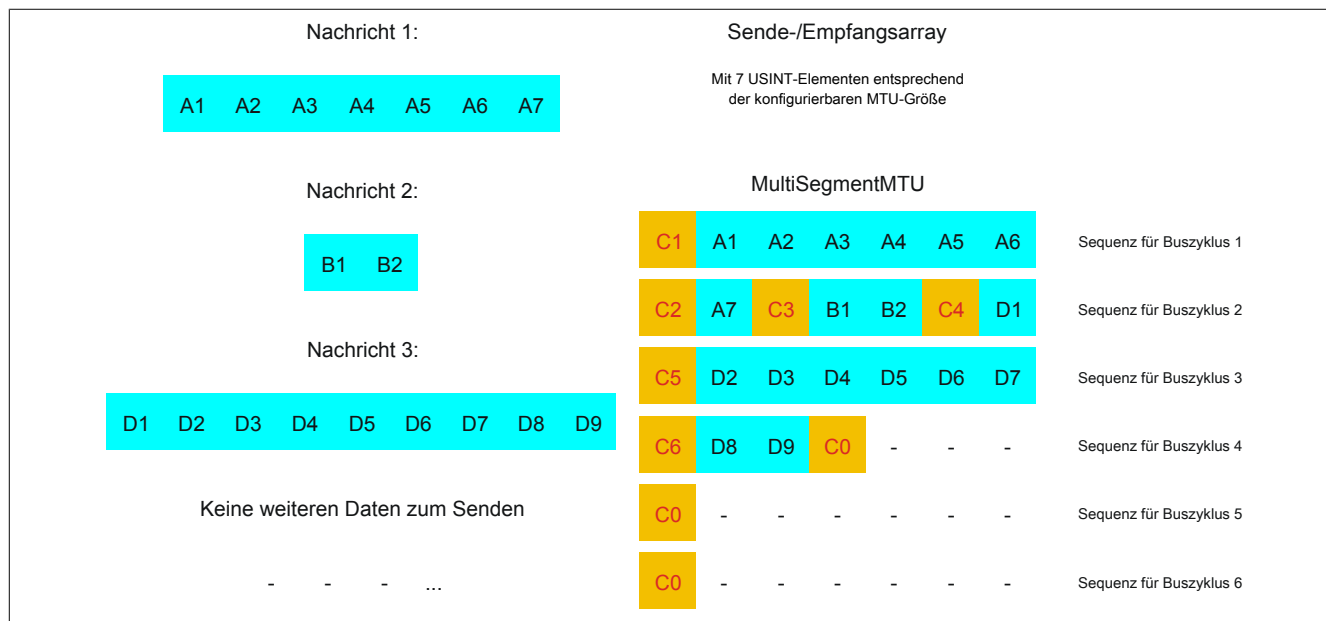


Abbildung 19: Sende-/Empfangsarray (MultiSegmentMTU)

Zunächst müssen die Nachrichten in Segmente geteilt werden. Wie in der Standardkonfiguration muss sichergestellt sein, dass jede Sequenz mit einem Controlbyte beginnt. Die freien Bits in der MTU am Ende einer Nachricht, werden allerdings mit Daten der Folgenachricht aufgefüllt. Bei dieser Option wird das Bit "nextCBPos" immer gesetzt, wenn im Anschluss an das Controlbyte Nutzdaten übertragen werden.

MTU = 7 Bytes → max. Segmentlänge 6 Bytes

- Nachricht 1 (7 Bytes)
 - ⇒ erstes Segment = Controlbyte + 6 Datenbytes (MTU voll)
 - ⇒ zweites Segment = Controlbyte + 1 Datenbyte (MTU noch 5 leere Bytes)
- Nachricht 2 (2 Bytes)
 - ⇒ erstes Segment = Controlbyte + 2 Datenbytes (MTU noch 2 leere Bytes)
- Nachricht 3 (9 Bytes)
 - ⇒ erstes Segment = Controlbyte + 1 Datenbyte (MTU voll)
 - ⇒ zweites Segment = Controlbyte + 6 Datenbytes (MTU voll)
 - ⇒ drittes Segment = Controlbyte + 2 Datenbytes (MTU noch 4 leere Bytes)
- Keine weiteren Nachrichten
 - ⇒ C0-Controlbyte

Für jedes gebildete Segment muss ein spezifisches Controlbyte generiert werden. Außerdem wird das Controlbyte C0 generiert, um die Kommunikation auf Standby halten zu können.

C1 (Controlbyte1)			C2 (Controlbyte2)			C3 (Controlbyte3)		
- SegmentLength (6)	=	6	- SegmentLength (1)	=	1	- SegmentLength (2)	=	2
- nextCBPos (1)	=	64	- nextCBPos (1)	=	64	- nextCBPos (1)	=	64
- MessageEndBit (0)	=	0	- MessageEndBit (1)	=	128	- MessageEndBit (1)	=	128
Controlbyte	Σ	70	Controlbyte	Σ	193	Controlbyte	Σ	194

Tabelle 5: Flatstream-Ermittlung der Controlbytes für Beispiel mit MultiSegmentMTU (Teil 1)

Warnung!

Die zweite Sequenz darf erst über den SequenceAck bestätigt werden, wenn sie vollständig verarbeitet wurde. Im Beispiel befinden sich 3 verschiedene Segmente innerhalb der zweiten Sequenz, das heißt, im Programmablauf müssen ausreichend Empfänger-Arrays gehandhabt werden können.

C4 (Controlbyte4)			C5 (Controlbyte5)			C6 (Controlbyte6)		
- SegmentLength (1)	=	1	- SegmentLength (6)	=	6	- SegmentLength (2)	=	2
- nextCBPos (6)	=	6	- nextCBPos (1)	=	64	- nextCBPos (1)	=	64
- MessageEndBit (0)	=	0	- MessageEndBit (1)	=	0	- MessageEndBit (1)	=	128
Controlbyte	Σ	7	Controlbyte	Σ	70	Controlbyte	Σ	194

Tabelle 6: Flatstream-Ermittlung der Controlbytes für Beispiel mit MultiSegmentMTU (Teil 2)

Große Segmente

Die Segmente werden auf maximal 63 Bytes begrenzt. Damit können sie größer sein als die aktive MTU. Diese großen Segmente werden bei der Übertragung auf mehrere Sequenzen aufgeteilt. Es können Sequenzen ohne Controlbyte auftreten, die vollständig mit Nutzdaten befüllt sind.

Information:

Um die Größe eines Datenpakets nicht ebenfalls auf 63 Bytes zu begrenzen, bleibt die Möglichkeit erhalten, eine Nachricht in mehrere Segmente zu untergliedern.

Beispiel

Es werden 3 unabhängige Nachrichten (7 Bytes, 2 Bytes, 9 Bytes) über eine 7-Byte breite MTU übermittelt. Die Konfiguration erlaubt die Übertragung von großen Segmenten.

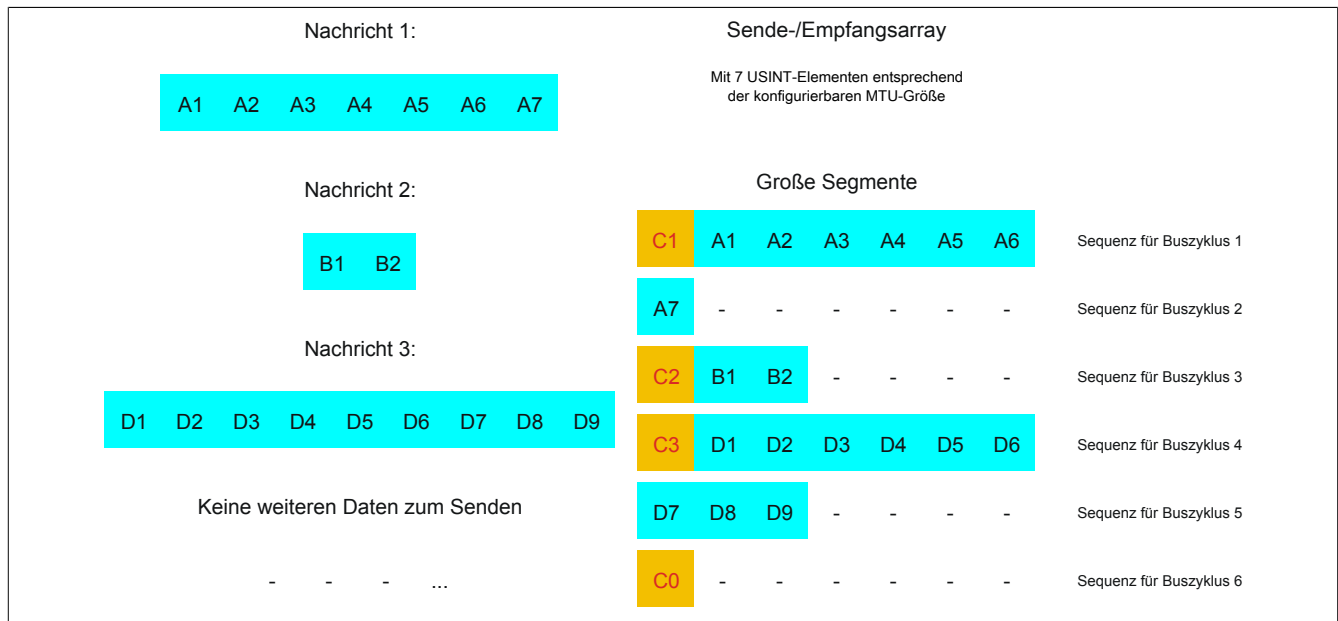


Abbildung 20: Sende-/Empfangsarray (große Segmente)

Zunächst müssen die Nachrichten in Segmente geteilt werden. Durch die Möglichkeit große Segmente zu bilden, müssen Nachrichten seltener geteilt werden, sodass weniger Controlbytes generiert werden müssen.

Große Segmente erlaubt → max. Segmentlänge 63 Bytes

- Nachricht 1 (7 Bytes)
 - ⇒ erstes Segment = Controlbyte + 7 Datenbytes
- Nachricht 2 (2 Bytes)
 - ⇒ erstes Segment = Controlbyte + 2 Datenbytes
- Nachricht 3 (9 Bytes)
 - ⇒ erstes Segment = Controlbyte + 9 Datenbytes
- Keine weiteren Nachrichten
 - ⇒ C0-Controlbyte

Für jedes gebildete Segment muss ein spezifisches Controlbyte generiert werden. Außerdem wird das Controlbyte C0 generiert, um die Kommunikation auf Standby halten zu können.

C1 (Controlbyte1)			C2 (Controlbyte2)			C3 (Controlbyte3)		
- SegmentLength (7)	=	7	- SegmentLength (2)	=	2	- SegmentLength (9)	=	9
- nextCBPos (0)	=	0	- nextCBPos (0)	=	0	- nextCBPos (0)	=	0
- MessageEndBit (1)	=	128	- MessageEndBit (1)	=	128	- MessageEndBit (1)	=	128
Controlbyte	Σ	135	Controlbyte	Σ	130	Controlbyte	Σ	137

Tabelle 7: Flatstream-Ermittlung der Controlbytes für Beispiel mit großen Segmenten

Große Segmente und MultiSegmentMTU

Beispiel

Es werden 3 unabhängige Nachrichten (7 Bytes, 2 Bytes, 9 Bytes) über eine 7-Byte breite MTU übermittelt. Die Konfiguration erlaubt sowohl die Übertragung von MultiSegmentMTUs als auch von großen Segmenten.

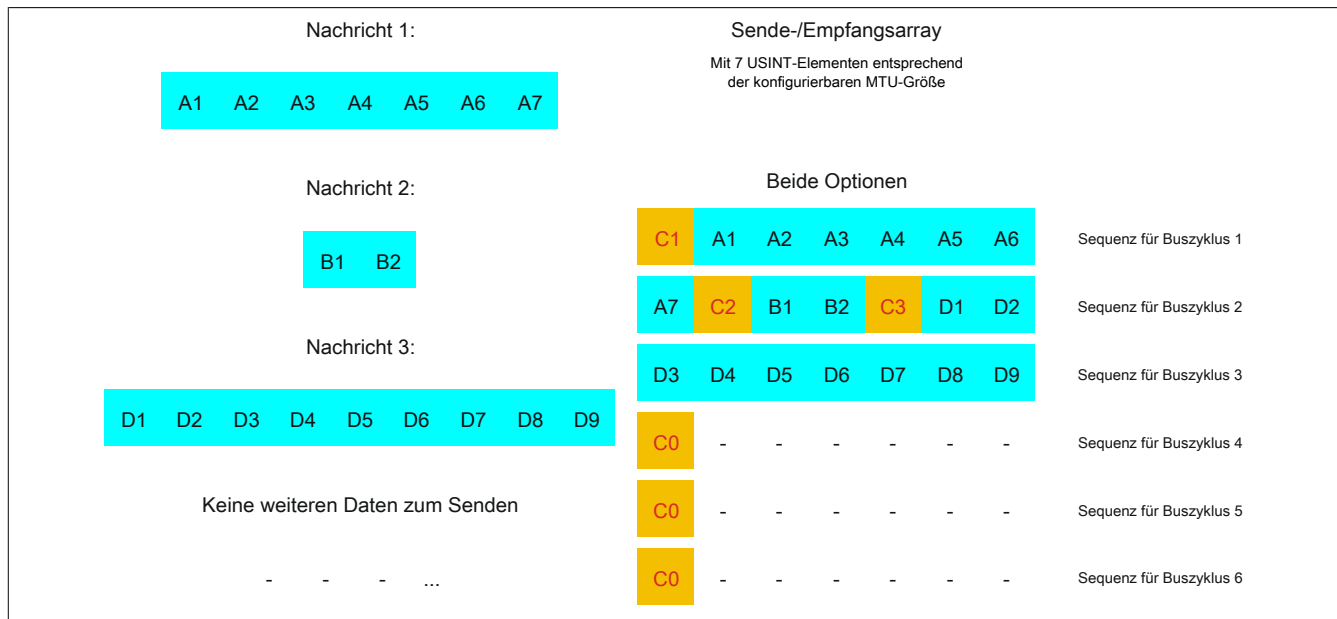


Abbildung 21: Sende-/Empfangsarray (große Segmente und MultiSegmentMTU)

Zunächst müssen die Nachrichten in Segmente geteilt werden. Wenn das letzte Segment einer Nachricht die MTU nicht komplett befüllt, darf sie für weitere Daten aus dem Datenstrom verwendet werden. Das Bit "nextCBPos" muss immer gesetzt werden, wenn das Controlbyte zu einem Segment mit Nutzdaten gehört.

Durch die Möglichkeit große Segmente zu bilden, müssen Nachrichten seltener geteilt werden, sodass weniger Controlbytes generiert werden müssen. Die Generierung der Controlbytes erfolgt auf die gleiche Weise, wie bei der Option "große Segmente".

Große Segmente erlaubt → max. Segmentlänge 63 Bytes

- Nachricht 1 (7 Bytes)
⇒ erstes Segment = Controlbyte + 7 Datenbytes
- Nachricht 2 (2 Bytes)
⇒ erstes Segment = Controlbyte + 2 Datenbytes
- Nachricht 3 (9 Bytes)
⇒ erstes Segment = Controlbyte + 9 Datenbytes
- Keine weiteren Nachrichten
⇒ C0-Controlbyte

Für jedes gebildete Segment muss ein spezifisches Controlbyte generiert werden. Außerdem wird das Controlbyte C0 generiert, um die Kommunikation auf Standby halten zu können.

C1 (Controlbyte1)			C2 (Controlbyte2)			C3 (Controlbyte3)		
- SegmentLength (7)	=	7	- SegmentLength (2)	=	2	- SegmentLength (9)	=	9
- nextCBPos (0)	=	0	- nextCBPos (0)	=	0	- nextCBPos (0)	=	0
- MessageEndBit (1)	=	128	- MessageEndBit (1)	=	128	- MessageEndBit (1)	=	128
Controlbyte	Σ	135	Controlbyte	Σ	130	Controlbyte	Σ	137

Tabelle 8: Flatstream-Ermittlung der Controlbytes für Beispiel mit großen Segmenten und MultiSegmentMTU

4.14.4.5 Die "Forward"-Funktion am Beispiel des X2X Link

Bei der "Forward"-Funktion handelt es sich um eine Methode, die Datenrate des Flatstreams deutlich zu erhöhen. Das grundsätzliche Prinzip wird auch in anderen technischen Bereichen angewandt, z. B. beim "Pipelining" für Mikroprozessoren.

4.14.4.5.1 Das Funktionsprinzip

Bei der Kommunikation mittels X2X Link werden 5 Teilschritte durchlaufen, um eine Flatstream-Sequenz zu übertragen. Eine erfolgreiche Sequenzübertragung benötigt deshalb mindestens 5 Buszyklen.

	Schritt I	Schritt II	Schritt III	Schritt IV	Schritt V
Aktionen	Sequenz aus Sendearray übertragen, SequenceCounter erhöhen	Zyklischer Abgleich MTU und Modulpuffer	Sequenz an Empfangsarray fügen, SequenceAck anpassen	Zyklischer Abgleich MTU und Modulpuffer	Prüfung des SequenceAck
Ressource	Sender (Task zum Versenden)	Bussystem (Richtung 1)	Empfänger (Task zum Empfangen)	Bussystem (Richtung 2)	Sender (Task zur Ack-Prüfung)

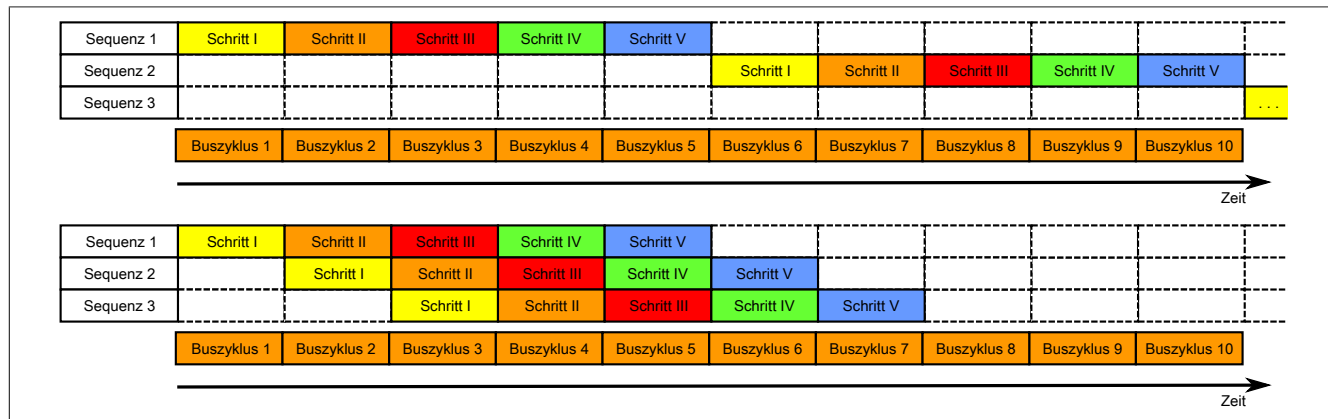


Abbildung 22: Vergleich Übertragung ohne bzw. mit Forward

Jeder der 5 Schritte (Tasks) beansprucht unterschiedliche Ressourcen. Ohne die Verwendung des Forward werden die Sequenzen nacheinander abgearbeitet. Jede Ressource ist nur dann aktiv, wenn sie für die aktuelle Teilaktion benötigt wird.

Beim Forward kann die Ressource, welche ihre Aufgabe abgearbeitet hat, bereits für die nächste Nachricht genutzt werden. Dazu wird die Bedingung zur MTU-Freigabe verändert. Die Sequenzen werden zeitgesteuert auf die MTU gelegt. Die Sendestation wartet nicht mehr auf die Bestätigung durch das SequenceAck und nutzt auf diese Weise die gegebene Bandbreite effizienter.

Im Idealfall arbeiten alle Ressourcen während jedes Buszyklus. Der Empfänger muss weiterhin jede erhaltene Sequenz bestätigen. Erst wenn das SequenceAck angepasst und vom Absender geprüft wurde, gilt die Sequenz als erfolgreich übertragen.

4.14.4.5.2 Konfiguration

Die Forward-Funktion muss nur für die Input-Richtung freigeschaltet werden. Zu diesem Zweck sind 2 weitere Register zu konfigurieren. Die Flatstream-Module wurden dahingehend optimiert, diese Funktion unterstützen zu können. In Output-Richtung kann die Forward-Funktion genutzt werden, sobald die Größe der OutputMTU vorgegeben ist.

4.14.4.5.2.1 Anzahl der unbestätigten Sequenzen

Name:

Forward

Über das Register "Forward" stellt der Anwender ein, wie viele unbestätigte Sequenzen das Modul abschicken darf.

Empfehlung:

X2X Link: max. 5

POWERLINK: max. 7

Datentyp	Werte
USINT	1 bis 7 Standard: 1

4.14.4.5.2.2 Verzögerungszeit

Name:

ForwardDelay

Mit dem Register "ForwardDelay" wird die Verzögerungszeit in μs vorgegeben. Das Modul muss nach dem Versand einer Sequenz diese Zeit abwarten, bevor es im darauf folgenden Buszyklus neue Daten in die MTU schreiben darf. Die Programmroutine zum Empfang von Sequenzen aus einem Modul kann somit auch in einer Taskklasse betrieben werden deren Zykluszeit langsamer ist als der Buszyklus.

Datentyp	Werte
UINT	0 bis 65535 [μs] Standard: 0

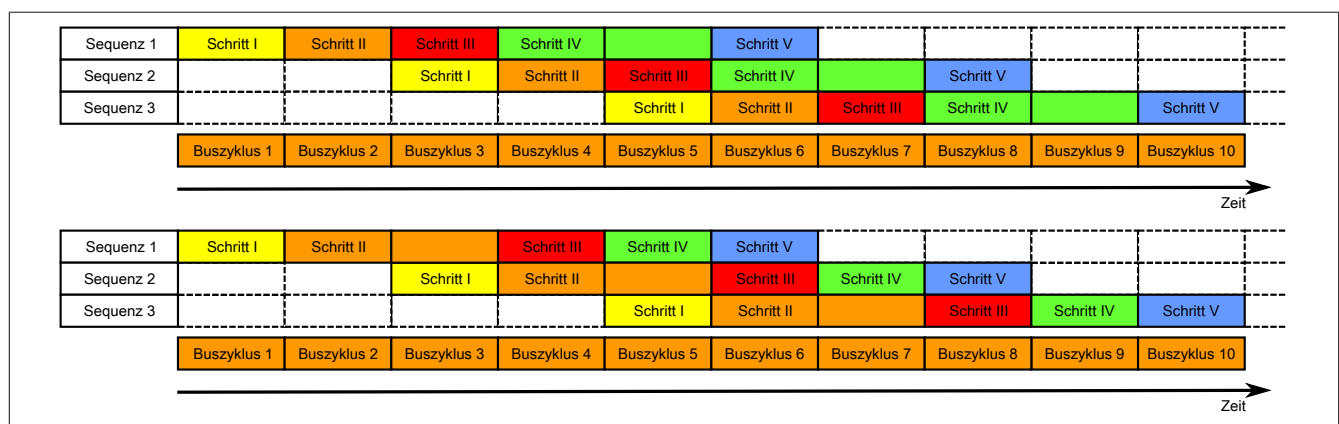


Abbildung 23: Auswirkung des ForwardDelay bei der Flatstream-Kommunikation mit Forward

Im Programmablauf muss sichergestellt werden, dass die CPU alle eintreffenden InputSequences bzw. InputMTUs verarbeitet. Der ForwardDelay-Wert bewirkt in Output-Richtung eine verzögerte Bestätigung und in Input-Richtung einen verzögerten Empfang. Auf diese Weise hat die CPU länger Zeit die eintreffende InputSequence bzw. InputMTU zu verarbeiten.

4.14.4.5.3 Senden und Empfangen mit Forward

Der grundsätzliche Algorithmus zum Senden bzw. Empfangen von Daten bleibt gleich. Durch den Forward können bis zu 7 unbestätigte Sequenzen abgesetzt werden. Sequenzen können gesendet werden, ohne die Bestätigung der vorangegangenen Nachricht abzuwarten. Da die Wartezeit zwischen Schreiben und Rückmeldung entfällt, können im gleichen Zeitraum erheblich mehr Daten übertragen werden.

Algorithmus zum Senden

Zyklische Statusabfrage: - Modul überwacht OutputSequenceCounter
0) Zyklische Prüfungen: - CPU muss OutputSyncAck prüfen → falls OutputSyncAck = 0; OutputSyncBit zurücksetzen und Kanal resynchronisieren - CPU muss Freigabe der OutputMTU prüfen → falls OutputSequenceCounter > OutputSequenceAck + 7, in diesem Fall nicht freigegeben, weil letzte Sequenz noch nicht quittiert
1) Vorbereitung (Sendearray anlegen): - CPU muss Nachricht auf zulässige Segmente aufteilen und entsprechende Controlbytes bilden - CPU muss Segmente und Controlbytes zu Sendearray zusammenfügen
2) Senden: - CPU muss aktuellen Teil des Sendearrays in die OutputMTU übertragen - CPU muss OutputSequenceCounter erhöhen, damit Sequenz vom Modul übernommen wird - CPU darf im nächsten Buszyklus erneut <i>senden</i> , falls MTU freigegeben ist
Reaktion des Moduls, weil OutputSequenceCounter > OutputSequenceAck: - Modul übernimmt Daten aus internem Empfangspuffer und fügt sie am Ende des internen Empfangsarrays an - Modul quittiert; aktuell empfangener Wert des OutputSequenceCounters auf OutputSequenceAck übertragen - Modul fragt Status wieder zyklisch ab
3) Abschluss (Bestätigung): - CPU muss OutputSequenceAck zyklisch überprüfen → Eine Sequenz gilt erst dann als erfolgreich übertragen, wenn sie über das OutputSequenceAck bestätigt wurde. Um Übertragungsfehler auch bei der letzten Sequenz zu erkennen, muss sichergestellt werden, dass der Algorithmus lange genug durchlaufen wird.
Hinweis: Für eine exakte Überwachung der Kommunikationszeiten sollten die Taskzyklen gezählt werden, die seit der letzten Erhöhung des OutputSequenceCounters vergangen sind. Auf diese Weise kann die Anzahl der Buszyklen abgeschätzt werden, die bislang zur Übertragung benötigt wurden. Übersteigt der Überwachungszähler eine vorgegebene Schwelle, kann die Sequenz als verloren betrachtet werden (das Verhältnis von Bus- und Taskzyklus kann vom Anwender beeinflusst werden, sodass der Schwellwert individuell zu ermitteln ist).

Algorithmus zum Empfangen

0) Zyklische Statusabfrage: - CPU muss InputSequenceCounter überwachen
Zyklische Prüfungen: - Modul prüft InputSyncAck - Modul prüft InputMTU auf Freigabe → Freigabekriterium: InputSequenceCounter > InputSequenceAck + Forward
Vorbereitung: - Modul bildet Controlbytes/Segmente und legt Sendearray an
Aktion: - Modul überträgt aktuellen Teil des Sendearrays in den Empfangspuffer - Modul erhöht InputSequenceCounter - Modul wartet auf neuen Buszyklus, nachdem Zeit aus ForwardDelay abgelaufen ist - Modul wiederholt Aktion, falls InputMTU freigegeben ist
1) Empfangen (InputSequenceCounter > InputSequenceAck): - CPU muss Daten aus InputMTU übernehmen und an das Ende des Empfangsarrays anfügen - CPU muss InputSequenceAck an InputSequenceCounter der aktuell verarbeiteten Sequenz angleichen
Abschluss: - Modul überwacht InputSequenceAck → Eine Sequenz gilt erst dann als erfolgreich übertragen, wenn sie über das InputSequenceAck bestätigt wurde.

Details/Hintergründe1. SequenceCounter unzulässig groß (Zählerversatz)

Fehlersituation: MTU nicht freigegeben

Wenn beim Senden der Unterschied zwischen SequenceCounter und SequenceAck größer wird, als es erlaubt ist, liegt ein Übertragungsfehler vor. In diesem Fall müssen alle unbestätigten Sequenzen mit dem alten Wert des SequenceCounters wiederholt werden.

2. Prüfung einer Bestätigung

Nach dem Empfang einer Bestätigung muss geprüft werden, ob die bestätigte Sequenz abgesendet wurde und bisher unbestätigt war. Falls eine Sequenz mehrfach bestätigt wird, liegt ein schwerwiegender Fehler vor. Der Kanal muss geschlossen und resynchronisiert werden (gleiches Verhalten wie ohne Forward).

Information:

In Ausnahmefällen kann das Modul bei der Verwendung des Forward den OutputSequenceAck um mehr als 1 erhöhen.

In diesem Fall liegt kein Fehler vor. Die CPU darf alle Sequenzen bis zur Bestätigten als erfolgreich übertragen betrachten.

3. Sende- und Empfangsarrays

Der Forward beeinflusst die Struktur des Sende- und Empfangsarrays nicht. Sie werden auf dieselbe Weise gebildet bzw. müssen auf dieselbe Weise ausgewertet werden.

4.14.4.5.4 Fehlerfall bei Verwendung des Forward

Im industriellen Umfeld werden in der Regel viele verschiedene Geräte unterschiedlicher Hersteller nebeneinander genutzt. Technische Geräte können sich gegenseitig durch ungewollte elektrische oder elektromagnetische Effekte störend beeinflussen. Unter Laborbedingungen können diese Situationen nur bis zu einem bestimmten Punkt nachempfunden und abgesichert werden.

Für die Übertragung per X2X Link wurden Vorkehrungen getroffen, falls es zu derartigen Beeinflussungen kommen sollte. Tritt beim Datentransfer z. B. eine unzulässige Prüfsumme auf, ignoriert das I/O-System die Daten dieses Buszyklus und der Empfänger erhält die letzten gültigen Daten erneut. Bei den herkömmlichen (zyklischen) Datenpunkten kann dieser Fehler oft ignoriert werden. Im darauffolgenden Zyklus wird der gleiche Datenpunkt wieder abgerufen, angepasst und übertragen.

Bei der Flatstream-Kommunikation mit aktiviertem Forward ist die Situation komplexer. Auch hier erhält der Empfänger ein weiteres mal die alten Daten, das heißt, die vorherigen Werte für SequenceAck/SequenceCounter und die alte MTU.

Ausfall einer Bestätigung (SequenceAck)

Wenn durch den Ausfall ein SequenceAck-Wert verloren geht, wurde die MTU bereits korrekt übertragen. Aus diesem Grund darf die nächste Sequenz vom Empfänger weiterverarbeitet werden. Der SequenceAck wird wieder an den mitgelieferten SequenceCounter angepasst und zum Absender zurückgeschickt. Für die Prüfung der eingehenden Bestätigungen folgt daraus, dass alle Sequenzen bis zur zuletzt Bestätigten erfolgreich übertragen sind (siehe Bild Sequenz 1, 2).

Ausfall einer Sendung (SequenceCounter, MTU)

Wenn durch den Ausfall eines Buszyklus der SequenceCounter-Wert bzw. die befüllte MTU verloren geht, kommen beim Empfänger keine Daten an. Zu diesem Zeitpunkt wirkt sich der Fehler noch nicht auf die Routine zum Absenden aus. Die zeitgesteuerte MTU wird wieder freigegeben und kann neu beschrieben werden.

Der Empfänger erhält SequenceCounter-Werte, die mehrfach inkrementiert sind. Damit das Empfangsarray korrekt zusammengestellt wird, darf der Empfänger nur Sendungen verarbeiten, die einen um eins erhöhten SequenceCounter besitzen. Die eintreffenden Sequenzen müssen ignoriert werden, das heißt, der Empfänger stoppt und gibt keine neuen Bestätigungen zurück.

Wenn die maximale Anzahl an unbestätigten Sequenzen abgesendet wurde und keine Bestätigungen zurück kommen, muss der Sender die betroffenen SequenceCounter und die dazugehörigen MTUs wiederholen (siehe Bild Sequenzen 3 und 4).

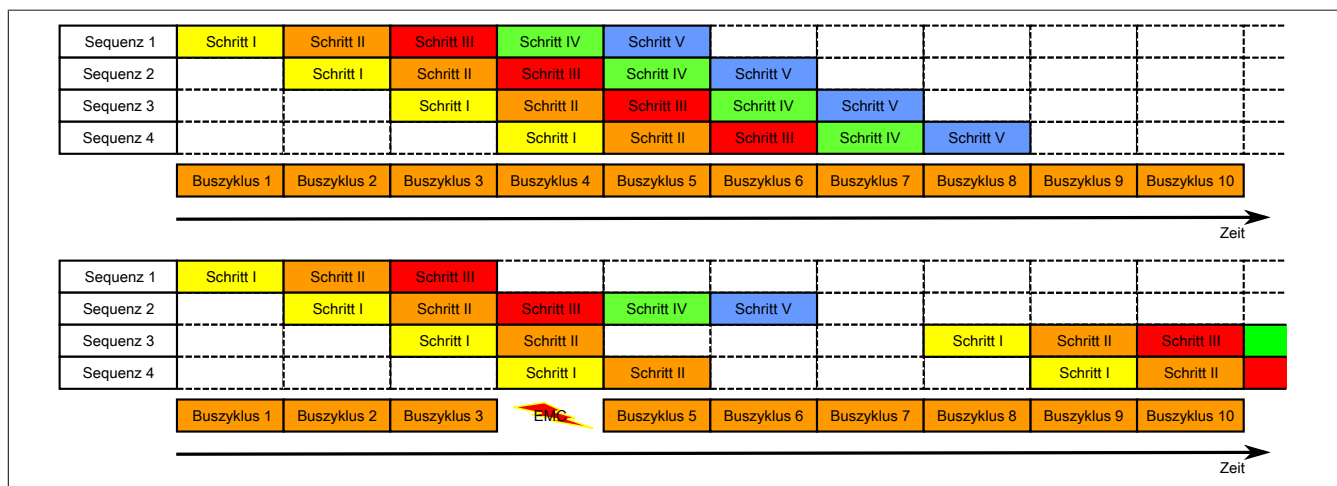


Abbildung 24: Auswirkung eines ausgefallenen Buszyklus

Ausfall der Bestätigung

Bei Sequenz 1 ging aufgrund der Störung die Bestätigung verloren. Im Schritt V der Sequenz 2 werden deshalb die Sequenzen 1 und 2 bestätigt.

Ausfall einer Sendung

Bei Sequenz 3 ging aufgrund der Störung die gesamte Sendung verloren. Der Empfänger stoppt und gibt keine Bestätigungen mehr zurück.

Der Sender sendet zunächst weiter, bis er die max. erlaubte Anzahl an unbestätigten Sendungen abgesetzt hat. Je nach Konfiguration beginnt er frühestens 5 Buszyklen später, die vergeblich abgesendeten Sendungen zu wiederholen.

4.15 Benutzung des Moduls am Feldbus

4.15.1 Bus Controller mit FeldbusDESIGNER Unterstützung

Es stehen nur das [Funktionsmodell 1 - Schneller Master](#) und [Funktionsmodell 2 - Langsamer Master](#) zur Auswahl. Die Konfiguration des Moduls erfolgt über den Automation Studio FeldbusDESIGNER. Die Implementierung muss am Master erfolgen.

4.15.2 Bus Controller ohne FeldbusDESIGNER Unterstützung

Hier steht nur das [Funktionsmodell 2 - Langsamer Master](#) zu Verfügung. Die Konfiguration und Implementierung muss am Master erfolgen.

4.15.3 B&R SG4 CPU mit Schnittstellenmodul

Diese Kombination bietet folgende Vorteile:

- Modulare Condition Monitoring Lösung möglich
- Das Automation Runtime bereitet alle Kennwerte auf (kein zusätzlicher Implementierungsaufwand)
- Die Kommunikation mit dem Master erfolgt über die Schnittstellenkarte

4.15.4 CANIO Bus Controller

Hier steht nur das [Funktionsmodell 254 - Bus Controller](#) zu Verfügung. Die Konfiguration und Implementierung muss am Master erfolgen.

4.16 Zuweisen der Taskklasse

Name:

Zykluszeit

Mit der Zuweisung der Zykluszeit zu einer Taskklasse kann der Datenaustausch zwischen der CPU und dem Modul priorisiert und den Bedürfnissen der Applikation angepasst werden. Schnellere Taskklassen führen zu einer Erhöhung des Datendurchsatzes, allerdings steigt damit die Systembelastung.

Information:

Die Verwendung einer zu langsamen Taskklasse kann dazu führen, dass neue Daten vom Modul nicht innerhalb eines Messzyklusses zur Verfügung gestellt werden können.

Die Taskklasse muss kleiner oder gleich der [maximalen Zykluszeit](#) sein.

Information:

Es handelt sich bei dieser Konfiguration um eine Treibereinstellung, welche während der Laufzeit nicht verändert werden kann!

Datentyp	Werte	Information
-	1 bis 8	Verwendete Taskklasse

4.17 Maximale Zykluszeit

Die maximale Zykluszeit gibt an, wie weit die Zykluszeit des X2X-Link erhöht werden kann, ohne dass Kommunikationsfehler oder Funktionsbeeinträchtigungen auftreten.

Maximale Zykluszeit	
Funktionsmodell 0 - Standard Funktionsmodell 1 - Schneller Master	10 ms
Funktionsmodell 2 - Langsamer Master Funktionsmodell 254 - Bus Controller	-

4.18 Minimale Zykluszeit

Die minimale Zykluszeit gibt an, bis zu welcher Zeit der Buszyklus heruntergefahren werden kann, ohne dass Kommunikationsfehler auftreten. Es ist zu beachten, dass durch sehr schnelle Zyklen die Restzeit zur Behandlung der Überwachungen, Diagnosen und azyklischen Befehle verringert wird.

Minimale Zykluszeit	
Alle Funktionsmodelle	400 µs

5 Condition Monitoring / Schwingungsanalyse

5.1 Grundlagen

Eine der wichtigsten Kenngrößen in der Maschinennutzung ist die Verfügbarkeit der Anlage. Diese Verfügbarkeit zu erhöhen bzw. langfristig sicherzustellen ist somit ein vorrangiges Ziel des Anlagenbetreibers.

Ein ungeplanter Stillstand und der daraus resultierende Produktionsausfall kann zu enormen Kosten führen. Die Einführung von Condition Monitoring hat sich als sehr gute Methode zur Unterstützung einer vorausschauenden Instandhaltung bewährt.

5.1.1 Was ist Condition Monitoring

Das Konzept des Condition Monitoring (Zustandsbeobachtung) basiert auf einer regelmäßigen Erfassung des Maschinenzustandes durch Messung aussagefähiger Größen, um sich anbahnende Probleme in der Anlage zu erkennen. Das Ziel ist es auftretende Schäden so frühzeitig zu erkennen, dass ein fehlerhaftes Maschinenteil repariert oder ausgetauscht werden kann, bevor es zu Folgeschäden bzw. einem Teil- oder Gesamtausfall der Anlage kommt.

Die konkrete Aufgabe des Condition Monitoring besteht im Erfassen und Aufbereiten von Sensordaten (z. B. Schwingungen, Temperaturen, Schmiermittelzustände, Drücke, Durchflüsse), um mittels dieser Maschinenparameter eine Beurteilung des Gesamtzustandes der Anlage abzugeben.

Abweichungen vom normalen Prozess- oder Anlagenzustand entstehen durch Fehler, für die es viele Ursachen geben kann. Wenn keine entsprechenden Gegenmaßnahmen gesetzt werden, können diese in kurzer Zeit zu einer Funktionsstörung und Ausfall der Anlage führen. Eine Überwachung eventueller Fehlerquellen durch die Auswertung von Maschinenparametern soll dazu dienen Störungen möglichst frühzeitig zu erkennen, um präventive Gegenmaßnahmen zu ergreifen. Mögliche Reaktionen können z. B. eine Fehlermeldung bzw. Warnung an das Bedienungspersonal sein oder eine automatisierte Aktion zur Störungsbeseitigung und Schadensvermeidung bis hin zur Selbstabschaltung.

Werden Condition Monitoring Techniken integriert und systematisch eingeführt, so bringt dies viele Vorteile mit sich:

- Anlagenkomponenten werden nur repariert oder ausgetauscht, wenn dies tatsächlich notwendig ist. Potentielle Fehler werden frühzeitig im laufenden Betrieb erkannt.
- Durch die Integration des Condition Monitorings in den Prozess kann die Aussagesicherheit wesentlich erhöht werden.

5.1.1.1 Badewannen- bzw. Abnutzungskurve

Jede mechanische Komponente ändert ihr Betriebsverhalten im Laufe des Betriebes und wird nach einiger Zeit defekt. Entscheidend ist es, eine solche Veränderung zu erkennen, bevor die Komponente ihre Funktion nicht mehr erfüllen kann. Dieses Verhalten kann sehr gut durch die sogenannten "Badewannenkurve" dargestellt werden. Diese gibt die wahrscheinliche Ausfallrate, über die Zeit betrachtet, an.

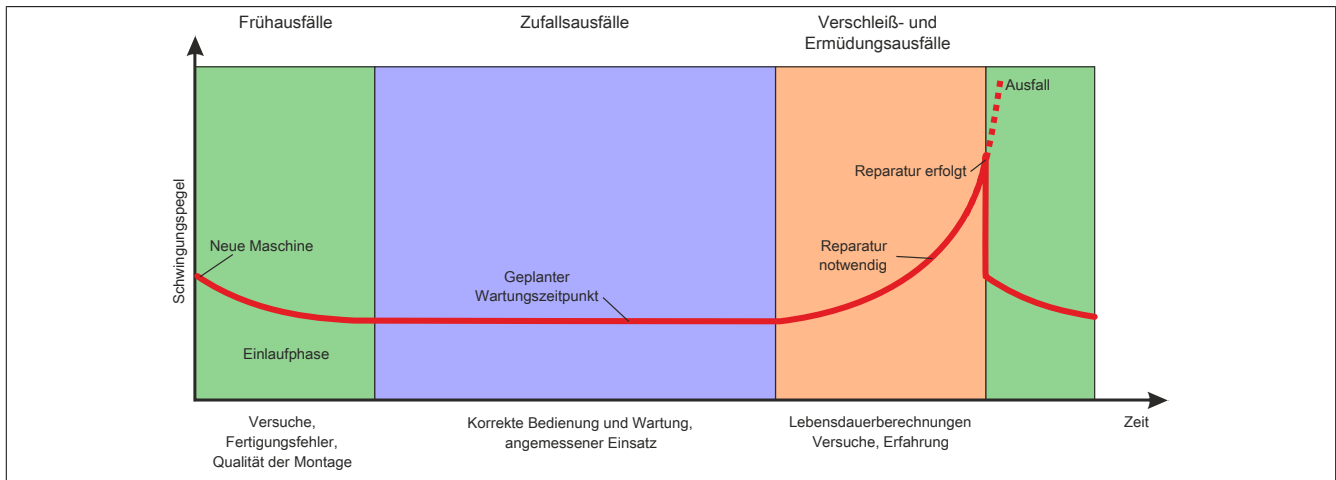


Abbildung 25: Badewannenkurve mit Einzeichnung der 3 typischen Phasen

Jede Komponente unterliegt den Gesetzmäßigkeiten dieser Kurve und es ergibt sich dabei ein typischer Verlauf.

- Der **Bereich 1 (Frühausfälle)** ist durch eine abnehmende Ausfallrate gekennzeichnet. Frühe Ausfälle werden fast ausschließlich durch Montagefehler oder Konstruktionsfehler verursacht. Besondere Sorgfalt und Qualität bei der Herstellung und Inbetriebnahme reduziert jedoch deutlich die Ausfallrate zu Beginn. Dieser Bereich erklärt auch das erhöhte Ausfallsverhalten nach einem Eingriff in ein gut funktionierendes System.
- Während des laufenden Betriebes im **Bereich 2 (Zufallsausfälle)** ist die Ausfallrate nahezu konstant. Diese zufälligen Ausfälle sind grundsätzlich schwer erfassbar und vor allem kaum beeinflussbar. Bedienungs- und Wartungsfehler tragen zur Erhöhung der Ausfallrate in diesem Bereich bei.
- Im **Bereich 3 (Verschleiß- und Ermüdungsausfälle)** steigt die Ausfallrate stark an. Die Verschleiß- und Ermüdungsausfälle werden vor allem durch langsam fortschreitende Schäden geprägt.

Wertvolle Informationen über die Ausfallwahrscheinlichkeit bietet der Trendverlauf der durch Condition Monitoring gemessenen Parameter. Dieser verhält sich meist wie die Badewannenkurve d. h. ein Anstieg der Kennwerte signalisiert eine Veränderung an der Anlage. Durch die Integration in die Automatisierung können die Kennwerte mit den Betriebsparametern abgeglichen werden. Die Beurteilung des Kurvenverhaltens und der Einbindung der Betriebsparameter erlaubt den optimalen Zeitpunkt eines Eingriffes in Abhängigkeit des Zustandes zu finden.

Weiters sind Schwingungen oft repräsentativ für den Zustand einer Maschine oder Komponente. Sie sind ein guter Hinweis um Verschleiß oder Schäden festzustellen. Ein Beispiel hierfür sind Wälzlager. Eine Oberflächenschädigung an einer Lauffläche ("Pitting") führt zu erhöhten Schwingungen am Lagergehäuse. Diese können gemessen und ausgewertet werden. Eine Erhöhung der Schwingungen während des Betriebs zeigt eine Schädigung oder auch eine Erhöhung des Verschleißes an. Durch ständige Beobachtung dieses Zustandes können Abweichungen vom Normalzustand sofort erkannt werden.

5.1.1.2 Schadensentwicklung und Schadenskette

Schadensentwicklung am Beispiel eines Wälzlagers

Die Mehrheit der Lagerschäden entwickeln sich zeitlich gesehen langsam und vor allem unbemerkt. Erst im Spätstadium der Schadenskette zeigt ein unruhiger Lauf und ein außergewöhnliches Laufgeräusch vielfach Lagerschäden an. Diese weisen auf eine Werkstoffermüdung hin, wie z. B. Abblätterung oder veränderte Radialluft durch Verschleiß.

Ist die Schädigung so weit fortgeschritten, dass sie ohne Messgerät wahrgenommen werden kann, kommt es oft zu sogenannten Spontanausfällen mit Blockieren und Bruch der Wälzlagerkomponenten.

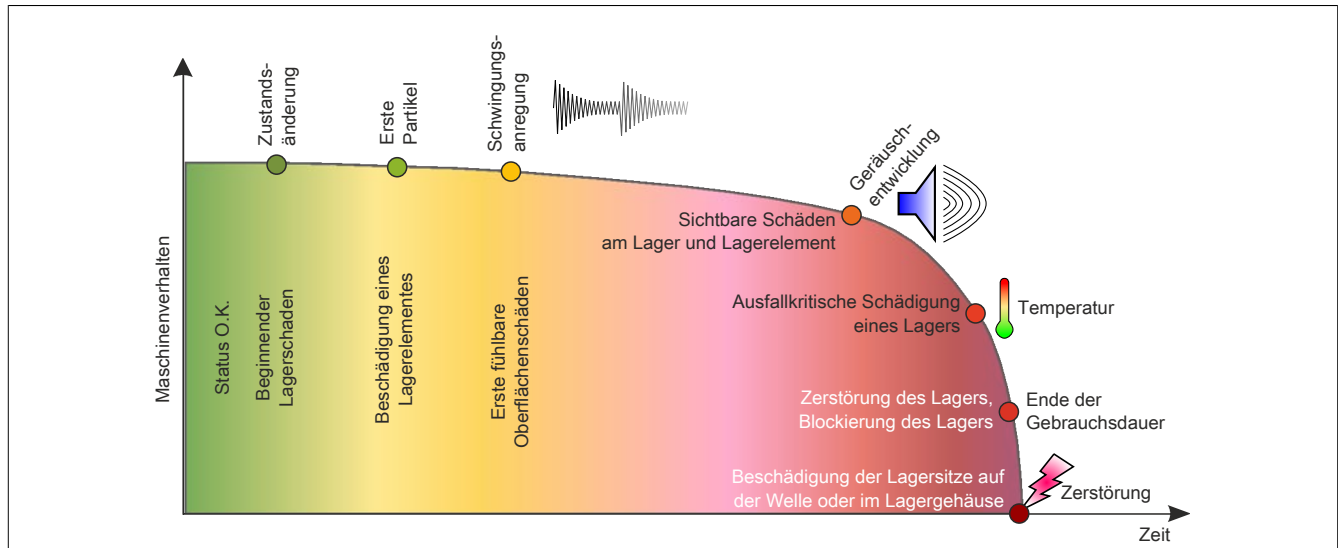


Abbildung 26: Grafische Darstellung der Schadenskette – lange Zeitschiene der Schadensentwicklung

Die in der Graphik dargestellte Schadenskette macht Condition Monitoring zu einem wichtigen Werkzeug des zustandsorientierten Anlagenbetriebes und der Instandhaltung.

Aus den durch die Sensoren erfassten Parametern wird ein Rückschluss auf den Zustand der Maschine oder deren Komponenten gezogen. Eine Verschlechterung des Zustandes von Komponenten oder Anlagenteilen zeigt sich durch das Detektieren und Beobachten des Zustandes, der Betrachtung des Trends und gegebenenfalls durch die detaillierte Analyse der gewonnenen Messdaten. Darauf basierend können gezielte Maßnahmen für die Instandhaltung gesetzt werden.

Condition Monitoring ist eine angemessene Option, wenn folgende Bedingungen zutreffen:

- Messbare Parameter, die mit einem Fehler korrelieren, müssen identifiziert und selektiert sein
- Der Fehler kann nicht durch eine Neugestaltung oder eine geänderte Gebrauchsweise verhindert werden
- Die Ereignisse führen zu zufällig auftretenden Fehlern
- Eine ausreichende Vorwarnzeit vor einem Funktionsversagen muss vorhanden sein

Die Betrachtung des Hilfsmittels Condition Monitoring darf nicht allein auf die Betrachtung der Erfassung von Betriebszuständen begrenzt werden, sondern soll als Bestandteil einer allgemeinen Asset Management Strategie in diese eingebunden werden. Um dem gerecht zu werden, sollen alle Arten des Condition Monitoring und der industriellen Diagnostik in einer Gesamtstrategie zusammengefasst werden.

Bei einer auf Ausfall orientierte Betriebsweise ("reaktive Instandhaltung") werden die Komponenten erst getauscht wenn diese Ihre Funktion nicht mehr erfüllen können. Bei einer geplanten Betriebsweise ("präventiven Instandhaltung") werden Komponenten zu einem bestimmten Zeitpunkt getauscht – unabhängig vom aktuellen Zustand.

Bei der zustandsorientierten Instandhaltung kann der Bereich der geplanten Instandhaltung deutlich eingegrenzt werden bei gleichzeitiger Reduzierung des Ausfallsrisikos.

	Vorteile	Nachteile
Reaktive Instandhaltung	- Ausnutzung der Abnutzungsreserve - Während der Gebrauchsdauer keine Kosten	- Unerwarteter Ausfall - Folgeschäden - hohe Stillstandskosten - Geringe Betriebssicherheit
Präventive Instandhaltung	- Gut planbar	- Keine Ausnutzung der Abnutzungsreserve - Erhöhtes Ausfallsrisiko nach der Wartung - Fixkosten
Zustandsorientierte Instandhaltung	- Früherkennung von Problemen - Planbarer Stillstand - Ausnutzung der Abnutzungsreserve - Hohe Betriebssicherheit - Vermeidung von Folgeschäden	- Beschäftigung mit der Thematik - Investitionskosten - Zusätzliche Kosten für Ausrüstungsmaterial zur Zustandserfassung

5.1.2 Klassisches Condition Monitoring

- Viele vorhandene Mess- und Prozessdaten werden nicht für Condition Monitoring (CM) genutzt. Verknüpfungen und Korrelationen mit Prozessgrößen und weiteren CM-Parametern können oft nur mit erheblichem Aufwand ermöglicht werden
- CM-Systeme werden als "Insellösungen" ausgeführt, d. h. eigenständige Sensorik- und Messsysteme mit eigenständiger Hard- und Software erfüllen die Aufgaben des Condition Monitoring
- Die Vielfalt der unterschiedlichen Systeme kann zu massiven Problemen bei der Bedienung der Systeme im Betrieb vor Ort führen. Unterschiedliche Softwareansätze erschweren dies noch, da für jede CM-Methode und jedes CM-Tool eine eigene Hardware und eine eigene Bedienoberfläche vorliegt.
- Teilweise notwendiges Expertenwissen ist oftmals nicht systemangepasst vorhanden. Die Komplexität mancher Konfigurationstools überschreitet das vorhandene Know-how der Anwender.

Die X20CM4810 in Verbindung mit den Standardmodulen von B&R bringt folgende Vorteile:

- Einfacher Austausch von Prozessdaten und Condition Monitoring Daten
- Einfache Einbindung der Parameter in den Gesamtprozess
- Modularer Aufbau

5.1.3 Condition Monitoring Methoden in der Übersicht

Methode	Kurzbeschreibung der Funktionsweise	Verfügbare Signale / Schnittstellen
Zustandsermittlung von Kühl- und Schmiermitteln, optische Beurteilung, Filterung, Ferrographie, Magnetdetektion, Spektroskopie, radioaktive Spurenanalyse	Quantitative Untersuchung der Verschleißprodukte (Filterung, Magnetfänge, Ferrografie, Ölspektralanalyse, Teilchenzählung), die aus Schmieröl oder Kühlflüssigkeit gewonnen werden Regelmäßige Entnahme von Proben nach definiertem Zeitplan oder betriebsstundenabhängig Quantitativer Vergleich der Proben	Durch geeignete Messsysteme können diese Condition Monitoring Methoden durch AI-, DI-Eingänge oder Bus Controlmodule im B&R System verarbeitet werden.
Thermische Diagnostik Temperatursensorik, Thermometrie, Infrarotmesstechnik	Erfassung von Temperaturen durch Sensoren mit unterschiedlichen physikalischen Wirkungsweisen Erfassung der Temperaturverteilung durch Erfassung der Infrarotstrahlung	
Schallemissionsanalyse Luftschallmessung, Impulsanalyse, Impulsdichteschwankungsanalyse, Schalldruckmessung, Schallemissionsortung	Luftschallmessung von Infraschall bis Ultraschall mittels Mikrofon Mehrdimensionaler Mikrofonaufbau zur Emissionsquellenortung Erkennung von Mikroschädigungen (Risse) etc. durch die Messung akustischer Emissionen von transienten Wellen mit hohen Frequenzen	
Vibrationsmessung Körperschallmessung, FFT Analysen, Ordnungsanalysen, Modalanalysen	Vibroakustische Diagnostik Messung des Körperschalls an Lagerstellen oder Struktur mit Beschleunigungsaufnehmern, Analyse und Auswertung	
Elektrische Parameter, Motorstromanalyse, Isolations-Widerstandsmessung	Erfassung elektrotechnischer Parameter und Auswertung in Hinsicht Condition Monitoring	Diese Condition Monitoring Methoden können durch AI-, DI- Eingänge, Bus Controlmodule oder durch B&R Antriebssysteme (unterschiedliche Funktionen von System zu System) im B&R System verarbeitet werden.

5.2 Schwingungsmesstechnik

5.2.1 Sensortechnik

Schwingungssensoren wandeln die an der zu überwachenden Maschine auftretenden mechanischen Schwingungen in elektrische Signale um.

Gemessen wird meistens der Körperschall, das ist der Schall, der sich in einem Festkörper ausbreitet.

Die im Bereich des Condition Monitoring gemessenen Beschleunigungen werden meistens mit piezoelektrischen Sensoren gemessen. Die verwendeten Schwingungssensoren nutzen den piezoelektrischen Effekt von Quarzen oder bestimmten Keramiken aus. Die eigentliche Messgröße ist eine Kraft, die zur Beschleunigung proportional ist.

Beim piezoelektrischen Effekt kommt es bei einer Krafteinwirkung auf das Piezomaterial zu einer Ladungstrennung - diese ist proportional zur Kraft und damit proportional zur Beschleunigung. Als Piezomaterial wird ein Piezoquarz oder eine Piezokeramik verwendet. Das Ausgangssignal ist eine elektrische Ladung, die in pC (picoCoulomb) angegeben wird. Um die Ladung in ein Spannungssignal umzuwandeln, ist ein Ladungsverstärker notwendig.

Die in den B&R Sensoren verwendete Integrated Electronics Piezo Electric (IEPE) Technologie verstärkt das Signal direkt im Sensor und gibt dieses als niederohmiges Spannungssignal aus. Die Empfindlichkeit wird in mV/g angegeben.

1 g = 9,81 m/s² (Erdbeschleunigung)

Information:

Piezoelektrische Sensoren können keine statischen Größen messen.

5.2.1.1 Grundsätzlicher Aufbau

Bei einem Compression-Sensor wird der piezoelektrische Kristall zwischen einer seismischen Masse und der Basis eingeklemmt. Durch die auftretende Beschleunigung erhöht oder vermindert sich die Kraft, die auf den Kristall wirkt. Je größer die eingebaute seismische Masse ist, umso größer ist das Ausgangssignal.

Sensoren dieser Bauform können mit einer sehr hohen Steifigkeit gebaut werden und haben damit eine entsprechend hohe Resonanzfrequenz.

5.2.1.2 Einflussgrößen auf den Sensor

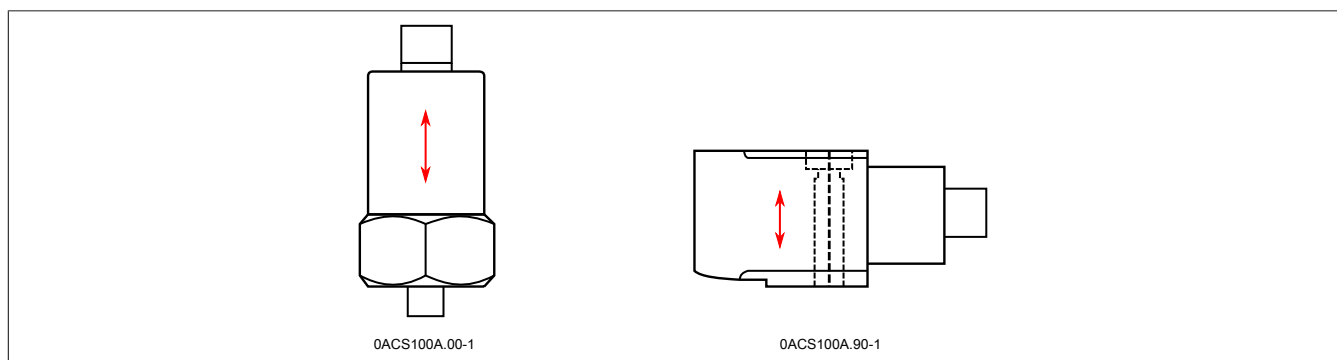
5.2.1.2.1 Einbaurichtung - Vorzugsrichtung

Grundsätzlich können Schwingungssensoren in jeder beliebigen Lage eingebaut werden. Die Einbaulage ergibt sich normalerweise durch die Messaufgabe selbst. Dennoch haben Schwingungssensoren eine bevorzugte Messrichtung. Diese ist normalerweise am Sensorgehäuse gekennzeichnet.

Auf den Sensor wirken auch Schwingungen in Querrichtung zur Einbaulage. Diese können durch entsprechende konstruktive Maßnahmen und geeignete Wahl des Piezokristalls weitestgehend kompensiert werden.

Einbausituation

Der B&R Schwingungssensor 0ACS100A.00-1 ist für Messungen in der Längsachse und der Sensor 0ACS100A.90-1 für Messungen in der Querachse gebaut.



5.2.1.2.2 Querempfindlichkeit

Grundsätzlich treten am Aufnehmer Schwingungen in allen Richtungen auf. Bevorzugt überträgt der Sensor Schwingungen in seiner Hauptrichtung, d. h. in die Richtung, die am Sensor gekennzeichnet ist. Die von dieser Hauptrichtung abweichenden Schwingungen werden vom Sensor jedoch ebenfalls registriert und je nach Bauart unterschiedlich stark im Gesamtsignal übertragen.

5.2.1.2.3 Kabel

Bei der Übertragung eines Signals über eine Verbindungsstrecke können Fehler wie Rauschen, Erdschleifen und Verzerrungen auftreten. Dieser Einfluss ist insbesondere bei der Übertragung von Ladungen von Bedeutung, da das Systemrauschen eine Funktion der Kabelkapazität ist.

Beim Einsatz der IEPE-Technologie erzeugt der Sensor aufgrund seiner internen Elektronik bereits ein hohes Spannungssignal mit einer niedrigen Quellimpedanz.

Diese Technologie eignet sich daher besonders für die Signalübertragung über lange Leitungen.

Die Versorgungseinheit liefert einen Konstantstrom für die Versorgung der IEPE-Elektronik am Sensor. Die maximale Frequenz, die ohne nennenswerte Verluste über die Messleitung übertragen werden kann, ist von der Kabellänge, der Kabelkapazität und dem Verhältnis zwischen Ausgangsamplitude und dem Konstantstrom abhängig.

Information:

Maximale Kabellänge bei Verwendung des B&R Kabels 0ACCxxx0.01-1: 100 m

5.2.1.2.4 Temperatureinfluss

Alle Piezomaterialien haben auch einen ausgeprägten pyroelektrischen Effekt. Dieser beschreibt die Änderung der elektrischen Polarisation ferroelektrischer Materialien infolge einer Temperaturänderung. Dieser Effekt ist unerwünscht, da er insbesondere bei der Schwingungsmessung zu Ladungstrennungen führt, die von der Temperaturänderung stammen und nicht durch die auftretenden mechanischen Vibrationen am Messobjekt.

Aufgrund der Bauweise kann dieser Effekt allerdings begrenzt werden. Die Störeinträge werden im Störübertragungsfaktor für Temperaturänderungen zusammengefasst.

Temperaturdrift

Die in den technischen Daten des Sensors angegebenen Werte wurden als Sprungantwort auf einen Temperatursprung bei der unteren elektrischen Grenzfrequenz $f_u = 1$ Hz ermittelt.

Veränderungen der Temperatur stören vor allem im tiefen Frequenzbereich unter $f = 10$ Hz, da diese normalerweise ebenfalls eher ein niederfrequentes Ereignis ist.

5.2.1.2.5 Einstreuungen

In der Nähe von elektromechanischen Maschinen, wie z. B. Motoren und Generatoren, kommt es aufgrund von elektromagnetischen Wechselfeldern und der damit verbundener Induktion und Magnetostraktion zu einer Rückwirkung auf den Sensor. Diese Rückwirkung ist aufgrund des verwendeten Schirmungskonzeptes und dem Einsatz der IEPE-Technologie jedoch sehr gering.

Der Einfluss wird im Störübertragungsfaktor angegeben. Dieser wird bei einer magnetischen Flußdichte von 0,01 T und einer Frequenz von 50 Hz ermittelt.

Information:

Sensoren von B&R haben eine isolierte Basis. Bei Verwendung von Fremdsensoren sollte auf die Isolierung/Schirmung geachtet werden, um Störbeeinflussungen zu minimieren. So sind z. B. Störfrequenzen mit der einfachen und doppelten Netzfrequenz möglich.

5.2.1.2.6 Linearität

Über weite Strecken des Frequenzganges verhält sich der piezoelektrische Sensor linear.

5.2.1.2.7 Frequenzgang

Der Frequenzgang des Sensors wird durch dessen mechanischen Aufbau bestimmt. Entscheidend ist hier die seismische Masse und die Steifigkeit des inneren Aufbaus, sowie dessen Konstruktionsaufbau.

Die Empfindlichkeit ist in weiten Bereichen des Frequenzganges linear, erst in der Nähe der Resonanzfrequenz steigt der Frequenzgang deutlich an. Da es in der Nähe der Resonanzfrequenz zu Fehlinterpretationen kommen kann, muss die Resonanzfrequenz entsprechend hoch sein.

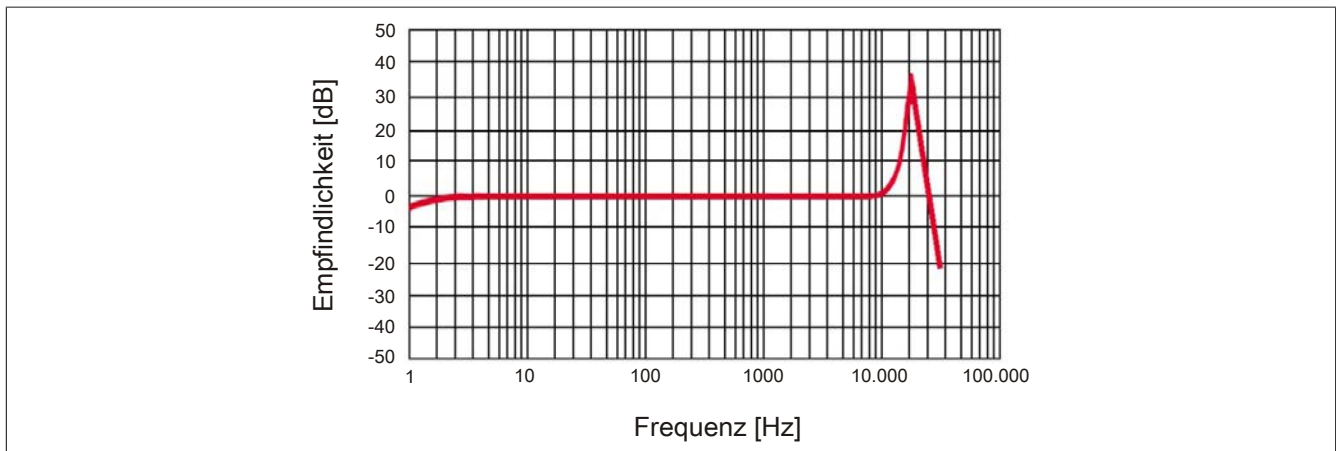


Abbildung 27: Frequenzverhalten der B&R Sensoren 0ACS100A.00-1 und 0ACS100A.90-1

5.2.1.3 Montage von Sensoren

Sensoren können über unterschiedliche Methoden mit dem Messobjekt verbunden werden. Die Montage des Sensors am Messobjekt ist entscheidend für die Qualität der Gesamtmessung.

Um alle zu messenden Frequenzanteile entsprechend auf den Sensor übertragen zu können, ist eine sehr gute Ankopplung des Sensors an das mechanische Bauteil notwendig.

Maschinenteile können temperaturbedingten Dehnungen und Verformungen unterliegen, so dass der Sensor nicht mehr über seine ganze Messfläche aufliegt. Dadurch wird die Qualität der Messung beeinträchtigt.



Abbildung 28: Temperaturverformung eines Maschinenteils (Verformung übertrieben dargestellt)

Information:

Für eine optimale Messung muss die Montagefläche glatt sein und der Sensor ganz aufliegen.

Um die Qualität der Messung entsprechend zu erhöhen ist eine starre Anbindung an das Messobjekt notwendig. Abdeckungsbleche und Kunststoffteile sind daher nicht für die Befestigung von Sensoren geeignet.

Für die Montage von Sensoren gibt es folgende Möglichkeiten :

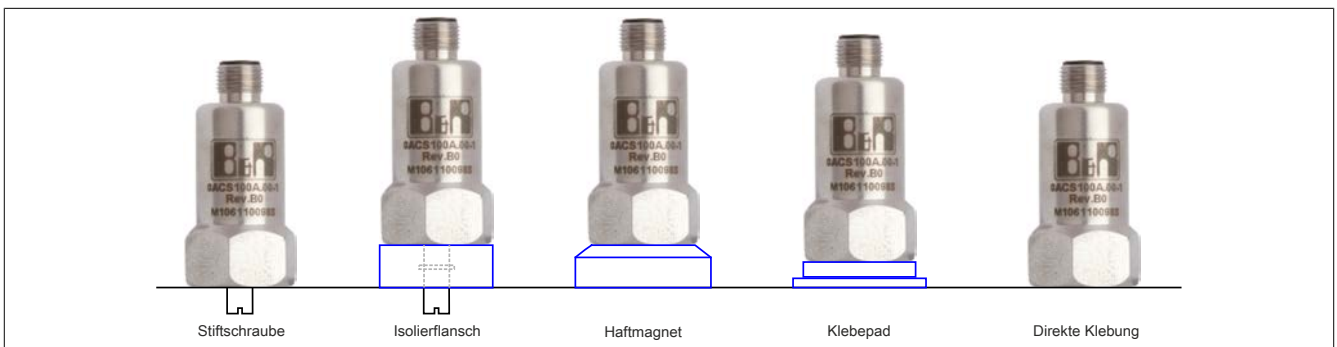


Abbildung 29: Übersicht über die Montagemethoden

Von allen Montagemethoden ist anschrauben aufgrund der niedrigsten Dämpfung zwischen Sensor und der Messoberfläche zu bevorzugen.

Die typische Sensormontage erfolgt über Verschrauben des Sensors mit dem Messobjekt mit einem sogenannten Stud. Dieser wird mit dem Sensor mitgeliefert. Studs sind speziell gefertigte Madenschrauben aus speziellen Materialien, die Vibrationen optimal übertragen.

Bei einer guten Vorbereitung der Montagestelle und einer korrekten Verschraubung des Sensors können Frequenzen bis zu etwa 10 kHz ohne nennenswerte Verluste übertragen werden.

Information:

Um eine entsprechende Qualität bei den Messungen zu erhalten empfiehlt B&R die Sensormontage durch Verschrauben.

Bei großen Entfernungen zwischen Modul und Sensor können Erdschleifen auftreten. Gegebenenfalls ist eine isolierte Montage mittels einer isolierten Montageklebeplatte durchzuführen.

Die Resonanzfrequenz wird durch zwischen den Koppelflächen angeordnete zusätzliche Elemente, wie z. B. Isolierflansch, Bundschraube, Haftmagnet und Tastspitze verringert. Die Ankopplung wird durch diese Elemente weicher. Die Unterschiede sind deutlich im Frequenzgangdiagramm zu sehen.

Verschiedene Frequenzgänge des relativen Spannungsübertragungsfaktors sind im folgenden Bild dargestellt.

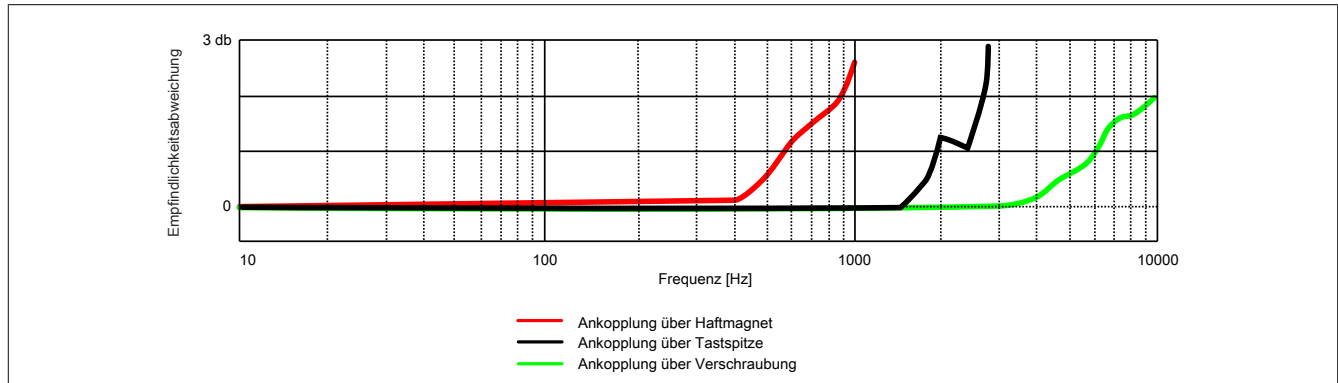


Abbildung 30: Dämpfung der unterschiedlichen Sensormontagearten

Die Art der Montage wird durch die Resonanzfrequenz und Temperatur beeinflusst. Die folgende Tabelle zeigt, wie stark sich diese Einflüsse bei den verschiedenen Montagemethoden auswirken.

	Resonanzfrequenz	Temperatur
Stiftschraube	●	●
Sekundenkleber	●	●
Bienenwachs	◐	○
Doppelseitiges Klebeband	○	◐
Haftmagnet	◐	●
Tastspitze	○	○

Einfluss auf die Montagemethode:

● hoch

◐ mittel

○ gering

5.2.1.3.1 Montagevorgehen

Abhängig von den Platzbedingungen können Sensoren direkt auf die Oberfläche des Messobjektes verschraubt werden.

SCHRITT 1: Für die Montage des Sensors wird eine möglichst glatte, flache Oberfläche benötigt. Die benötigte Größe hängt vom Sensor ab und ist dem Datenblatt zu entnehmen.

Für senkrechte Einbaulage:

siehe "[Abmessungen](#)" auf Seite 166

Für waagrechte Einbaulage:

siehe "[Abmessungen](#)" auf Seite 169

SCHRITT 2: Für die Befestigung der B&R Sensoren wird eine M8 Sacklochbohrung benötigt.

SCHRITT 3: Um das Übertragungsverhalten noch zu verbessern, kann zwischen der Objektoberfläche und der Sensormontagefläche eine dünne Schicht Silikonfett aufgetragen werden. Dies ist im überwiegenden Teil der Anwendungen nicht erforderlich und nur bei Messungen von besonders hochfrequenten Schwingungen sinnvoll.

SCHRITT 4: Der Sensor sollte mit einem Moment von ca. 8 Nm bei Verwendung des M8x1 Gewindes angezogen werden. Gegebenenfalls kann der Sensor gegen unbeabsichtigtes Lösen mittels Verklebung gesichert werden.

5.2.1.3.2 Positionierung der Sensoren

Um die Schallausbreitung von Schäden optimal messen und erkennen zu können, ist die Position des Sensors sorgfältig auszuwählen. Die ideale Position zum Montieren des Sensors an der Mechanik ist jedoch häufig nicht erreichbar, aber auch nicht immer notwendig. Durch die Schallausbreitung im gesamten Körper werden die Schallwellen des Schadens an den verschiedenen Stellen mit unterschiedlicher Intensität (Amplitude) gemessen (grüne Pfeile). Nach einer elastischen Verbindung ist keine gültige Messung mehr möglich (roter Pfeil).

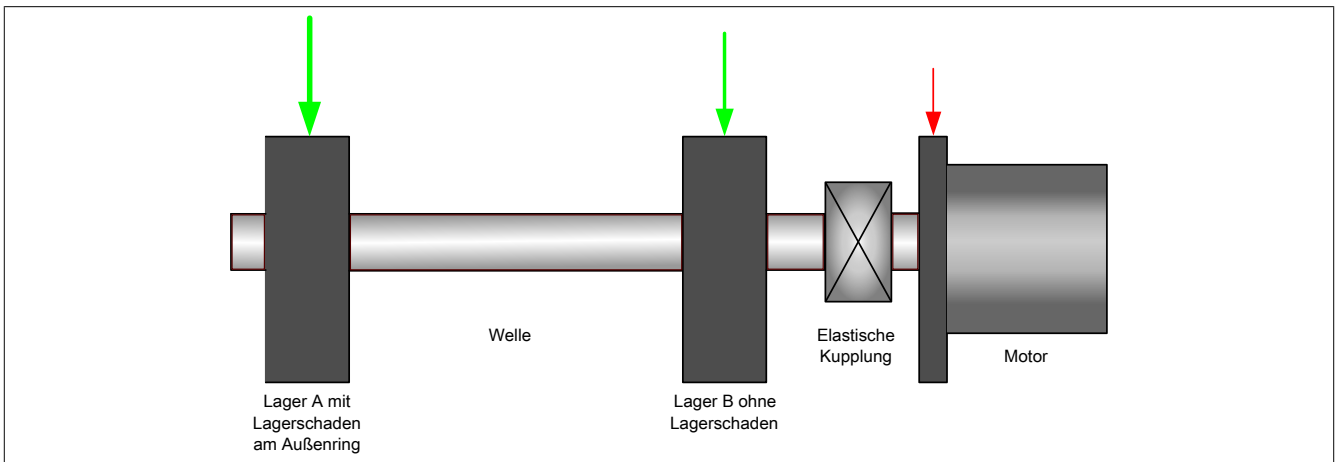


Abbildung 31: Geeignete und ungeeignete Sensorbefestigungen

5.2.2 Schwingungen - Übersicht über die Messung des Körperschalls

5.2.2.1 Schwingungen

Schwingungen sind in der Natur sehr häufig auftretende Bewegungsformen. Die harmonische Schwingung ist neben der gleichförmigen Bewegung und der gleichmäßig beschleunigten Bewegung der dritte Grundtyp der Bewegungsformen. Eine Schwingung bzw. Vibration ist eine periodische, d. h. in gleichen Zeiten sich wiederholende Bewegung eines Körpers um seine Ruhe- oder Gleichgewichtslage.

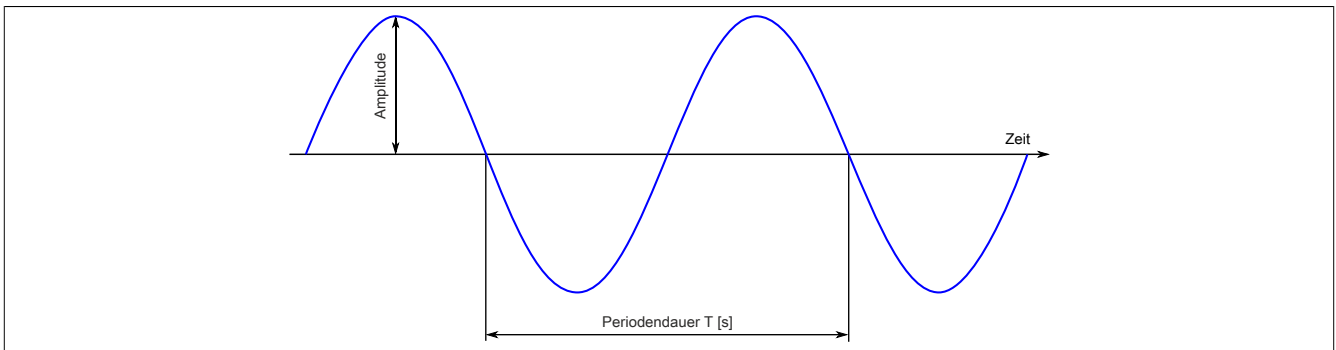


Abbildung 32: Abbildung einer grundlegenden Schwingung

Wird ein festes Medium durch einen Stoß angeregt, so breitet sich darin Körperschall aus. Dieser besteht aus zusätzlichen Frequenzen, welche durch Form und Material bestimmt werden (z. B. Gong, Betonblock).

Ein Teil der Energie des Körperschalles wird durch die umgebende Atmosphäre in Luftschall umgewandelt.

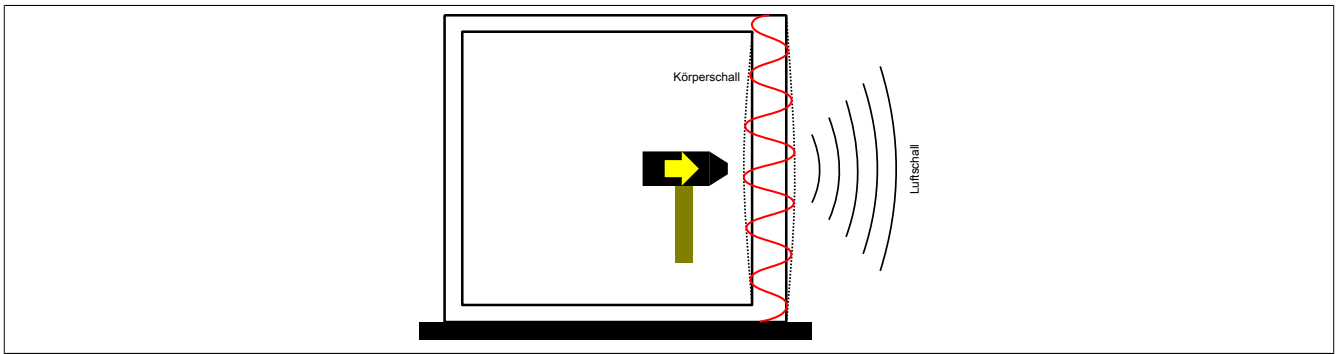


Abbildung 33: Körperschallausbreitung

Die Messung und anschließende Analyse der mechanischen Schwingungen an feststehenden und rotierenden Teilen von Maschinen, Trägerkonstruktionen und Rohrleitungen hat sich als technisch gut mögliches und praktisch anwendbares Überwachungsverfahren durchgesetzt.

Am Gehäuse der Maschine wird die absolute Lagerschwingung gemessen. Dies sind die Bewegungen des Gehäuses relativ zu einem fixen Bezugspunkt im Raum.

Mechanische Schwingungen sind ein guter Parameter für beginnende Fehler und Schäden und können für die Maschinendiagnose genutzt werden.

Schwingungen werden von einer Vielzahl von Erregern erzeugt und überlagern sich gegenseitig. Die Größe (Amplitude) der Schwingung hängt unter Anderem von folgenden Faktoren ab: die Dämpfung durch Fügestellen oder Schmiermittel, die Steifigkeit der Bauteile, Gehäuse und Fundamente und vielem mehr.

Schädigungen sind immer die Folge von Beanspruchungen. Treten zyklische Beanspruchungen auf, können sie durch die Erregerfrequenz und ihre Intensität charakterisiert werden.

Resonanzfrequenz

Jeder Maschine hat sogenannte Resonanzfrequenzen. Diese müssen beim Betrieb beachtet werden, da bei diesen Frequenzen die Amplitude der Schwingung sehr stark ansteigt, welche die Mechanik belasten. Treten für längere Zeit harmonische Schwingungen in der Nähe der Resonanzfrequenz auf, kann dies zur sogenannten "Resonanzkatastrophe" führen, die eine Zerstörung des betroffenen Teiles zur Folge hat.

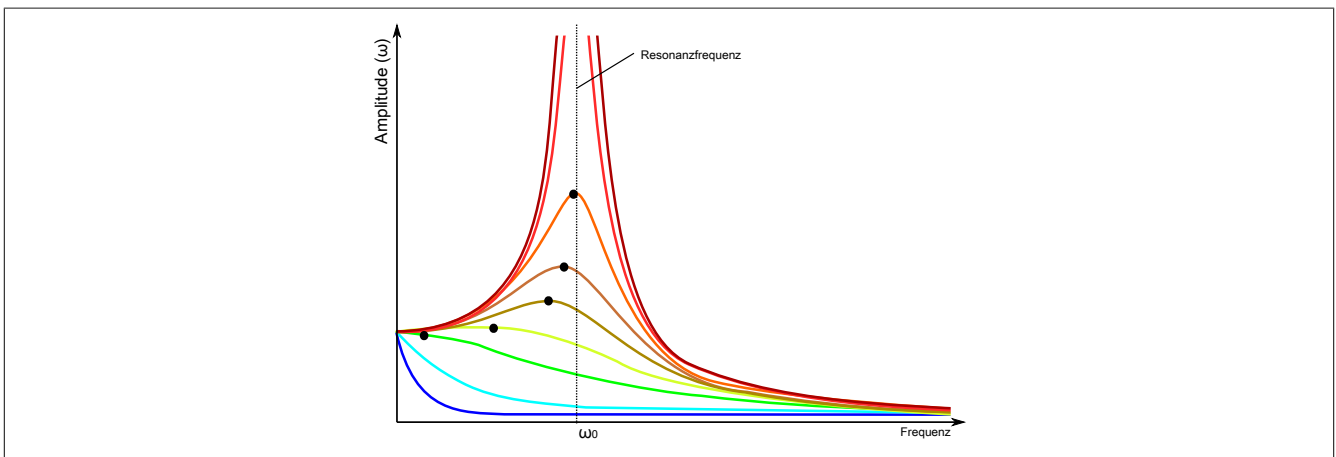


Abbildung 34: Zunahme der Amplitude in der Nähe der Resonanzfrequenz

5.2.2.1.1 Ursachen von Schwingungen

Unwucht

Nach DIN ISO 1925¹⁾ ist in einem rotierenden System eine Unwucht vorhanden wenn, als Folge von nicht ausgeglichenen Fliehkräften, Kräfte oder Schwingbewegungen auf die Lager übertragen werden.

Unwucht an einem rotierenden Körper verursachen nicht nur Kräfte auf Lager und Fundamente, sondern auch Schwingungen in der Maschine. Diese Schwingungen haben harmonischen Charakter. Die Erregerfrequenz entspricht der Drehfrequenz des unwuchtbehafteten Rotors.

Ausrichtefehler

Die Hauptaufgabe von Kupplungen ist die Verbindung von 2 Wellen, sodass sich ein statisch bestimmtes Gesamtsystem ergibt. Neben der Übertragung des Drehmomentes gleichen Kupplungen bis zu einem bestimmten Maße auch Fluchtungsfehler (radial, axial, Winkel) aus. Übersteigt jedoch die Fehlausrichtung die Kompensationsfähigkeit der Kupplung, entstehen zusätzliche Belastungen wie erhöhte Lagerkräfte, Wellenzwangsverformungen und Axialkräfte für die beteiligten Maschinenelemente.

Die entstehenden Schwingungen haben einen harmonischen Charakter und sind an die Drehfrequenz der fehlausgerichteten Welle und den Vielfachen dieser Frequenz gebunden. Die Fehlausrichtung kann bei der Drehfrequenz des fehlausgerichteten Teiles gemessen werden oder bei deren harmonischen Schwingungen.

Stöße

Fremdkörper, lose und anschlagende Teile bewirken Stöße zwischen den sich drehenden und ruhenden Teilen. Diese Stöße wiederholen sich periodisch einmal oder mehrmals pro Umdrehung der Welle.

Die Stoßwiederholffrequenz entspricht der Drehfrequenz der Welle bzw. deren harmonischen Frequenz.

Wälzlagerschäden

Die meisten Lagerschäden beginnen durch Veränderungen an den Oberflächen (Pittings). Bei Überrollung der Schädstellen am Innenring, Außenring, Käfig oder Wälzkörper kommt es zu impulsförmigen Stößen, welche die Struktur des Lagers und deren Anbauteile zum Schwingen anregen.

Jeder dieser Stöße zeigt sich im Schwingungssignal durch den typischen Verlauf einer Stoßfolge. Aus diesen Messungen können Kennwerte gewonnen werden, die einen Rückschluss auf den Zustand des Lagers zulassen.

Die Erregerfrequenzen bei Innenring-, Außenring-, Käfig- und Wälzkörperschäden werden von den Lagerherstellern angegeben.

Magnetische Induktion

Ein rotierendes Magnetfeld erzeugt Wechselkräfte im Stator der Maschine. Diese elektrisch-magnetische Anregung verursacht oft schwer feststellbare Schwingungen an Elektromotoren.

Häufig führen auch Umrichter zu einer Einstreuung von Schwingungen, deren Ursache elektrisch-magnetischer Natur ist.

5.2.2.1.2 Auswirkungen

Maschinen und Apparate mit bewegten Teilen verursachen mechanische Schwingungen. Die Auswirkungen auf den Aufstellort und auf die Umgebung sind Erschütterungen und Körperschall und damit oftmals verbunden eine störende Geräuscentwicklung.

Erhöhte Vibrationen können zu Funktionsstörungen an der Maschine insbesondere an Mess- und Regeleinrichtungen führen. Durch das "Mitschwingen" der Messmittel kommt es zu Fehlmessungen und damit verbunden zu Qualitätseinbußen in der Fertigung.

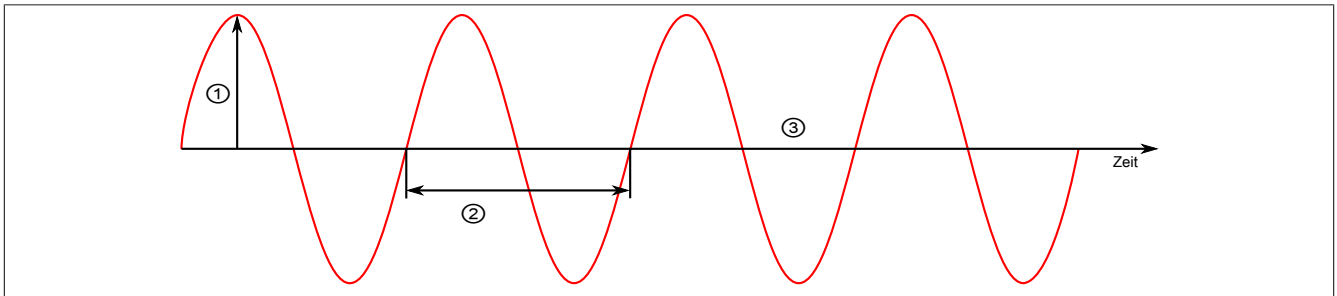
Zusätzlich entstehen Belastungen an den Bauteilen der Maschine. Ungewollte Vibrationen führen zu erhöhtem Verschleiß mit teils plastischer Verformung von Bauteilen und vermehrter Rissbildung bis hin zum Bruch.

Spürbare Schwingungen werden durch das Gleichgewichtsorgan und den Tastsinn wahrgenommen. Andauernde Belastungen können zu einer Beeinträchtigung der Arbeitsleistung und des Wohlbefindens, bis hin zu gesundheitlichen Schäden, führen.

¹⁾ DIN ISO 1925: Ausgabe:1996-11 Mechanische Schwingungen - Auswuchttechnik - Begriffe (ISO 1925:1990 + AMD 1:1995)

5.2.2.1.3 Parameter einer Schwingung

	Parameter	Erklärung	Zeichen	Formel
①	Amplitude	Größte Auslenkung aus der Ruhelage	A	
②	Periodendauer	Minimale Zeitspanne für eine vollständige Schwingung, nach der ein Körper wieder seine Ausgangslage und Ausgangsgeschwindigkeit erreicht hat.	T	
③	Ruhelage	Position, in der sich der nicht ausgelenkte Körper befindet	s_0	
	Frequenz	Zahl der Schwingungen pro Zeiteinheit	f	$f = 1 / T$
	Winkelgeschwindigkeit	Änderung des Winkels im Bogenmaß mit der Zeit	ω	$\omega = 2 \cdot \pi \cdot f$



5.2.2.1.4 Größen mechanischer Schwingungen

Die Amplitude einer Schwingung kann durch die Größen Weg (s) und Geschwindigkeit (v) oder Beschleunigung (a) definiert werden.

Diese 3 Größen stehen in festen Beziehungen zueinander und können von einer Größe in eine Andere mittels einfacher Berechnungen umgewandelt werden.

Gemessen wird von den B&R Sensoren die Schwingbeschleunigung. Die Einheit ist m/s^2 , oftmals wird die Beschleunigung auch mit der Einheit g ($1g = 9,81 \text{ m/s}^2$) für die Erdbeschleunigung angegeben.

Für manche Diagnose ist jedoch die Schwinggeschwindigkeit oder der Schwingweg aussagekräftiger. Dann kann die Beschleunigung durch Integration in die Schwinggeschwindigkeit übergeführt werden. Der Weg kann aus der Beschleunigung durch 2-malige Integration errechnet werden.

Mathematischer Zusammenhang

Schwingweg

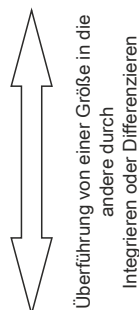
$$s = \hat{s} \cdot \sin(\omega t + \varphi)$$

Schwinggeschwindigkeit

$$v = \frac{ds}{dt} = \hat{s} \cdot \omega \cdot \cos(\omega t + \varphi)$$

Schwingbeschleunigung

$$a = \frac{d^2s}{dt^2} = \frac{dv}{dt} = -\hat{s} \cdot \omega^2 \cdot \sin(\omega t + \varphi)$$



Information:

Der Schwingweg wird vom Modul nicht berechnet.

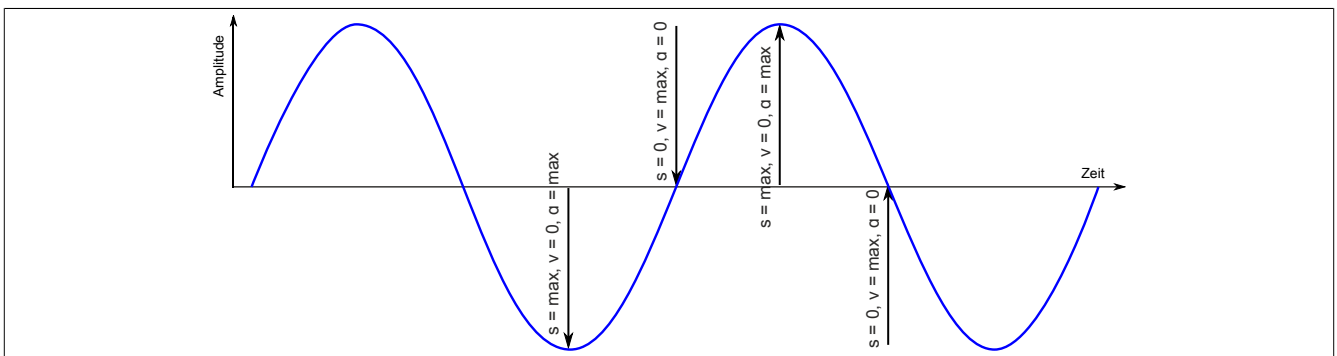


Abbildung 35: Zusammenhang s-v-a

Eine harmonische Schwingung kann durch die Amplitude, die Frequenz und den Phasenwinkel eindeutig beschrieben werden.

- Die Amplitude in Weg, Geschwindigkeit oder Beschleunigung gibt den Augenblickswert an.
- Die Frequenz beschreibt wie oft sich eine Schwingung innerhalb einer Sekunde ändert. In der Schwingungsdiagnose spielt diese eine große Rolle, da viele Frequenzen einem Verursacher zugeordnet werden können.
- Die Phasenlage beschreibt den Startpunkt der Schwingung. In den meisten Fällen handelt es sich um ein Gemisch von Schwingungen, daher ist der Phasenwinkel normalerweise nicht wichtig.

5.2.2.2 Die Fast-Fourier-Transformation (FFT)

Schwingungssignale bestehen im Allgemeinen aus einer Vielzahl von gleichzeitig auftretenden Einzelschwingungen die sich überlagern. Die einzeln auftretenden Frequenzen sind aus dem Zeitdiagramm nicht unmittelbar ersichtlich.

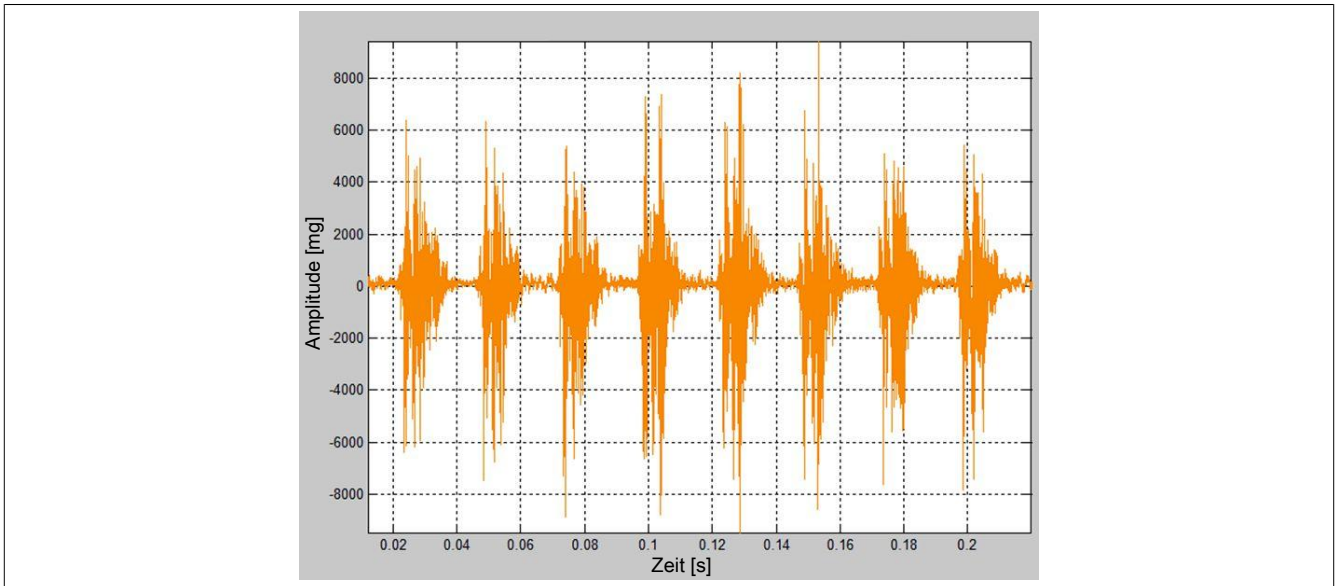


Abbildung 36: Zeitdiagramm einer Schwingung

Zur Analyse eines Gemisches von Schwingungen unterschiedlicher Frequenzen eignet sich die Berechnung eines Linienspektrums, in welchem alle beteiligten Schwingungen mit ihren Frequenzen und Amplituden als jeweils eine Linie dargestellt sind.

Im Umfeld des Condition Monitoring sind Spektren eine wertvolle Unterstützung bei der Suche nach der Ursache eines Fehlers. Viele Frequenzen können einzelnen Bauteilen zugeordnet werden und lassen damit einen Rückschluss auf die geschädigte Komponente zu.

5.2.2.2.1 Addition von Sinusschwingungen zur Signalerzeugung

Die folgenden Abbildungen zeigen, wie ein Rechtecksignal durch Überlagerung von sinusförmigen Schwingungen erzeugt wird.

Sinusschwingung mit 1 Hz und einer Amplitude von 1.

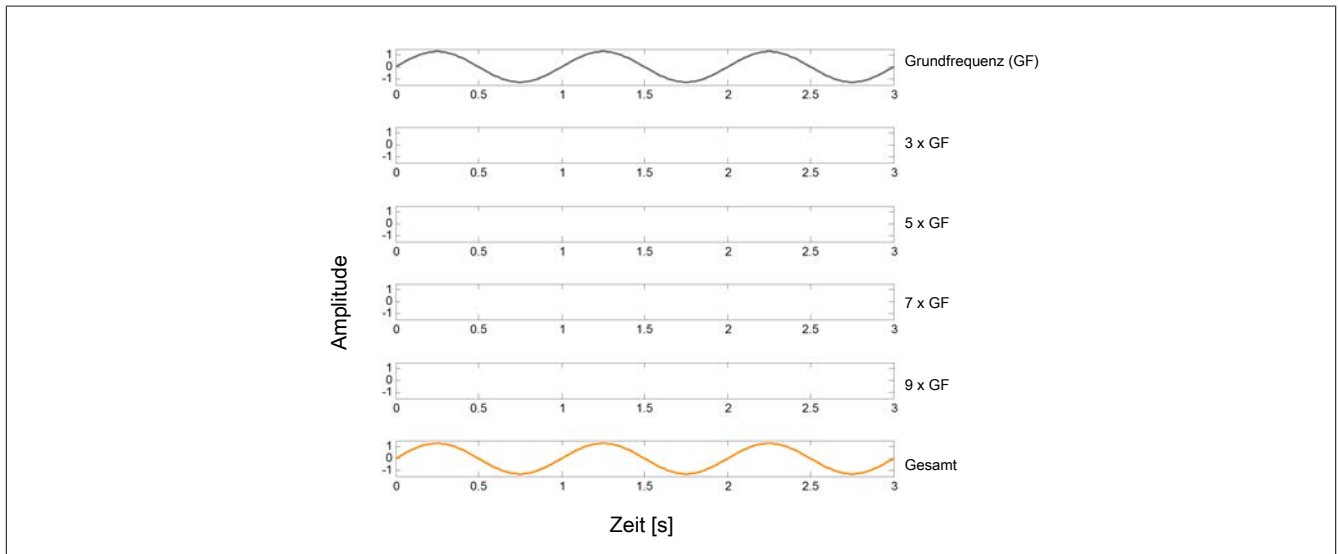


Abbildung 37: Reine Sinusschwingung

Sinusschwingung mit 1 Hz und einer Amplitude von 1 und Sinusschwingung mit 3x der Grundfrequenz, d. h. 3 Hertz und kleinerer Amplitude.

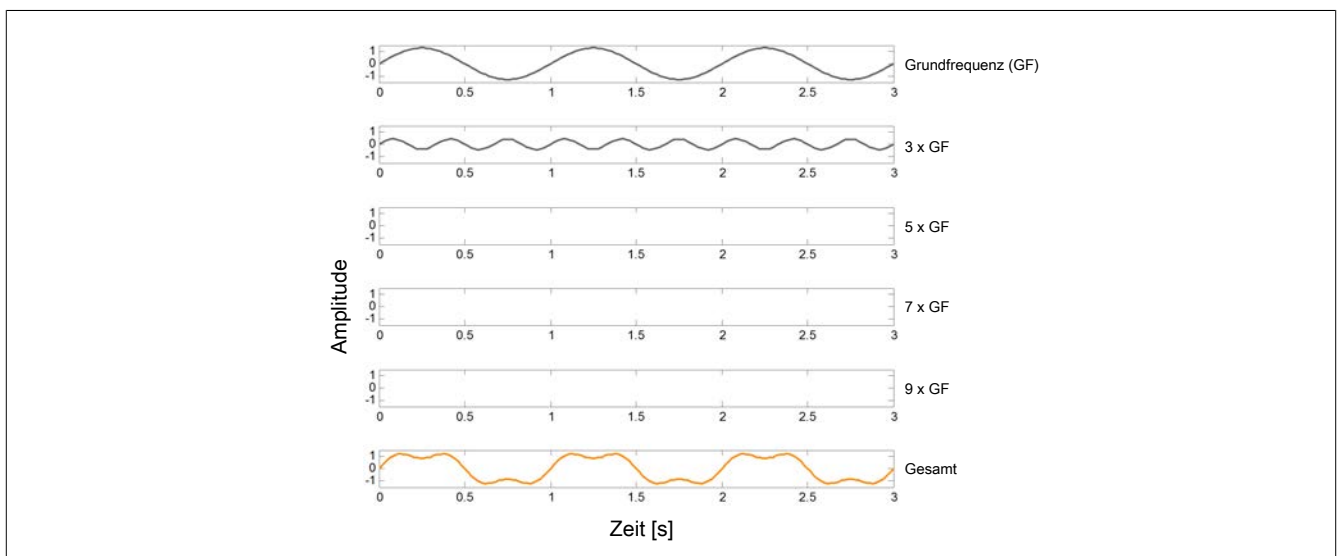


Abbildung 38: Sinusschwingung mit einer Oberwelle

Sinusschwingung mit 1 Hz und einer Amplitude von 1 und Sinusschwingungen mit 3x, 5x, 7x und 9x der Grundfrequenz und kleinerer Amplitude.

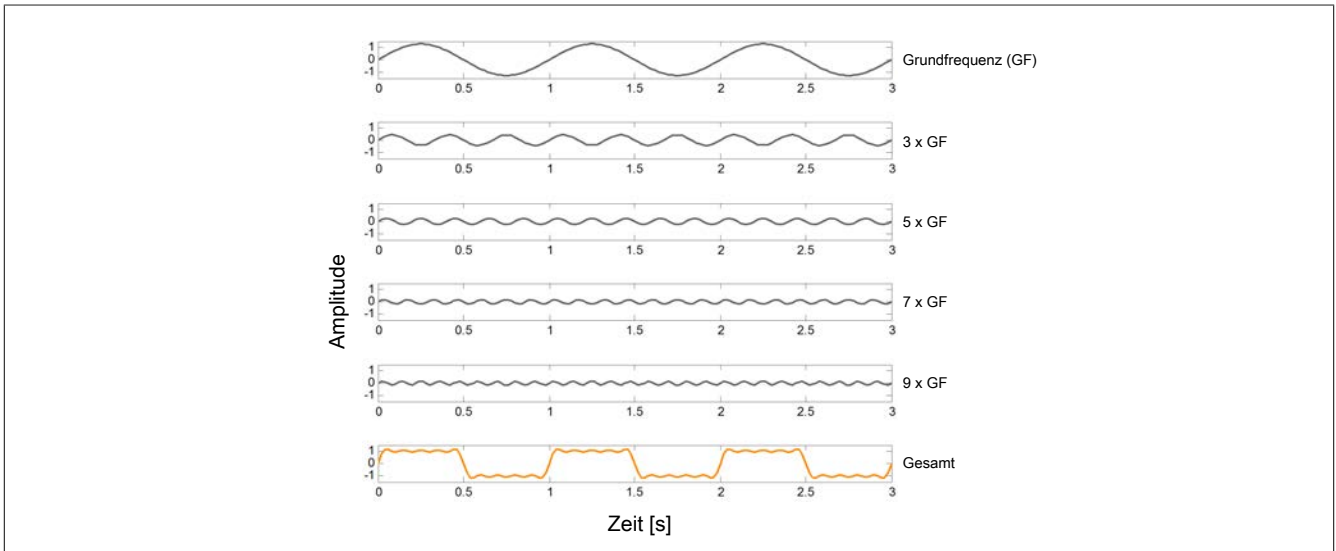


Abbildung 39: Sinusschwingung mit mehreren Oberwellen

5.2.2.2 Allgemeine Beschreibung

Die Fourier Transformation ist die Grundlage der Frequenzanalyse. Sie geht davon aus, dass sich jede harmonische Schwingung in beliebig viele Sinus- und Kosinusschwingungen zerlegen lässt, deren Summe die ursprüngliche Schwingung nachbildet. Dementsprechend lassen sich verknüpfte Einzelschwingungen wieder "zerlegen".

Der wahrscheinlich bekannteste Begriff im Zusammenhang mit der Signalverarbeitung bzw. Frequenzanalyse ist die schnelle Fouriertransformation, auch "FFT" oder "Fast-Fourier-Transformation" genannt.

Um die einzelnen Teilschwingungen in Amplitude und Frequenz beurteilen zu können, wird das digitalisierte Zeitsignal in ein Frequenzspektrum umgewandelt. Dazu wird dem Signal ein kleiner Ausschnitt entnommen, das sogenannte Zeitfenster. Daraus wird mit Hilfe des FFT-Algorithmus das Frequenzspektrum errechnet, in dem alle beteiligten Schwingungen mit ihren Frequenzen und Amplituden als einzelne Linien im Linienspektrum dargestellt sind.

Beispiel

Bei einem einfachen Sinussignal mit einer konstanten Frequenz wird im Frequenzspektrum nur eine einzige Linie dargestellt.

5.2.2.2.3 Zu beachten bei der FFT

5.2.2.2.3.1 Fensterfunktionen

An den Zeitfenstergrenzen des entnommenen Ausschnittes können je nach Signalstruktur und Randbedingungen Unstetigkeiten auftreten. Diese spiegeln Schwingungsanteile vor, die in der Realität gar nicht existieren.

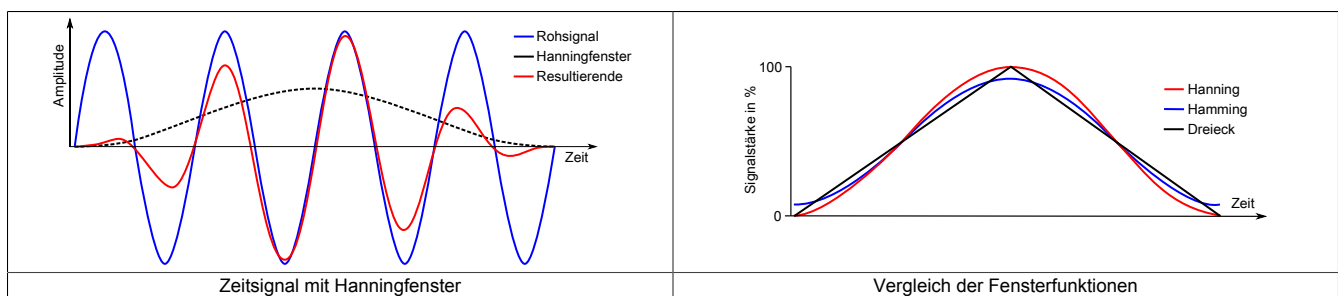
Diese Unstetigkeiten entstehen dann, wenn die Periode der Abtastung nicht einen ganzzahligen Vielfachen der Periode des Zeitsignals entspricht. Dies tritt praktisch bei jedem gemessenen Signal auf, da sich dieses aus einer Vielzahl von Signalen mit unterschiedlichen Periodendauern zusammensetzt.

Zur Unterdrückung dieser Unstetigkeiten werden sogenannte Fensterfunktionen verwendet. Dabei wird das Eingangssignal mit der Fensterfunktion multipliziert und das entstehende Signal der Fourier Transformation zugeführt.

Bekannte Fensterfunktionen sind:

- Dreieckfenster
- Hanningfenster
- Hammingfenster

Alle diese Funktionen haben gemeinsam, dass sie an den Rändern gegen Null gehen, sodass die periodische Fortsetzung nun keine Sprungstellen mehr besitzt.



Information:

Im Modul X20CM4810 wird das Hanningfenster verwendet.

5.2.2.2.3.2 Abtastung

Unter Abtastung bzw. Sampling versteht man die Erfassung eines analogen Messwertes in bestimmten Zeitabständen.

Dabei wird zu bestimmten Zeitpunkten der genaue Spannungspegel des Signals erfasst und gespeichert. Die Abstände Δt (Delta t) zwischen den Erfassungspunkten nennt man Abtastintervall.

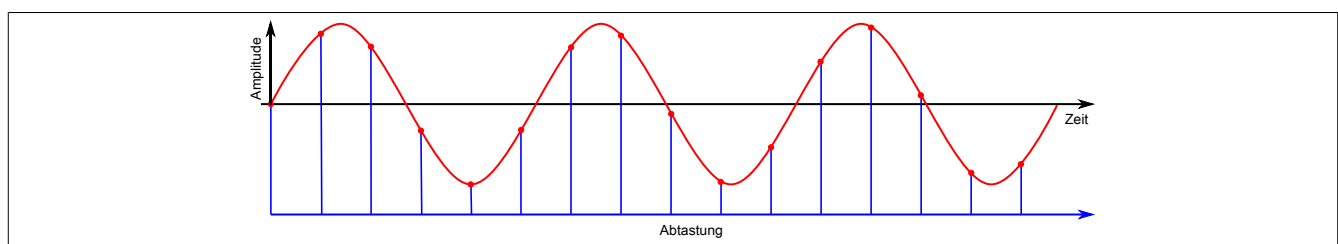


Abbildung 40: Abtastung einer Kurve

Ist die tatsächliche Abtastrate um ein Vielfaches höher als die theoretisch erforderliche Abtastrate, spricht man von "Oversampling". Das Reduzieren der Abtastrate auf die benötigte Rate wird "Downsampling" genannt.

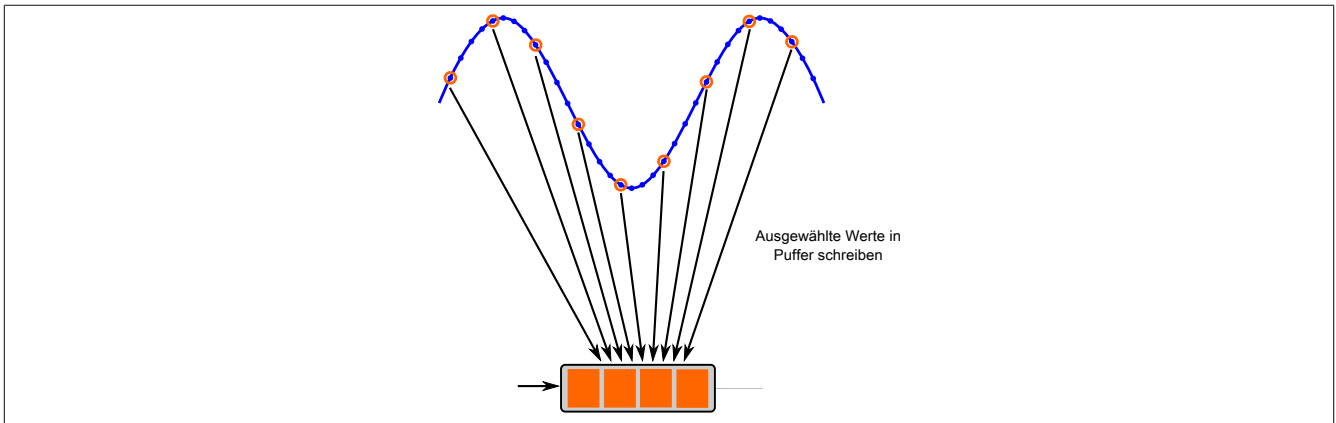
Wendet man die Fourier-Transformation unter Berücksichtigung des Abtasttheorems auf ein reines Sinussignal an, dessen Frequenz genau einer Stützstelle (= ganzzahliges Vielfaches der Frequenzauflösung) im Frequenzspektrum entspricht, so wird dieses Signal als eine einzelne Linie im Spektrum dargestellt. Liegt die Frequenz der Sinusschwingung hingegen nicht auf einer Stützstelle im Frequenzspektrum, so resultieren daraus 2 Linien an den beiden benachbarten Stützstellen, deren vertikale Position zueinander indirekt proportional zur Frequenzabweichung von der jeweiligen Stützstelle ist.

In der Praxis wird selten ein reines Sinussignal auftreten. Vielmehr wird sich ein Signal aus einer Vielzahl von Sinusschwingungen unterschiedlichster Frequenzen zusammensetzen. Demzufolge besteht auch das resultierende Frequenzspektrum aus einer großen Anzahl von Linien.

Abhängig von der gewählten Auflösung kommt es zu unterschiedlichen Darstellungen dieser Linien im Spektrum.

Speicherung im Puffer

Die abgetasteten Werte werden im internen Puffer des Moduls für 300 ms gespeichert und müssen innerhalb dieser Zeit übertragen werden. (Siehe ["Kennwertübertragung via Flatstream" auf Seite 68](#)).



Der Puffer hat eine konstante Größe und kann 8192 Messwerte aufnehmen. Damit ergibt sich folgender Zusammenhang zwischen Abtastrate und Messzeit:

Messzeit = Puffergröße / Abtastungen pro Sekunde

Da die Anzahl der gespeicherten Werte von der konfigurierten und nicht von der hardwaremäßigen Abtastrate des Moduls abhängt, werden nicht alle gemessenen Werte gespeichert. Bei einer Messzeit von 318 ms ist dies jeder zweite und bei einer Messzeit von 15,9 s jeder hundertste Wert.

Indirekt einstellbar ist die Anzahl der gespeicherten Werte durch die Register ["MaxFrequencyEnvelope" auf Seite 57](#) und ["MaxFrequencyRaw" auf Seite 58](#)

5.2.2.2.3 Aliasing-Effekt

Das Eingangssignal wird in regelmäßigen Zeitabständen abgetastet. Wird eine zu niedrige Abtastrate verwendet, so wird das Eingangssignal falsch abgetastet und ein fehlerhaftes Schwingungsabbild entsteht. Dieses unerwünschte Phänomen wird Aliasing-Effekt genannt.

Um solche falschen Ergebnisse zu vermeiden muss bei der Abtastung das sogenannte Nyquist-Abtasttheorem erfüllt sein. Dieses Abtasttheorem beschreibt das nötige Frequenzverhältnis zwischen Abtastung und Signal und besagt, dass die Abtastrate mehr als das Doppelte der maximalen Frequenz des gemessenen Signals betragen muss.

Beispiel falscher Abtastung

Sinus mit 4 kHz, abgetastet mit 6 kHz. In Rot sieht man die durch die zu niedrige Abtastung gemessene Sinuskurve mit 2 kHz.

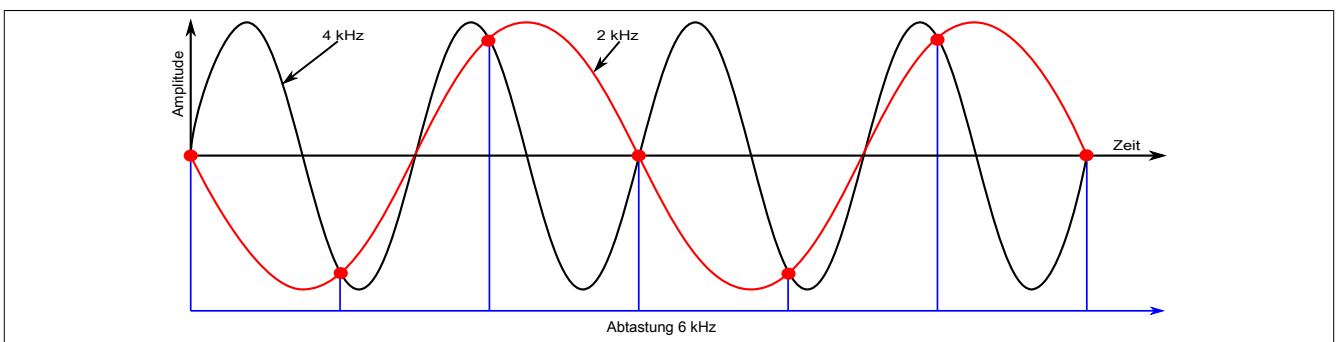


Abbildung 41: falsche Abtastung mit 6 kHz und 4 kHz

Information:

Im Modul ist sichergestellt, dass das Nyquist-Abtasttheorem immer erfüllt wird.

Bei einem Nutzsignal von 10 kHz wird eine reduzierte Abtastfrequenz von 25,7812 kHz verwendet!

5.2.2.3.4 Auflösung

Das zeitkontinuierliche analoge Eingangssignal wird in eine Folge von diskreten digitalen Ausgangswerten umgewandelt. Die A/D-Wanderauflösung bestimmt dabei die maximal mögliche Anzahl von digitalen Werten. Da diese in jedem Fall endlich ist, entsteht durch die Digitalisierung eine Abweichung vom eigentlichen Eingangssignal, die als Quantisierungsfehler bezeichnet wird.

Die Abtastrate legt den zeitlichen Abstand zwischen 2 Wandlungen fest und muss abhängig von der weiteren Verarbeitung des aufgenommenen Signals geeignet gewählt werden. So ist z. B. in vielen Signalverarbeitungsaufgaben die Einhaltung des Abtasttheorems von grundlegender Bedeutung. Das Modul stellt durch geeignete Filter- und Dezimierungsstufen sicher, dass das Abtasttheorem unabhängig von der konfigurierten Abtastrate zu jedem Zeitpunkt erfüllt ist.

Im Zeitbereich ermöglicht eine hohe Abtastrate eine detaillierte Darstellung des Signals, erhöht allerdings den dafür benötigten Speicherbedarf. Im Frequenzbereich steigt durch die Erhöhung der Abtastrate die maximal auswertbare Frequenz. Da für die FFT eine definierte Anzahl von Werten verwendet wird, sinkt damit jedoch die Auflösung und somit die Möglichkeit, nahe aneinander liegende Frequenzbereiche zu unterscheiden.

Information:

Beim X20CM4810 ergibt sich bei einer maxFrequency von 200 Hz ein Abstand zwischen 2 Linien von 0,0629 Hz.

Zeitsignal

Im Zeitsignal ist die Auflösung anhand des Abstandes zwischen 2 benachbarten Messpunkten zu erkennen.

Beispiel

In diesem Beispiel wurde ein **MaxFrequencyRaw** von 2 kHz verwendet.

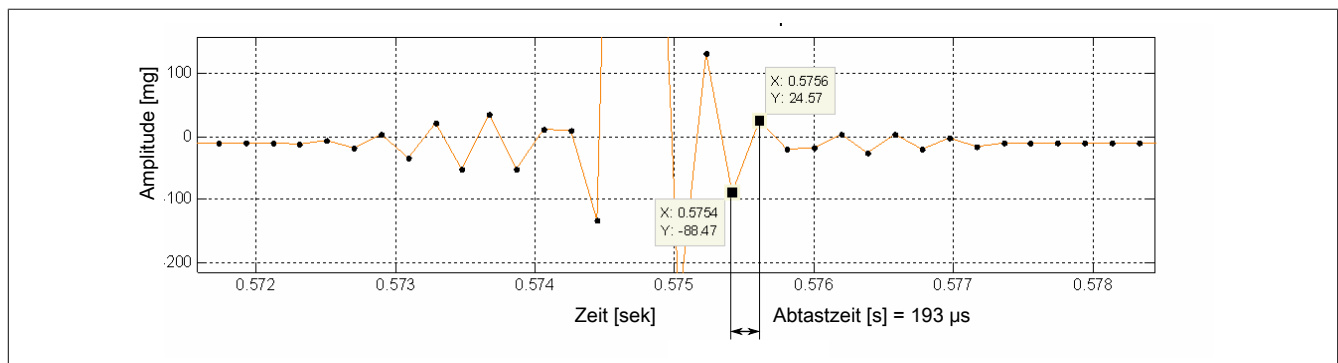


Abbildung 42: Zeitsignal mit entsprechender Auflösung

Frequenzspektrum

Die Auflösung im Frequenzspektrum gibt den Abstand zwischen den einzelnen noch auswertbaren Frequenzlinien (Spektrallinien) an.

Beispiel

Frequenzspektrum bei einer MaxFrequency von 2 kHz.

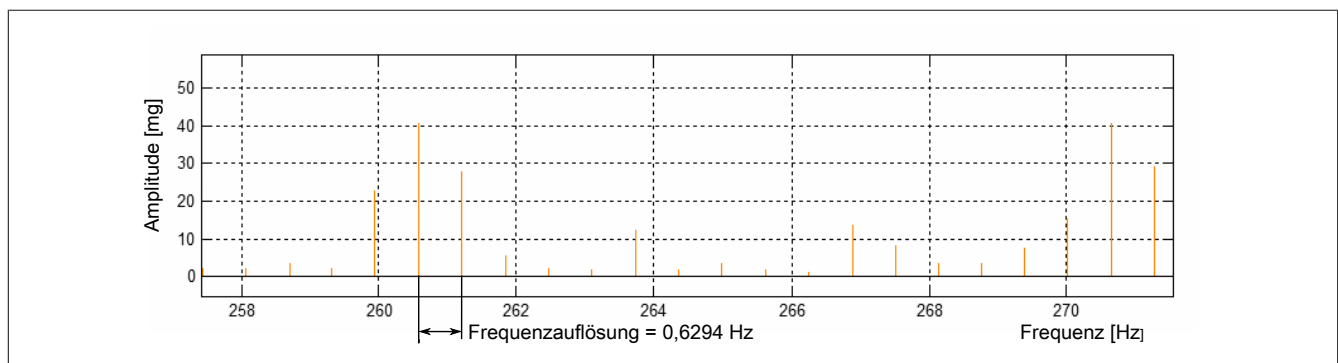


Abbildung 43: Frequenzspektrum

Quantisierung

Um analoge Signale digital verarbeiten zu können, ist eine Analog-Digital-Wandlung (A/D-Wandlung) notwendig. Durch einen A/D-Wandler können allerdings nur Spannungen in Stufen gemessen werden. Dies sind die sogenannten Spannungs-Quanten. Der zu messende Bereich wird also quantisiert.

Bei der Analog-Digital-Wandlung beschreibt die digitale Auflösung die Anzahl der Pegelstufen bei der Quantisierung. Damit wird die Genauigkeit und Empfindlichkeit bestimmt, mit der ein vorher analoger Pegelwert abgebildet wird. Je mehr Stufen vorhanden sind umso genauer ist das erhaltene diskrete Signal, bzw. umso höher die Empfindlichkeit der Messung.

Die Auflösung gibt an, in wie viele unterschiedliche digitale Werte ein analoges Signal umgewandelt werden kann. Die Auflösung wird in Bit ausgedrückt.

8-Bit Auflösung = 256 Pegelwerte
 16-Bit Auflösung = 65.536 Pegelwerte
 24-Bit Auflösung = 16.777.216 Pegelwerte

Beispiel

Eine 24-Bit Auflösung bei einem Sensormessbereich von ± 10 V und einer Sensorempfindlichkeit von 100 mV/g ergibt:

$20 \text{ V} / 2^{24} = 1,192 \text{ } \mu\text{V} \rightarrow \text{das entspricht } 11,92 \text{ } \mu\text{g}$

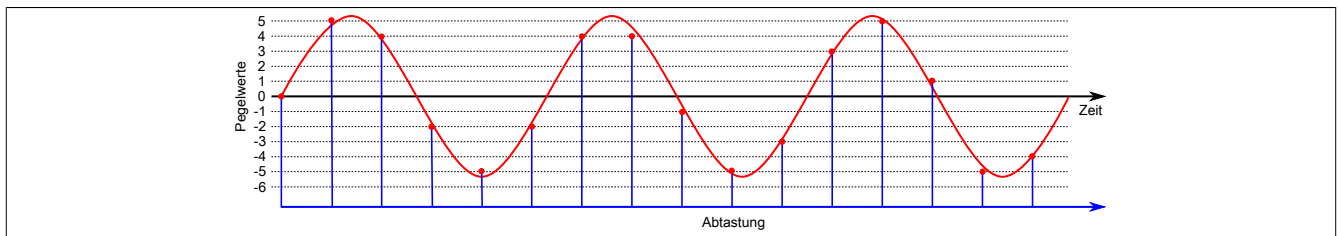


Abbildung 44: Quantisierungsfehler bei niedriger Auflösung

Information:

Das Modul hat eine Auflösung von 24 Bit.

5.2.2.2.3.5 Messzeit

Die Messzeit ist abhängig von der konfigurierten maximalen Frequenz. Je nachdem, ob die Hüllkurve oder die Rohwerte gemessen werden sollen, werden folgende Register für die Konfiguration verwendet:

- "MaxFrequencyEnvelope" auf Seite 57 für die Hüllkurvenmessung
- "MaxFrequencyRaw" auf Seite 58 für die Rohwertmessung

Maximale Frequenz	Abtastfrequenz	Messzeit
10000 Hz	25781 Hz	0,3178 s
5000 Hz	12891 Hz	0,6355 s
2000 Hz	5156 Hz	1,5888 s
1000 Hz	2578 Hz	3,1775 s
500 Hz	1289 Hz	6,3550 s
200 Hz	516 Hz	15,8875 s

5.2.2.3 Hüllkurve - Envelope

Bei Betrachtung unterschiedlicher Schwingungsursachen stellt man fest, dass es dafür hauptsächlich 2 Gründe gibt:

1. Unwucht und Ausrichtung

Unwucht und Ausrichtungsschäden führen zu überwiegend harmonischen, sinusförmigen Schwingungen.

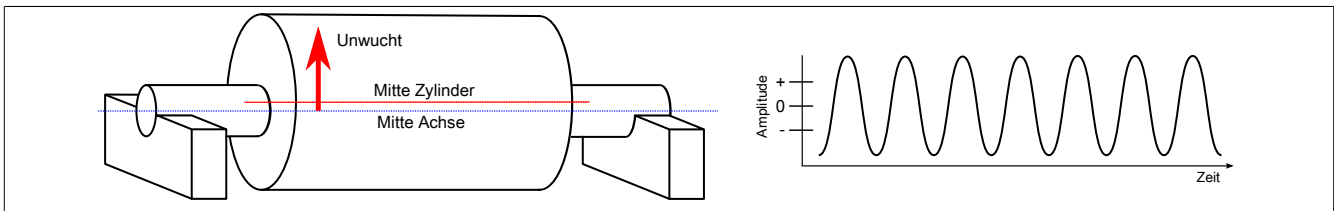


Abbildung 45: Unwucht - harmonische Schwingung

2. Stöße

Viele Maschinenschäden führen zu Stößen, welche die Maschinenstruktur oder auch angrenzende Maschinenteile zu Schwingungen mit deren Eigenfrequenz anregen. Stoßartige Anregungen werden durch Aushöhlungen oder durch einen am Maschinengehäuse anstreifenden Rotor oder durch beginnende Wälzlager- oder Verzahnungsschäden an Getrieben verursacht.

Bei Wälzlagerschäden treten diese Stöße auf, wenn entweder die Wälzkörper einen Schaden auf der Innen- oder Außenringlaufbahn überrollen oder aber einer der Wälzkörper selbst beschädigt ist.

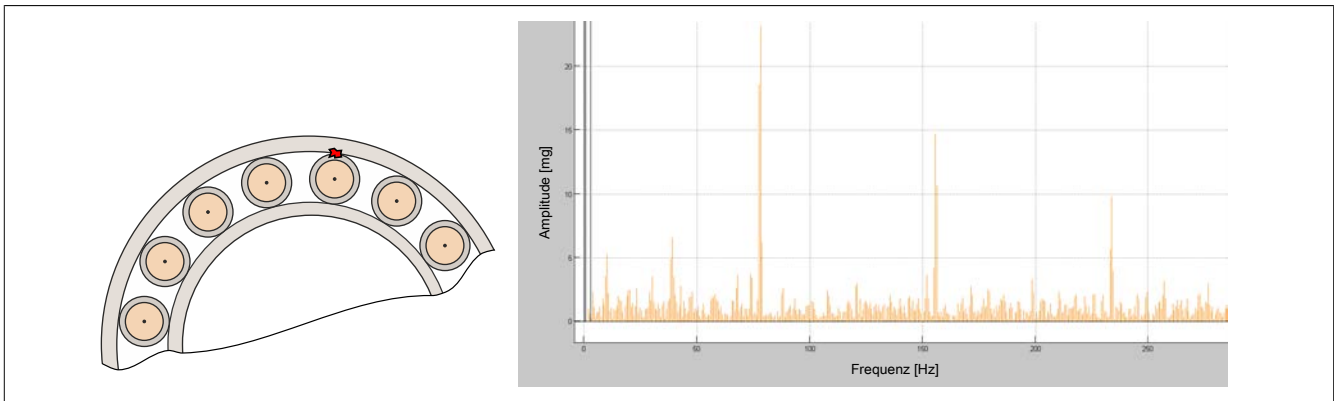


Abbildung 46: Lagerschaden – FFT des Hüllkurvensignals

Man kann eine solche Stoßanregung mit dem Anschlagen eines Klöppels an eine Glocke vergleichen. Wird die Glocke 2x pro Sekunde angeschlagen, so schwingt sie nicht mit 2 Hz, sondern mit ihrer Eigenfrequenz. Die Eigenfrequenz wird durch die konstruktive Gestaltung und den Materialeigenschaften bestimmt. Wie bei jedem Klangkörper spielt dabei auch die Befestigung eine Rolle.

Liegt ein Wälzlagerlaufbahnschaden vor, führt jeder Stoßimpuls zu einer entsprechenden Reaktion des schwingfähigen Systems. Praktisch messbar ist nur die Summe aller Impulse, also das Gesamtsignal.

Bei Wälzlagerschäden werden in erster Linie die Lagerringe zum Schwingen angeregt.

Analyse der Schwingungen

Die Schwingungen werden als Welle in der Maschine fortgeleitet und sind an der Maschinenoberfläche messbar. Voraussetzung ist natürlich, dass ein Weg für den Schalltransport bereitsteht, d. h., dass zwischen Wälzlager und Sensor keine schalldämpfenden Grenzübergänge wie Luft, Gummi etc. existieren.

An der Maschinenoberfläche kann mit Hilfe eines Beschleunigungssensors dieses Signal gemessen werden. Das Signal besteht aus einer Vielzahl von Schwingungen und Stößen, die sich überlagern. Betrachtet man das Zeitsignal so ist leicht ersichtlich, dass eine Zuordnung zu einzelnen Frequenzen, wenn überhaupt, nur schwer zu bewerkstelligen ist.

Wendet man an dieses Zeitsignal eine Fast-Fourier-Transformation an, findet man die Eigenfrequenzen des Systems. Diese liegen im höherfrequenten Bereich. Deutlich feststellbar ist die Rotorfrequenz bzw. deren Harmonische als dominante Frequenzanteile.

Die Überrollfrequenzen der Wälzlager liegen bei einer Drehzahl von etwa 600 rpm je nach Lager zwischen 15 und 70 Hz.

Insbesondere lassen sich Stoßimpulse niedriger Intensität, wie sie häufig bei beginnenden Schädigungen auftreten, kaum erkennen und bewerten. Erst bei fortgeschrittenen Lagerschaden sind deutlich erkennbare Signalspitzen vorhanden.

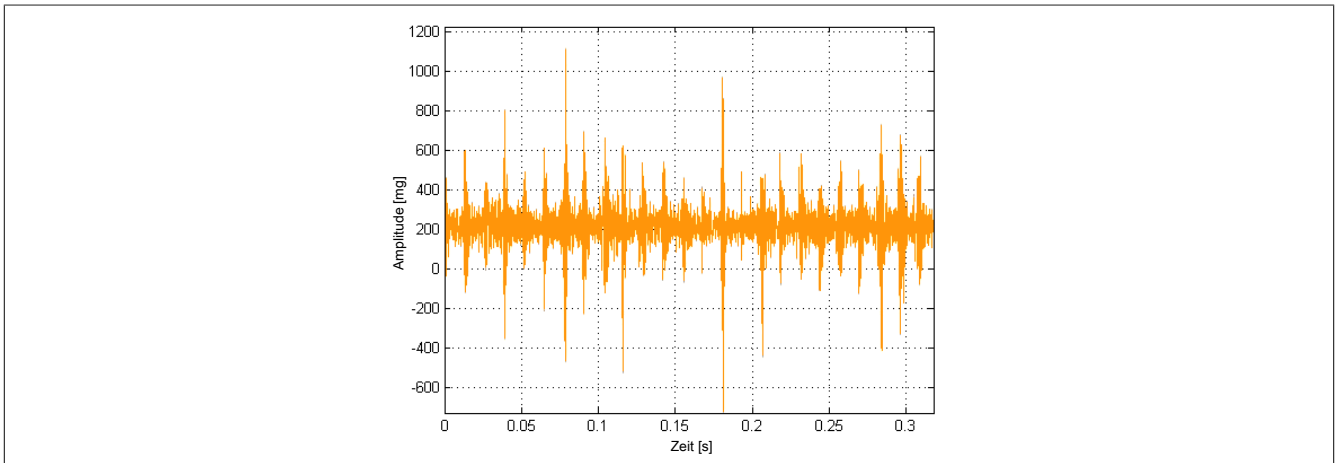


Abbildung 47: Fortgeschrittener Lagerschaden bei 600 rpm

Um das eigentliche Ereignis, nämlich die Stoßfolge deutlich sichtbar zu machen, genügt es offenbar nicht, einfach nur das Amplitudenspektrum zu bilden. Vielmehr muss der bei der Signalentstehung erfolgte Prozess der Faltung in geeigneter Weise umgekehrt werden, um die Erregerfunktion von den Eigenfrequenzen wieder zu trennen. Dazu bietet sich die Hüllkurvenanalyse an.

Ein amplitudenmoduliertes Signal besteht aus einem hochfrequenten Trägersignal und einem niederfrequenten Nutzsignal. Dabei ändert sich die Amplitude des Trägersignals in Abhängigkeit vom Nutzsignal. Beim Empfänger wird durch die Hüllkurvenbildung das Nutzsignal wieder vom Trägersignal getrennt (Demodulation).

Im Falle der durch periodische Stöße angeregten Maschinenresonanzen, können die Maschinenresonanzen als Trägersignal und die tiefpassgefilterten Stoßimpulsfolgen als niederfrequentes Modulationssignal angesehen werden. Durch die Demodulation erfolgt eine Trennung der Stoßimpulse von den Resonanzfrequenzen.

5.2.2.3.1 Hüllkurvenanalyse

Bei der Hüllkurvenanalyse wird in der Regel das Frequenzspektrum des Hüllkurvensignals untersucht. Wird der Gleichanteil unterdrückt, so erhält man ein Frequenzspektrum, das nur eine erhöhte Amplitude bei der Frequenz der niederfrequenten Sinusschwingung zeigt.

Das Hüllkurvensignal der periodisch stoßförmig angeregten Maschinenresonanz zeigt meist erhöhte Amplituden bei der Stoßimpulsfolgefrequenz und deren Vielfachen.

Die Hüllkurvenanalyse ist also eine Methode um harmonische Anregungen (Unwucht, Ausrichtung) von Stoßanregungen (Wälzlagerschäden, Verzahnungsschäden etc.) zu unterscheiden.

Umgekehrt muss festgehalten werden, dass harmonische Anregungen in einem Hüllkurvenspektrum nicht entsprechend festgestellt werden können.

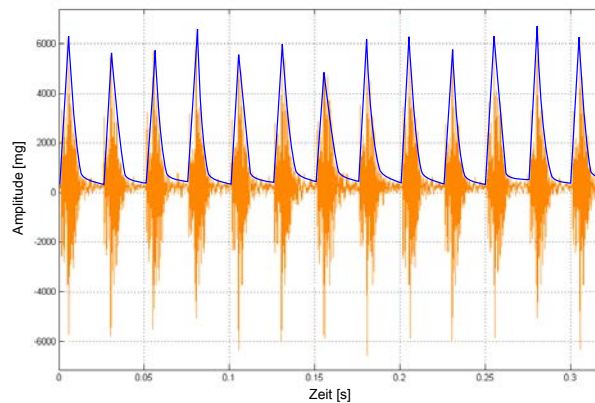


Abbildung 48: Zeitsignal mit Hüllkurve

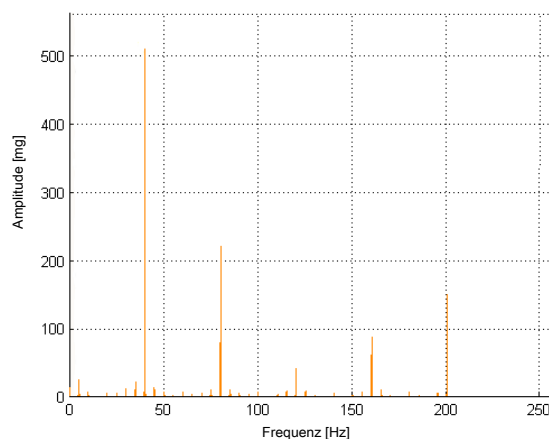


Abbildung 49: Resultierendes Frequenzspektrum der Hüllkurve

5.2.2.4 Weg - Geschwindigkeit - Beschleunigung

Sensoren können Schwingbeschleunigung, Schwinggeschwindigkeit oder Schwingweg erfassen. Unabhängig davon, welche physikalische Größe der Aufnehmer erfasst, ist die Darstellung der Schwingung als Kombination von Beschleunigung, Geschwindigkeit oder Weg möglich, denn es gilt:

$$s = \int v dt = \iint a dt^2, \text{ bzw. } a = \frac{dv}{dt} = \frac{d^2s}{dt^2}$$

Legende:

- s- Schwingweg
- v- Schwinggeschwindigkeit
- a- Schwingbeschleunigung
- t- Zeit

Da sich die Schwinggeschwindigkeit durch Integration aus der Schwingbeschleunigung und der Schwingweg aus der Integration der Schwinggeschwindigkeit berechnen lässt, ist eine eventuell notwendige Umrechnung des Sensorwertes jederzeit möglich.

Die Beschleunigung ist der Kraft proportional. Dem gegenüber lässt sich aus der Geschwindigkeit auf die Energie schließen.

Praktische Bedeutung hat ebenfalls, dass in die Umrechnung von einer physikalischen Größe in die Andere auch die Frequenz enthalten ist. Für eine Sinusschwingung gilt:

$$s = \frac{a}{(2 \cdot \pi \cdot f)^2} \quad v = \frac{a}{2 \cdot \pi \cdot f}$$

Die Frequenz steht im Nenner. Das hat zur Folge, dass hochfrequente Signalanteile bei der Verwendung der Schwinggeschwindigkeit gegenüber der Schwingbeschleunigung unterbewertet werden. Bei der Verwendung des Schwingweges quadriert sich dieser Einfluss.

5.2.3 Grenzwertfindung und Alarmgrenzen

Eine Basisaussage über den Maschinenzustand lässt sich durch den Vergleich der gemessenen Kennwerte mit, durch Normen (z. B. Iso10816) und vom Maschinenhersteller vorgegebenen Grenzwerten treffen.

Aus den gemessenen Signalen werden Kennwerte berechnet, die repräsentativ für den Zustand der Anlage am jeweiligen Messpunkt sind.

Für die Festlegung der Alarmgrenzen können teilweise Normen herangezogen werden. Darüber hinaus ergeben sich Grenzen durch die Historie der Maschine und die Erfahrung des Betreibers.

5.2.3.1 Vergleich mit Referenzen und Normen

Für bestimmte Maschinen und Anlagen sind Grenzwerte in Normen verankert. Abgesehen von ISO10816 geben diese jedoch nur wenige Informationen zur Beurteilung des tatsächlichen Maschinenzustandes.

5.2.3.2 Grenzwerte des Herstellers

Einige wenige Maschinenhersteller geben Maximalwerte für zulässige Schwingungen und andere relevante Größen zur Beurteilung des Maschinenzustandes an. Diese basieren teilweise auf Berechnungen, aber auch auf dem Wissen und der Erfahrung des Herstellers.

Diese vorgegebenen Grenzwerte sind bevorzugt für eine Beurteilung heranzuziehen.

5.2.3.3 Grenzwerte des Betreibers - Erfahrungswerte

Für die Beurteilung des Zustandes der Maschine können auch die Erfahrung durch den Betreiber selbst herangezogen werden. Aus der Langzeitbeobachtung der Kennwerte und der Historie der Maschine lassen sich entsprechende Erfahrungswerte ableiten.

Die sich daraus ergebenden Grenzwerte können deutlich von Grenzwerten aus Normen und den Vorgaben des Maschinenherstellers abweichen. Diese Beurteilung ist jedoch nur möglich wenn bereits viel Erfahrung über die Schwingungen in der Maschine bzw. ein Gut- / Schlechtvergleich vorhanden ist.

Für die Grenzwertfindung ist zu beachten, dass die gemessenen Vibrationen unter anderem von folgenden Faktoren abhängig sind:

- der Position des Schadens
- der Position des Sensors
- der Geschwindigkeit der bewegten Maschinenteile
- der Belastung der Maschinenteile

5.2.3.4 Beurteilung des Trends

Vielfach ist das tatsächliche Verhalten einer Maschine im Betriebs- und vor allem im Schadensfall nicht ausreichend bekannt.

Um eine zuverlässige Aussage über den Zustand treffen zu können wird der zeitliche Verlauf der Kennwerte (Kennwerttrend) zusätzlich zur Zustandsbeurteilung herangezogen.

Im Trendverlauf wird vom "Normalzustand" ausgegangen. Der Referenzpegel (Normalpegel) ist die Höhe des Kennwertes, der sich beim Normalzustand einstellt.

Ausgehend vom Normalzustand wird der Trend auf Veränderungen beobachtet. Im Schadensfall steigen die jeweils relevanten Kennwerte zumeist an, aber auch ein Abfall kann ein Hinweis auf eine Problem in der Anlage bedeuten.

Für eine sichere Beurteilung über Trends ist es unerlässlich, dass die Schwingungskennwerte immer bei gleichen Betriebsbedingungen aufgenommen und eventuell Klassen zugeordnet werden. Insbesondere die Drehzahl und gegebenenfalls Lastverhältnisse haben einen starken Einfluss auf die Kennwerte. Erhöhungen im Trend können bei ungleichen Betriebsbedingungen nicht eindeutig einer Veränderung des eigentlichen Zustandes zugeordnet werden. Mit Hilfe der Trendbeobachtung lässt sich in vielen Fällen der Maschinen- und Lagerzustand gut überwachen.

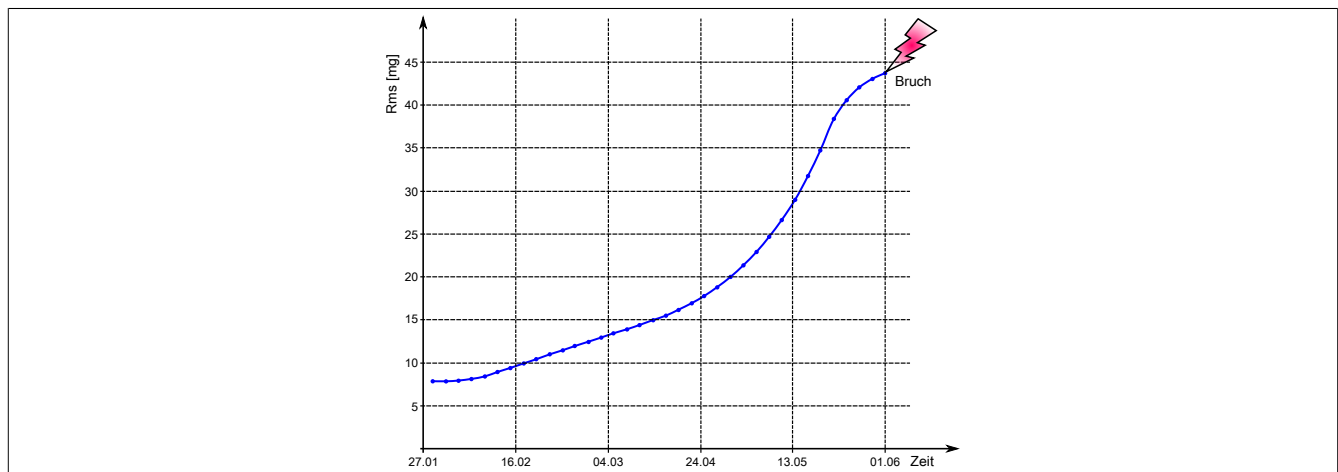


Abbildung 50: Typischer Trendverlauf

Die erste bzw. zweite Ableitung der Trendlinie des Schadensverlaufs kann einen guten Rückschluss auf den Schadenszustand eines Bauteiles liefern.

Beispiel

Benutzung der ersten und zweiten Ableitung für die Bestimmung des besten Reparaturzeitpunktes. Der Zeitpunkt des Austausches wird in Bezug auf die maximale Lebensdauer gewählt. Ein eventuelles vorhergehendes Absinken der Produktionsqualität bleibt unberücksichtigt.

Anhand des Trendverlaufs können verschiedene Austauschzeitpunkte gewählt werden

- k1 = Erster Anstieg. Sehr Konservativ, Lebensdauer wird verschenkt
- k2 = Zweiter Anstieg. Guter Kompromiss zwischen konservativ und optimal.
- k3 = Absinken der zweiten Ableitung. Optimale Ausnutzung der Lebensdauer

Information:

Dieses Beispiel der Nutzung einer Ableitung darf nicht als Regel in eigene Anwendungen übernommen werden.

Erste Ableitung

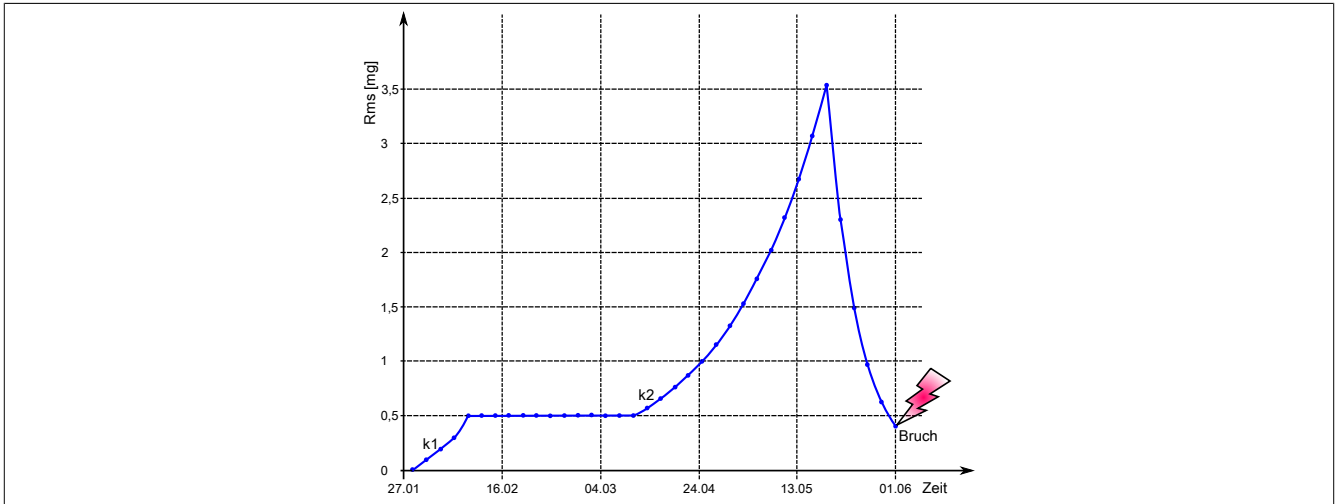


Abbildung 51: Erste Ableitung des Trendverlaufs

Zweite Ableitung

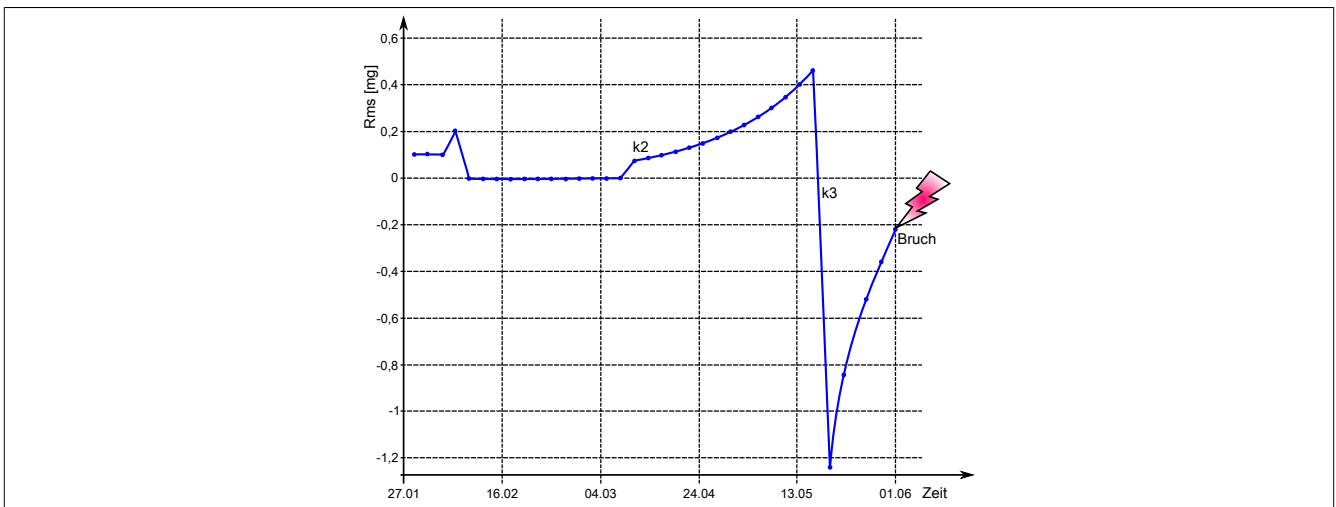


Abbildung 52: Zweite Ableitung des Trendverlaufs

5.2.3.5 Dynamische Drehzahländerung

Bei der FFT-Berechnung im Modul wird davon ausgegangen, dass sich das Frequenzspektrum, und damit die daraus resultierenden Linien, während der Berechnungsdauer für einen Datenpuffer nicht verändert. Dies ist bei Wellen aber nur bei einer konstanten Drehzahl der Fall.

Bei dynamischen Drehzahlen können keine schmalbandigen Frequenzbänder zur Überwachung oder Analyse von Schadfrequenzen verwendet werden, da diese keine gültigen Resultate liefern.

Das folgende Beispiel zeigt, welche Auswirkung die Drehzahländerung einer Welle auf das resultierende Frequenzspektrum während der FFT-Berechnung hat.

Beispiel

Die Drehzahl einer Welle wird von 100 Hz auf 200 Hz geändert.

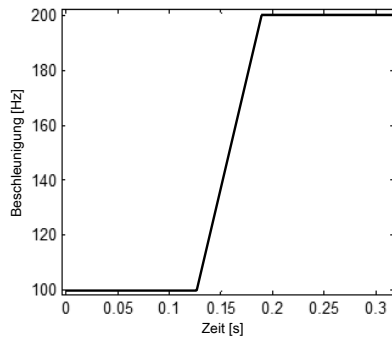


Abbildung 53: Drehzahlprofil

Veränderung der Drehzahl einer Welle
(innerhalb einer Puffermesslänge. Mit der
Drehzahlfrequenz wird ein Schaden simu-
liert)

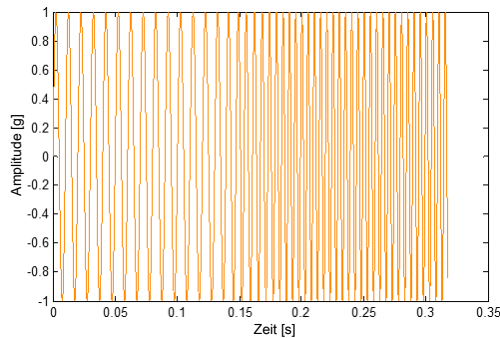


Abbildung 54: Zeitsignal

Veränderung im Zeitsignal

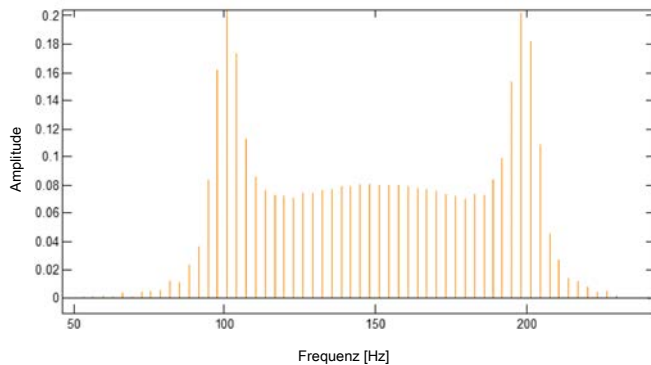


Abbildung 55: Ungültiges Ergebnis im Frequenzspektrum

Resultierendes Spektrum,
berechnet aus dem Zeitsignal

Mögliche Ansätze zum Messen in einem dynamischen System

- **Beste Lösung:** Einführen einer eigenen Messung für die Trendanalyse, bei der die Drehzahl für die Dauer der Messung konstant gehalten wird.
- Falls eine Messung mit konstanter Geschwindigkeit nicht möglich ist, kann eine Auswertung des Maschinenzustandes über die Kennwerte oder ein breitbandiges Frequenzband erfolgen. (In dem Beispiel wären dies 80 bis 220 Hz)

5.3 Praktische Anwendung Schadenserkennung

Der Zustand einer Maschine bzw. Anlage kann durch das Messen von fehlerrelevanten Parametern gut erfasst werden. Aus diesen Parameterdaten werden über verschiedene Algorithmen im Modul Kennwerte berechnet. Diese Kennwertbildung erfolgt kontinuierlich und automatisch. Sie erfordert nur geringes Fachwissen und ist leicht anzuwenden und zu implementieren. Die Maschine kann damit auf einfache Art und Weise auf Schädigungen und fehlerhafte Zustände überprüft werden.

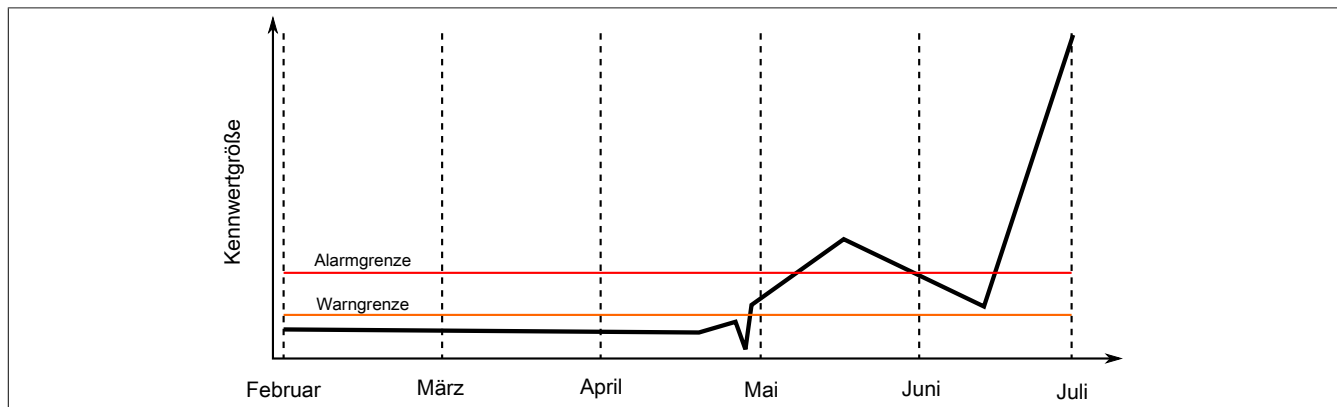
Die Auswahl geeigneter Kennwerte und deren Auswertung über einen längeren Zeitraum sind die Basis für eine effektive und erfolgreiche Maschinenüberwachung, dem sogenannten 'Trending'. Sie ermöglichen die Überwachung vieler Aggregate mit relativ wenig messtechnischem und personellem Aufwand.

Dabei können aber nicht nur einfache Grenzwertsetzungen zur Warnung herangezogen werden, sondern vielfach auch logische Korrelationen zu anderen Parametern, wie Last, Drehzahl oder eben die Form der Trendkurve.

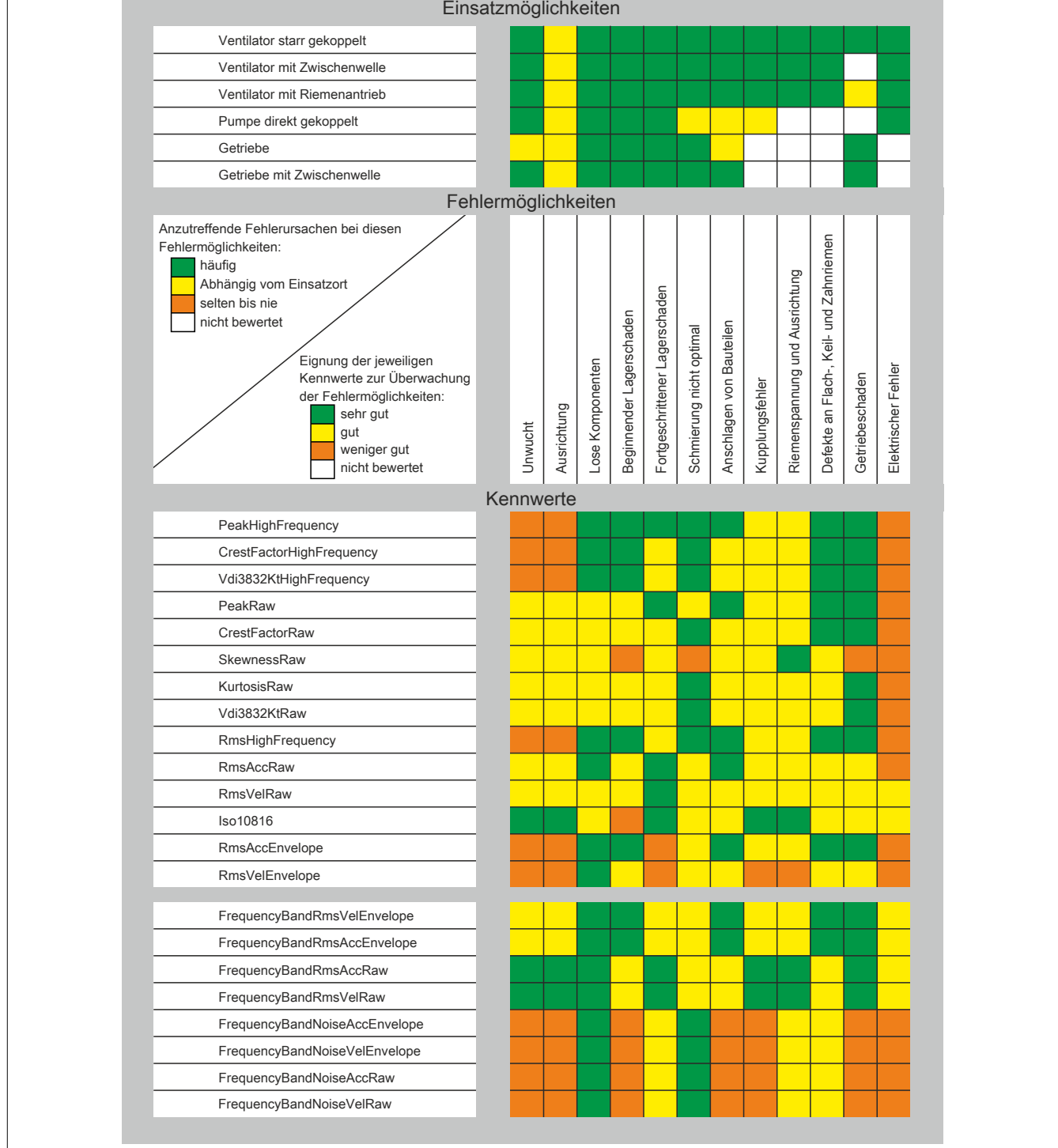
In der Trendkurve werden Kennwerte erfasst und mit Normen oder vorhandenen Erfahrungswerten über einen längeren Zeitraum verglichen. Dieser Trendverlauf kann gut für die Beurteilung des Maschinenzustandes herangezogen werden. Die Trendentwicklung dieser Kenngrößen zeigt Zustandsverschlechterungen, d. h. sich anbahnende Schädigungen an.

Aus der Aufzeichnung der Messwerte über einen längeren Zeitraum in einem Diagramm zeigt sich die Lage der Kennwerte zu voreingestellten Warn- und Alarmschwellen. Werden diese Schwellen überschritten, liefert die Schwingungsdiagnose die Fehlerursache, sodass geeignete Instandhaltungsmaßnahmen eingeleitet werden können.

Beispiel eines Kennwertverlaufs



Die Bewertung basiert auf einer Einschätzung typischer Anwendungsfälle.



5.3.2 Fehlermöglichkeiten

Die Auswirkung eines Schadens auf das Schwingungsverhalten ist abhängig von der Art und dem Ausmaß des Schadens und somit bei jedem Auftreten einzeln zu beurteilen.

Einsatzmöglichkeiten									
	Elektromotor								
	Ventilator								
	Kupplung								
	Flach- und Keilriemen								
	Zahnriemen								
	Pumpe								
	Getriebe								

Fehlermöglichkeiten												
Anzutreffende Fehlerursachen bei diesen Einsatzmöglichkeiten: häufig applikationsabhängig selten bis nie nicht bewertet Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet	Unwucht	Ausrichtung	Lose Komponenten	Begleitender Lagerschaden	Fortgeschrittener Lagerschaden	Schmierung nicht optimal	Anschlagen von Bauteilen	Kupplungsfehler	Riemenspannung und Ausrichtung	Defekte an Flach-, Keil- und Zahnriemen	Getriebeschaden	Elektrischer Fehler

Kennwerte (KW)											
Vibrationen	KW aus dem Rohsignal (Zeitverlauf) der Schwingbeschleunigung	PeakHighFrequency									
		CrestFactorHighFrequency									
		Vdi3832KtHighFrequency									
		PeakRaw									
		CrestFactorRaw									
		SkewnessRaw									
		KurtosisRaw									
		Vdi3832KtRaw									
	KW aus dem Effektivwert des Rohsignals (Rawsignal) der Schwingbeschleunigung	RmsHighFrequency									
		RmsAccRaw									
	KW aus dem Effektivwert des Rohsignals (Rawsignal) der Schwinggeschwindigkeit	RmsVelRaw									
		Iso10816									
	KW aus dem Effektivwert des Rohsignals (Rawsignal) der Schwinggeschwindigkeit	RmsAccEnvelope									
		RmsVelEnvelope									
	KW aus einem selektiven Effektivwert	FrequencyBandRmsVelEnvelope									
		FrequencyBandRmsAccEnvelope									
		FrequencyBandRmsAccRaw									
		FrequencyBandRmsVelRaw									
		FrequencyBandNoiseAccEnvelope									
		FrequencyBandNoiseVelEnvelope									
		FrequencyBandNoiseAccRaw									
		FrequencyBandNoiseVelRaw									

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.2.1 Unwucht

Von einer Unwucht spricht man bei rotierenden Körpern, deren Masse nicht rotationssymmetrisch verteilt ist. Der Schwerpunkt liegt nicht auf der Rotationsachse!

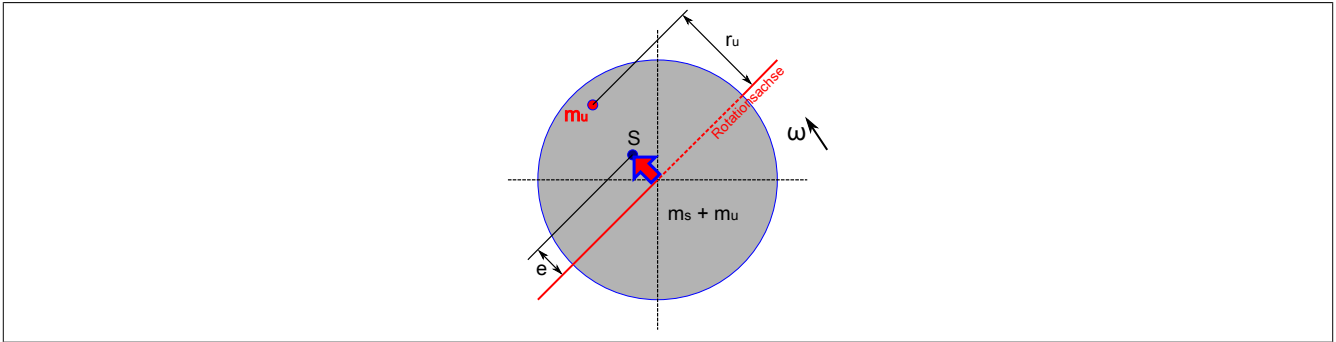


Abbildung 56: Symbolische Skizze einer Unwucht

Unwucht führen vor allem bei hohen Drehzahlen zu Vibrationen und erhöhtem Verschleiß, weshalb sie normalerweise durch Anbringen von Gegengewichten ausgeglichen, d. h. ausgewuchtet wird. Ein vollständiges Ausgleichen gelingt in der Praxis nur sehr selten, sodass der rotierende Körper mit einer Restunwucht behaftet bleibt.

Die durch die Unwucht verursachte Fliehkraft ist vom Quadrat der Drehzahl abhängig und wirkt sich daher bei höheren Drehzahlen (höhere Drehzahlfrequenz) wesentlich stärker aus. In einem Spektrum ist die Frequenzlinie der Drehzahl daher deutlich erhöht.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Unwucht	$1 \times f_n$	-	Erst eine sehr starke Unwucht führt zu Schlägen mit Drehzahl die sich im Hüllkurvenspektrum zeigen.

f_n ... Nenndrehzahl

Information:

Das Modul kann die Stärke der Unwucht, aber nicht deren Position an der Welle messen. Es kann daher nicht zum Auswuchten verwendet werden.

Fehlerursachen			
Abrasive Medien	häufig	häufig	häufig
Staubanpackungen	häufig	häufig	häufig
Fremdkörper	häufig	häufig	häufig
Überlastung	häufig	häufig	häufig
Mögliche Auswirkungen			
Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten: häufig Abhängig vom Einsatzort selten bis nie nicht bewertet Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet			
	Unruhiger Lauf	Erhöhter Verschleiß	Bruch von Komponenten
Kennwerte			
PeakHighFrequency	gut	gut	gut
CrestFactorHighFrequency	gut	gut	gut
Vdi3832KtHighFrequency	gut	gut	gut
PeakRaw	sehr gut	gut	gut
CrestFactorRaw	sehr gut	gut	gut
SkewnessRaw	sehr gut	gut	gut
KurtosisRaw	sehr gut	gut	gut
Vdi3832KtRaw	sehr gut	gut	gut
RmsHighFrequency	gut	gut	gut
RmsAccRaw	gut	gut	gut
RmsVelRaw	sehr gut	sehr gut	sehr gut
Iso10816	sehr gut	sehr gut	sehr gut
RmsAccEnvelope	gut	gut	gut
RmsVelEnvelope	gut	gut	gut
FrequencyBandRmsVelEnvelope	gut	gut	gut
FrequencyBandRmsAccEnvelope	gut	gut	gut
FrequencyBandRmsAccRaw	sehr gut	sehr gut	sehr gut
FrequencyBandRmsVelRaw	sehr gut	sehr gut	sehr gut
FrequencyBandNoiseAccEnvelope	gut	gut	gut
FrequencyBandNoiseVelEnvelope	gut	gut	gut
FrequencyBandNoiseAccRaw	gut	gut	gut
FrequencyBandNoiseVelRaw	gut	gut	gut

Abbildung 57: Fehlerursachen und Symptome für eine Unwucht

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.2.2 Fehlausrichtung

Im Betrieb kann eine Vielzahl unterschiedlicher Einflüsse dazu führen, dass die Maschinenteile nicht oder nicht mehr miteinander fluchten.

Wellen sollten sich an den Kupplungsstellen um eine lineare Achse drehen, damit die Rückstellkräfte an der Kupplung und die Belastungskräfte in den Lagern möglichst klein gehalten werden. Eine Fehlausrichtung führt zu erhöhten Vibrationen und Verschleiß.

Üblicherweise setzt sich eine Fehlausrichtung aus einem Parallel- und einem Winkelversatz zusammen. Bei einem starken Parallelversatz sind deutlich erhöhte Werte im Bereich der doppelten Drehzahl zu beobachten.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Fehlausrichtung bei Kupplung	1 x f_n , 2 x f_n (Teilweise 3 x f_n , 4 x f_n)	-	Bei Parallelversatz tritt meist nur '1 x f_n ' auf

f_n ... Nenndrehzahl

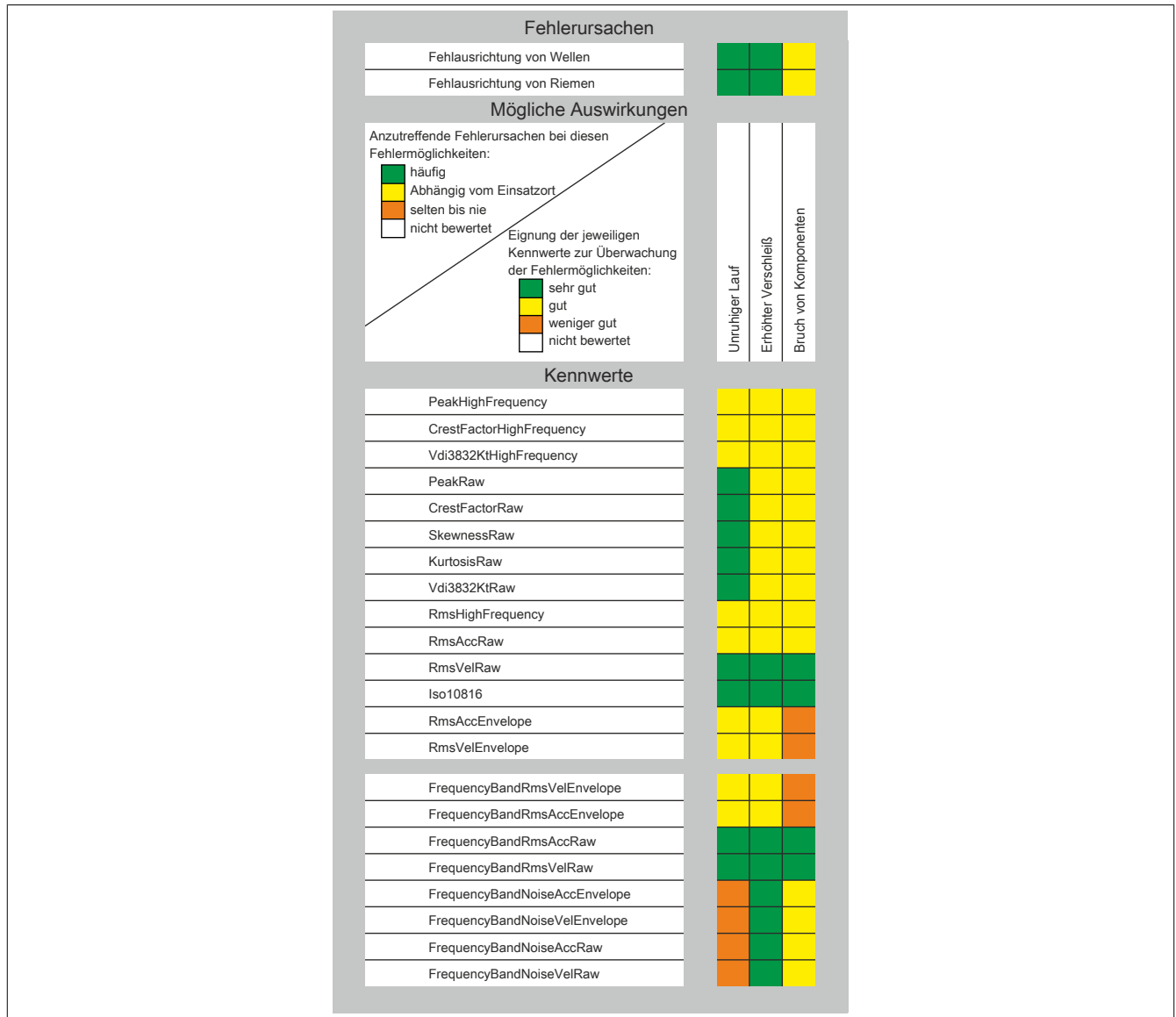


Abbildung 58: Fehlerursachen und Symptome für einen Ausrichtfehler

Für die Bedeutung der einzelnen Kennwerte siehe "[Kennwerte](#)" auf Seite 41 und "[Konfiguration](#)" auf Seite 64

5.3.2.3 Riemenschäden

Riemen führen immer wieder zu vielfältigen Problemen bei Antrieben. Zu hohe oder niedrige Riemen Spannung oder schlecht ausgerichtete Riemen führen zu Riemenschäden. Überrollt die beschädigte Riemenstelle die Riemenscheibe, kommt es zu Stoßanregungen die gemessen werden können.

5.3.2.3.1 Flach- und Keilriemen

Bei Flach- und Keilriemen wird das Drehmoment durch die Kontaktfläche zwischen dem Riemen und der Riemenscheibe übertragen. Riemenantriebe sind hinsichtlich der Ausrichtung wenig empfindlich, führen aber dennoch bei zu hoher Abweichung zu Schwingungsentwicklung und vor allem zu einem erhöhten Verschleiß und Energieverbrauch.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
schadhafter Riemen	1 x fr, 2 x fr, 3 x fr ...	1 x fn1, 1 x fn2, 1 x fr	Meist treten die Riemenfrequenzen in beiden Spektren auf

fr ... Riemendrehzahl

fn1 ... Nenndrehzahl Welle 1

fn2 ... Nenndrehzahl Welle 2

Schlecht ausgerichtet oder nicht richtig gespannte Riemen führen ebenfalls zu einer starken Belastung der Lagerstellen. Die erhöhte Vibration verursacht zusätzlich einen stärkeren Verschleiß von Riemen und Riemenscheiben.

Die Vibrationen zeigen sich deutlich an den Lagerstellen der einzelnen Wellen

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Fehlausrichtung bei Riemen	1 x fn, 2 x fn 1 x fr	1 x fn	Stoßen die Riemen seitlich an die Riemenscheibe an, so können auch Stöße mit Drehzahl und Riemenfrequenz auftreten

fn ... Nenndrehzahl

fr ... Riemendrehzahl

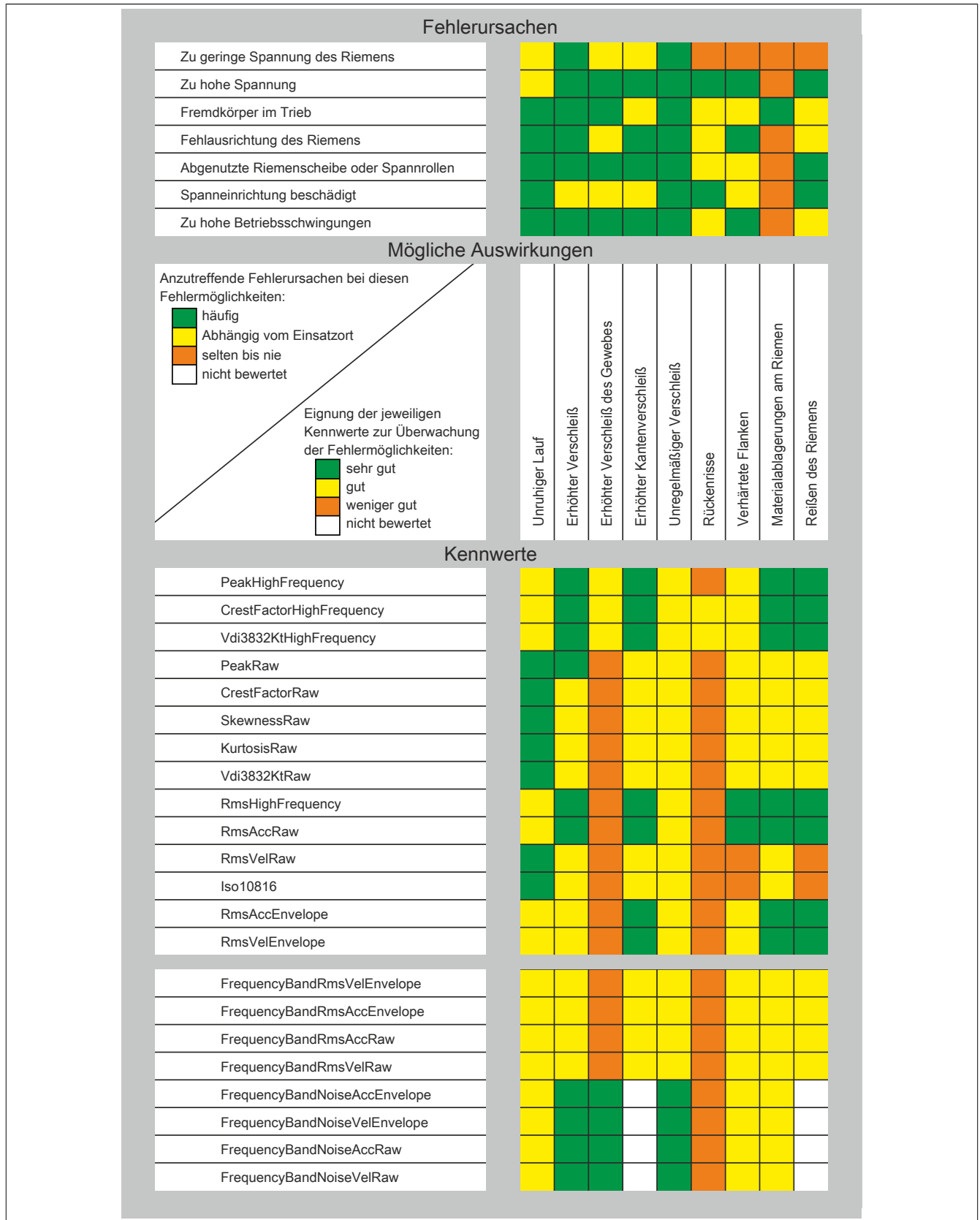


Abbildung 59: Fehlerursachen und Symptome für Riemenschäden an Flach- und Keilriemen

Für die Bedeutung der einzelnen Kennwerte siehe ["Kennwerte"](#) auf Seite 41 und ["Konfiguration"](#) auf Seite 64

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Riemenverzahnungsschaden	1 x fr	1 x fr, 1 x fzn	Die Zahneingriffsfrequenzen in Kombination mit der jeweiligen Drehzahl sind gut zu sehen

Fehlerursachen

Zu geringe Spannung des Riemens
Zu hohe Spannung
Fremdkörper im Trieb
Fehlausrichtung des Riemens
Abgenutzte Riemenscheibe oder Spannrollen
Spanneinrichtung beschädigt
Zu hohe Betriebsschwingungen

Fehlermöglichkeiten

Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten:

- häufig
- Abhängig vom Einsatzort
- selten bis nie
- nicht bewertet

Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten:

- sehr gut
- gut
- weniger gut
- nicht bewertet

Kennwerte

PeakHighFrequency
CrestFactorHighFrequency
Vdi3832KtHighFrequency
PeakRaw
CrestFactorRaw
SkewnessRaw
KurtosisRaw
Vdi3832KtRaw
RmsHighFrequency
RmsAccRaw
RmsVelRaw
Iso10816
RmsAccEnvelope
RmsVelEnvelope
FrequencyBandRmsVelEnvelope
FrequencyBandRmsAccEnvelope
FrequencyBandRmsAccRaw
FrequencyBandRmsVelRaw
FrequencyBandNoiseAccEnvelope
FrequencyBandNoiseVelEnvelope
FrequencyBandNoiseAccRaw
FrequencyBandNoiseVelRaw

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.2.4 Lose oder anschlagende Teile

Wenn einzelne Teile in der Maschine lose sind, so verursachen diese ungewollte Schwingungen. Sehr ähnlich zeigt sich auch das Anschlagen von Einzelteilen an Gehäusen oder Anbauteilen, daher können diese beiden Schwingungsverursacher nicht getrennt analysiert werden.

Jeweils einmal pro Umdrehung kommt es zum Anschlagen von Komponenten an ihre Gegenteile. Diese wiederum regen die Anbauteile zum Schwingen in deren Eigenfrequenz an. Mit der Hüllkurvenanalyse können die Stoßanregungen separiert werden.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
lockere Teile, Anschlagen von Teilen	$(1 \times f_n)$	$1 \times f_n$	Es kommt meist einmal pro Umdrehung zum Anschlagen. Tritt ein Umschlagen der Last auf, so kann auch doppelte Frequenz festgestellt werden.

f_n ... Nenndrehzahl

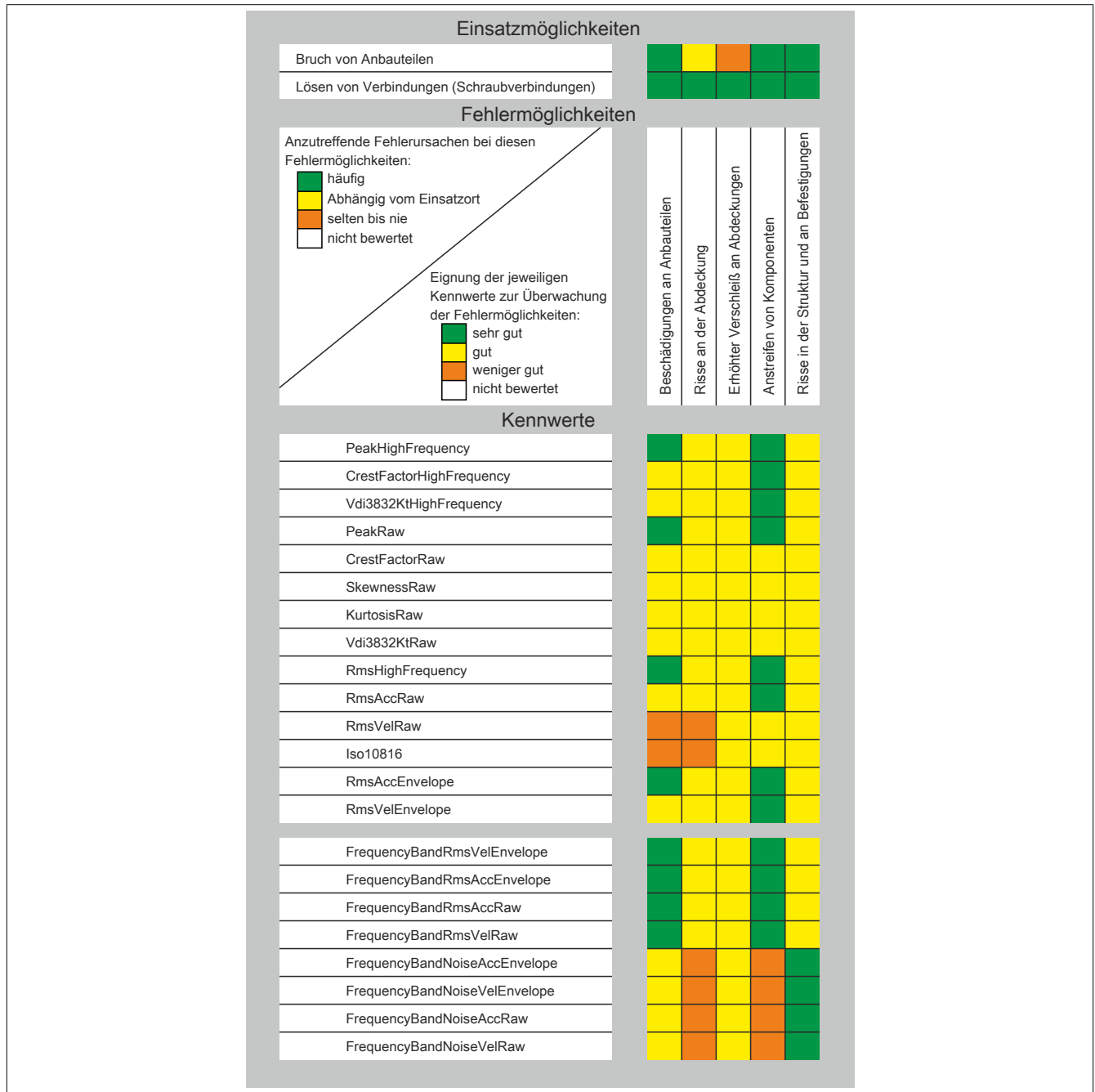


Abbildung 61: Fehlerursachen und Symptome von losen oder anschlagenden Teilen

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.2.5 Gleitlagerschäden

In einer Körperschallmessung zeigen sich die typischen Schadensfrequenzen eines Gleitlagers erst in einem sehr späten Stadium. Damit ist diese Methode weniger für eine Früherkennung geeignet.

5.3.2.6 Wälzlagerschäden

Viele Schäden von Lagern haben als Ausgangspunkt Ungenauigkeiten an den Laufflächen. Diese Pittings, d. h. Materialschäden bzw. Mikrorisse werden von den Wälzkörpern überrollt und führen zu Stoßanregungen des Wälzlagers und dessen Anbauteile.

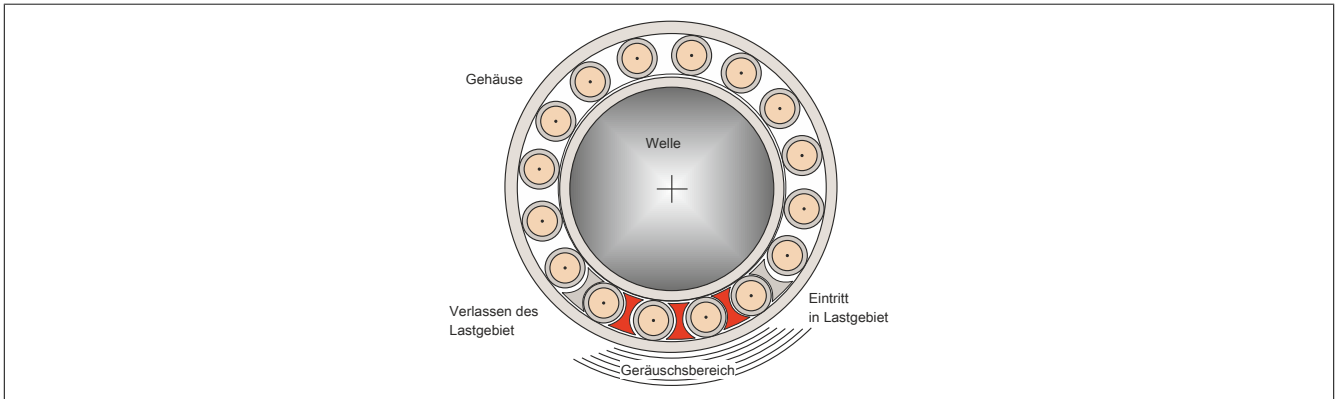


Abbildung 62: Überrollvorgänge im Lager

Der Mechanismus ist dem des Anschlagens einer Glocke sehr ähnlich: Der Klöppel schlägt auf den Glockenkörper auf und die Glocke beginnt mit ihrer Eigenfrequenz zu schwingen.

Im Lager ist jede Überrollung des Schadens durch einen Wälzkörpers wie das Anschlagen des Klöppels und die Wälzkörperteile und die Anbauteile werden dabei zum Schwingen angeregt.

Diese sehr kleinen Schwingungen können als Modulation bzw. Überlagerung der Erregerfrequenz an der Oberfläche des Lagers gemessen werden.

Entsprechende Analysemethoden wie z. B. die Bildung der Hüllkurve können die Überlagerung wieder separieren und so die Überrollfrequenzen der Lager gut sichtbar machen.

Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet								
		An allen Komponenten	Risse	Korrosion	Freß- und Gleiterscheinungen	Stromriefen	Aufschmelzungen	Schmelzkragen
Fehlerursachen								
Montagefehler								
Montagefehler			sehr gut	gut	sehr gut			
Verspannung			sehr gut	weniger gut	sehr gut			
Beschädigung bei der Montage			sehr gut	gut	sehr gut			
Schiefstellung der Lagerringe zueinander			gut		weniger gut			
Unsachgemäße Erwärmung beim Einbau			sehr gut	gut	weniger gut			
Fremdkörper (Späne, Schmutz)			gut	gut	sehr gut			
Unsachgemäße Lagerung, fehlende Konservierung			weniger gut	gut	weniger gut			
Unsachgemäßer Transport, fehlende Konservierung			weniger gut	gut	weniger gut			
Durch die Montage veränderte Lagerluft			sehr gut	weniger gut	sehr gut			
Eindringen von Wasser oder anderen Medien			weniger gut	sehr gut	weniger gut			
Schweißstrom			gut	weniger gut	weniger gut	sehr gut	sehr gut	sehr gut
Hammerschläge beim Einbau			sehr gut					
Fehler der Anbauteile								
Zu lose Passung			gut	sehr gut	sehr gut			
Zu feste Passung			sehr gut	gut	sehr gut			
Unzureichende Gehäusesteifigkeit				gut	sehr gut			
Unpassende Lagerluft			gut	sehr gut	sehr gut			
Fehler im Betrieb								
Vorangegangene Transportschwingungen			sehr gut	gut	weniger gut			
Dreh-/Torsionsschwingungen			sehr gut	gut	weniger gut			
Vorangegangene Stillstandsschwingungen			sehr gut	gut	weniger gut			
Mikroschwingungen			sehr gut	sehr gut	weniger gut			
Betriebsschwingungen			sehr gut	gut	weniger gut			
Überschreiten der zulässigen Drehzahl der Wälzlager				weniger gut	sehr gut			
Überlastung				weniger gut	sehr gut			
Unterschreiten der Mindestbelastung			gut	gut	sehr gut			
Durch den Betrieb veränderte Lagerluft			sehr gut	gut	sehr gut			
Stetiger Stromdurchgang mit geringer Stromstärke			weniger gut	gut	weniger gut	sehr gut	sehr gut	sehr gut
Überschläge (Strom)			weniger gut	gut	weniger gut	sehr gut	sehr gut	sehr gut
Temperaturschwankungen			gut	gut	weniger gut			
Eindringen von Wasser oder aggressiven Medien			weniger gut	sehr gut	weniger gut			
Stark wechselnde Dynamik-/Lastwechsel			sehr gut	weniger gut	sehr gut			
Fehler mit der Schmierung								
Schmierstoffmangel			sehr gut	sehr gut	sehr gut			
Überschmierung			sehr gut	sehr gut	sehr gut			
Alterung des Schmiermittels			sehr gut	sehr gut	sehr gut			
Ungeeignetes Schmiermittel			sehr gut	sehr gut	sehr gut			
Verunreinigung des Schmiermittels			gut	sehr gut	sehr gut			
Fehler der Anbauteile								
Formabweichung des Wellensitzes			sehr gut	weniger gut	sehr gut			
Formabweichung des Lagergehäuses			sehr gut	weniger gut	sehr gut			
Ungenügende Oberflächengüte (Polierflächen)			sehr gut	gut	sehr gut			

Abbildung 63: Frequenzanteile für Fehlermerkmale bei Wälzlagern

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

Fehlermöglichkeiten						
Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet						
An allen Komponenten Risse Korrosion Freß- und Gleiterscheinungen Stromriefen Aufschmelzungen Schmelzkragen						
Kennwerte						
PeakHighFrequency						
CrestFactorHighFrequency						
Vdi3832KtHighFrequency						
PeakRaw						
CrestFactorRaw						
SkewnessRaw						
KurtosisRaw						
Vdi3832KtRaw						
RmsHighFrequency						
RmsAccRaw						
RmsVelRaw						
Iso10816						
RmsAccEnvelope						
RmsVelEnvelope						
FrequencyBandRmsVelEnvelope						
FrequencyBandRmsAccEnvelope						
FrequencyBandRmsAccRaw						
FrequencyBandRmsVelRaw						
FrequencyBandNoiseAccEnvelope						
FrequencyBandNoiseVelEnvelope						
FrequencyBandNoiseAccRaw						
FrequencyBandNoiseVelRaw						

Abbildung 64: Frequenzanteile für Fehlermerkmale bei Wälzlagern

Für die Bedeutung der einzelnen Kennwerte siehe ["Kennwerte"](#) auf Seite 41 und ["Konfiguration"](#) auf Seite 64

5.3.2.6.1 Typischer Außen- und Innenringschaden

Außenringschaden

In den meisten Fällen steht der Außenring still und der Innenring dreht sich. Damit ergibt sich eine klar definierte, feststehende Lastzone. Die meisten Schäden treten in dieser Lastzone auf. Bilden sich Pittings oder andere Oberflächenbeschädigungen, so werden diese von den Wälzkörpern überrollt. Bei der Überrollung entstehen Vibrationen, die an den Gehäuseteilen gemessen werden können.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Außenringschaden	$(1 \times f_a)$	$1 \times f_a, 2 \times f_a, 3 \times f_a \dots$	Bei fortgeschrittener Schädigung sind die Außenringfrequenzen auch im Rohsignalspektrum zu sehen

f_a ... Frequenz des Außenringschadens

Innenringschaden

Ein eintretender Innenringschaden läuft mit der rotierenden Welle mit. Aufgrund der unterschiedlichen Drehzahlen zwischen umlaufenden Wälzkörpern und dem Innenring kommt es zu ausgeprägten Modulationen. Daher zeigen sich im Spektrum die Innenringschadensfrequenzen meist mit Seitenbändern.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Innenringschaden	$(1 \times f_i)$	$i \times f_i \pm i \times f_n$	Der Innenringschaden tritt meist aufgrund der Modulation mit ausgeprägten Seitenbändern der Drehzahl auf.

f_i ... Frequenz des Innenringschadens

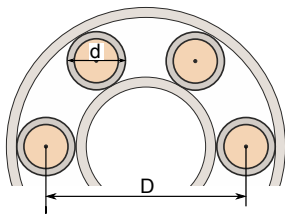
f_n ... Nenndrehzahl

Berechnung der Schadensfrequenzen

Die Lagerschadenfrequenzen werden normalerweise von den Herstellern zur Verfügung gestellt und können den Datenblättern der Lager entnommen werden.

Sie lassen sich allerdings auch auf einfache Art selbst berechnen. Dafür werden folgende Werte benötigt.

N	Geschwindigkeit in rpm
n_b	Anzahl der Wälzelemente
d	Durchmesser der Wälzelemente
β_c	Druckwinkel
D	Wälzringdurchmesser



Formel zur Berechnung der Innenring-Schadensfrequenz

$$f_i = \frac{n_b \cdot N}{2 \cdot 60} \cdot \left[1 + \frac{d}{D} \cdot \cos \beta_c \right]$$

Abbildung 65: Berechnung der Innenring-Schadensfrequenz

f_i ... Schadensfrequenz Innenring

Formel zur Berechnung der Außenring-Schadensfrequenz

$$f_a = \frac{n_b \cdot N}{2 \cdot 60} \cdot \left[1 - \frac{d}{D} \cdot \cos \beta_c \right]$$

Abbildung 66: Berechnung der Außenring-Schadensfrequenz

f_a ... Schadensfrequenz Außenring

Formel zur Berechnung der Schadensfrequenz am Wälzkörper

Schaden schlägt an einzelner Wälzbahn an

$$f_{ew} = \frac{1}{2} \cdot \frac{D \cdot N}{d \cdot 60} \cdot \left[1 - \left(\frac{d}{D} \cdot \cos \beta_c \right)^2 \right]$$

Abbildung 67: Berechnung der Schadensfrequenz an einem Wälzkörper

few ... Schadensfrequenz des einzelnen Wälzkörpers

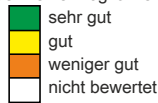
Schaden schlägt an beiden Wälzbahnen an

$$f_w = \frac{D \cdot N}{d \cdot 60} \cdot \left[1 - \left(\frac{d}{D} \cdot \cos \beta_c \right)^2 \right]$$

Abbildung 68: Berechnung der Schadensfrequenz an beiden Wälzkörpern

fw ... Schadensfrequenz beider Wälzkörper

Eignung der jeweiligen
Kennwerte zur Überwachung
der Fehlermöglichkeiten:



Montagefehler

Montagefehler
Verspannung
Beschädigung bei der Montage
Schiefstellung der Lagerringe zueinander
Unsachgemäße Erwärmung beim Einbau
Fremdkörper (Späne, Schmutz)
Unsachgemäße Lagerung, fehlende Konservierung
Unsachgemäßer Transport, fehlende Konservierung
Durch die Montage veränderte Lagerluft
Eindringen von Wasser oder anderen Medien
Schweißstrom
Hammerschläge beim Einbau

Fehler der Anbauteile

Zu lose Passung
Zu feste Passung
Unzureichende Gehäusesteifigkeit
Unpassende Lagerluft

Fehler im Betrieb

Vorangegangene Transportschwingungen
Dreh-/Torsionsschwingungen
Vorangegangene Stillstandsschwingungen
Mikroschwingungen
Betriebsschwingungen
Überschreiten der zulässigen Drehzahl der Wälzlager
Überlastung
Unterschreiten der Mindestbelastung
Durch den Betrieb veränderte Lagerluft
Stetiger Stromdurchgang mit geringer Stromstärke
Überschläge (Strom)
Temperaturschwankungen
Eindringen von Wasser oder aggressiven Medien
Stark wechselnde Dynamik-/Lastwechsel

Fehler mit der Schmierung

Schmierstoffmangel
Überschmierung
Alterung des Schmiermittels
Ungeeignetes Schmiermittel
Verunreinigung des Schmiermittels

Fehler der Anbauteile

Formabweichung des Wellensitzes
Formabweichung des Lagergehäuses
Ungenügende Oberflächengüte (Polierflächen)

Fehlerursachen

An Außen- oder Innenring	Erhöhter Verschleiß	Rollspuren über den Ringumfang	Klassische Ermüdung	Pittings	Ablätungen	Gewaltbruch	Heißlauf	Rollkörpereindrücke	Bruch des Ringes	Elastische-plastische Verformung	Laufspuren an den Ringen	Rattermarken	Schürfmacken	Anschmierungen	Rauhe verformte Flächen	Schälungen
Montagefehler																
Verspannung																
Beschädigung bei der Montage																
Schiefstellung der Lagerringe zueinander																
Unsachgemäße Erwärmung beim Einbau																
Fremdkörper (Späne, Schmutz)																
Unsachgemäße Lagerung, fehlende Konservierung																
Unsachgemäßer Transport, fehlende Konservierung																
Durch die Montage veränderte Lagerluft																
Eindringen von Wasser oder anderen Medien																
Schweißstrom																
Hammerschläge beim Einbau																
Fehler der Anbauteile																
Zu lose Passung																
Zu feste Passung																
Unzureichende Gehäusesteifigkeit																
Unpassende Lagerluft																
Fehler im Betrieb																
Vorangegangene Transportschwingungen																
Dreh-/Torsionsschwingungen																
Vorangegangene Stillstandsschwingungen																
Mikroschwingungen																
Betriebsschwingungen																
Überschreiten der zulässigen Drehzahl der Wälzlager																
Überlastung																
Unterschreiten der Mindestbelastung																
Durch den Betrieb veränderte Lagerluft																
Stetiger Stromdurchgang mit geringer Stromstärke																
Überschläge (Strom)																
Temperaturschwankungen																
Eindringen von Wasser oder aggressiven Medien																
Stark wechselnde Dynamik-/Lastwechsel																
Fehler mit der Schmierung																
Schmierstoffmangel																
Überschmierung																
Alterung des Schmiermittels																
Ungeeignetes Schmiermittel																
Verunreinigung des Schmiermittels																
Fehler der Anbauteile																
Formabweichung des Wellensitzes																
Formabweichung des Lagergehäuses																
Ungenügende Oberflächengüte (Polierflächen)																

Abbildung 69: Frequenzanteile für Fehlermerkmale bei Wälzlagern

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

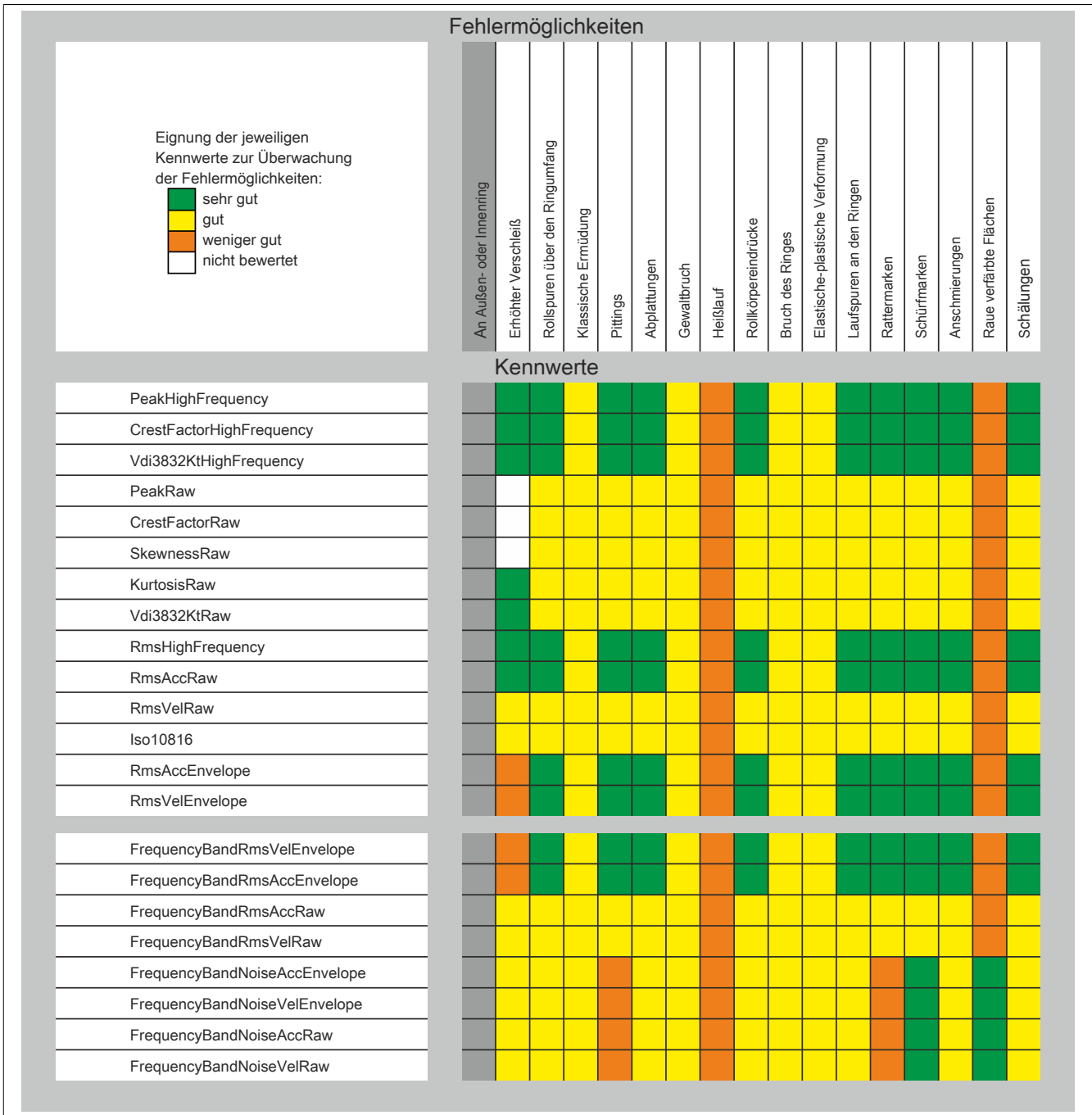


Abbildung 70: Frequenzanteile für Fehlermerkmale bei Wälzlagern

Für die Bedeutung der einzelnen Kennwerte siehe ["Kennwerte"](#) auf Seite 41 und ["Konfiguration"](#) auf Seite 64

5.3.2.6.2 Typischer Käfig- und Wälzkörperschaden

Käfigschaden

Die Käfigfrequenzen treten häufig bei Elektromotoren auf, insbesondere bei der Verwendung von Lagern mit höherer Lagerluft. Wird diese nicht ausgenutzt, so kommt es oftmals zum Schwingen des Käfigs. Dies zeigt sich dann in einem erhöhten Laufgeräusch.

Wälzkörperschaden

Ein Wälzkörperschaden ohne Außen- oder Innenringschaden tritt äußerst selten auf, die Einzelmerkmale sind daher hier nur exemplarisch angeführt.

Bei einem Schaden am Wälzkörper entsteht ein Schlag entweder am Innenring, am Außenring oder an beiden. Somit kann man einen Wälzkörperschaden durch die Überrollfrequenz bzw. der Doppelten davon erkennen. Aus diesem Grund sollte man, wenn möglich, für die Erkennung eines Wälzkörperschadens die Harmonischen in die Kennwertberechnung miteinbeziehen.

Für das Schadensbild siehe ["Typischer Außen- und Innenringschaden"](#) auf Seite 141

Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten: häufig Abhängig vom Einsatzort selten bis nie nicht bewertet									
Fehlerursachen									
Montagefehler									
Montagefehler									
Verspannung									
Beschädigung bei der Montage									
Schiefstellung der Lagerringe zueinander									
Unsachgemäße Erwärmung beim Einbau									
Fremdkörper (Späne, Schmutz)									
Unsachgemäße Lagerung, fehlende Konservierung									
Unsachgemäßer Transport, fehlende Konservierung									
Durch die Montage veränderte Lagerluft									
Eindringen von Wasser oder anderen Medien									
Schweißstrom									
Hammerschläge beim Einbau									
Fehler der Anbauteile									
Zu lose Passung									
Zu feste Passung									
Unzureichende Gehäusesteifigkeit									
Unpassende Lagerluft									
Fehler im Betrieb									
Vorangegangene Transportschwingungen									
Dreh-/Torsionsschwingungen									
Vorangegangene Stillstandsschwingungen									
Mikroschwingungen									
Betriebsschwingungen									
Überschreiten der zulässigen Drehzahl der Wälzlager									
Überlastung									
Unterschreiten der Mindestbelastung									
Durch den Betrieb veränderte Lagerluft									
Stetiger Stromdurchgang mit geringer Stromstärke									
Überschläge (Strom)									
Temperaturschwankungen									
Eindringen von Wasser oder aggressiven Medien									
Stark wechselnde Dynamik-/Lastwechsel									
Fehler mit der Schmierung									
Schmierstoffmangel									
Überschmierung									
Alterung des Schmiermittels									
Ungeeignetes Schmiermittel									
Verunreinigung des Schmiermittels									
Fehler der Anbauteile									
Formabweichung des Wellensitzes									
Formabweichung des Lagergehäuses									
Ungenügende Oberflächengüte (Polierflächen)									

Abbildung 71: Frequenzanteile für Fehlermerkmale bei Wälzlagern

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

Fehlermöglichkeiten									
Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet		An den Wälzkörpern	Wälzkörperschäden an der Oberfläche	Erhöhter Verschleiß	Wälzkörperrisse	Wälzkörperbruch	Anschmierungen	Abplattungen	Am Käfig
									Käfigbruch
									Erhöhter Käfigverschleiß
Kennwerte									
PeakHighFrequency									
CrestFactorHighFrequency									
Vdi3832KtHighFrequency									
PeakRaw									
CrestFactorRaw									
SkewnessRaw									
KurtosisRaw									
Vdi3832KtRaw									
RmsHighFrequency									
RmsAccRaw									
RmsVelRaw									
Iso10816									
RmsAccEnvelope									
RmsVelEnvelope									
FrequencyBandRmsVelEnvelope									
FrequencyBandRmsAccEnvelope									
FrequencyBandRmsAccRaw									
FrequencyBandRmsVelRaw									
FrequencyBandNoiseAccEnvelope									
FrequencyBandNoiseVelEnvelope									
FrequencyBandNoiseAccRaw									
FrequencyBandNoiseVelRaw									

Abbildung 72: Frequenzanteile für Fehlermerkmale bei Wälzlagern

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.2.7 Getriebebeschäden

In der DIN 3979 sind Fehler und Defekte eines Getriebes aufgelistet - die häufigsten Fehler an einfachen Getrieben werden im Weiteren beschrieben.

Die Komplexität der Maschinendynamik ist bei den einzelnen Anwendungsfällen zu beachten.

Übersetzung

Zu beachten ist die Umrechnung der Drehzahlverhältnisse entsprechen der Getriebeübersetzung. Die Frequenzen an den Zahnrädern sind immer auf die Drehzahl der jeweiligen Achse bezogen.

5.3.2.7.1 Fertigungsbedingte Defekte

Fertigungsbedingt treten an einem Getriebe immer Schwingungen auf. Typische Fertigungsfehler sind Teilungsfehler, Profilabweichungen, Rundlauf- und Abstandsfehler.

Diese Einzelfehler können sich nun je nach Zahnradpaarung verstärken oder abschwächen, je nachdem wie die Fehler aufeinandertreffen. Das Aufeinandertreffen der Einzelfehler ist damit auch verantwortlich für das Gesamt-schwingungsverhalten.

Als Beispiel sei hier der häufig auftretende Teilungsfehler genannt. Je nachdem ob der Teilungsfehler positiv oder negativ ist, wird dieser durch eine vorhandene Belastung verstärkt oder kompensiert. Die Auswirkungen auf das Schwingungsverhalten hängen sehr stark von den Steifigkeiten der Zahnräder ab. Tritt ein Teilungsfehler auf, so kommt es im Augenblick des Eingriffs zu einem Stoß, der sehr gut gemessen werden kann.

Neben dem Teilungsfehler führen alle Form- und Maßfehler an den Zahnrädern zu Schwingungen.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Zahneingriffsfrequenzen	$1 \times fz$	$1 \times fz$	Die auftretenden Getriebefrequenzen sind von den jeweiligen Geometrieverhältnissen der Räder abhängig, können aber in jedem Fall eindeutig berechnet werden.

fz ... Zahneingriffsfrequenz

5.3.2.7.2 Verschleißbedingte Defekte

Wenn Form- und Maßabweichungen im Laufe des Betriebes auftreten, kann die Betrachtung des Trends zu einer entsprechenden Diagnosesicherheit führen.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Verschleiss	$1 \times fz$	$1 \times fz \pm i \times fn$	Geometriefehler zeigen sich vermehrt auch mit Seitenbändern zur Zahneingriffsfrequenz

fz ... Zahneingriffsfrequenz

fn ... Nenndrehzahl

5.3.2.7.3 Taumel

Wenn die Verzahnungsachse und die Rotationsachse nicht parallel zueinander liegen tritt sogenanntes Taumeln auf. Pro Umdrehung kommt es damit 2x zu einer Flankenabweichung. Je nach Stellung ist die Kraftübertragung einmal an der Innenkante und einmal an der Außenkante des Zahnrades.

Im Frequenzspektrum ist die doppelte Drehzahlfrequenz gut zu sehen.

Schadensbild	Frequenzanteile im Rohsignalspektrum	Frequenzanteile im Hüllkurvenspektrum	Bemerkung
Taumel	$1 \times fn, 2 \times fn$	$1 \times fz, 2 \times fz$	Die Taumelbewegung zeigt sich mit doppelter Drehfrequenz, zu-meist auch noch mit Seitenbändern.

fn ... Nenndrehzahl

fz ... Zahneingriffsfrequenz

5.3.2.7.4 Rundlauf- und Achsabstandsfehler

In der DIN 3960 ist der Achsabstandsfehler als Abweichung vom Soll- zum Istwert definiert.

Bei einem Fehler, welcher den Abstand von Achsen zueinander verändert, kommt es auch zu einer Änderung der Berührung der beiden Zahnräder. Der sogenannte Überdeckungsgrad wird negativ beeinflusst.

Bereits geringe Abstandsfehler führen zu einem erhöhten Geräusch des Getriebes.

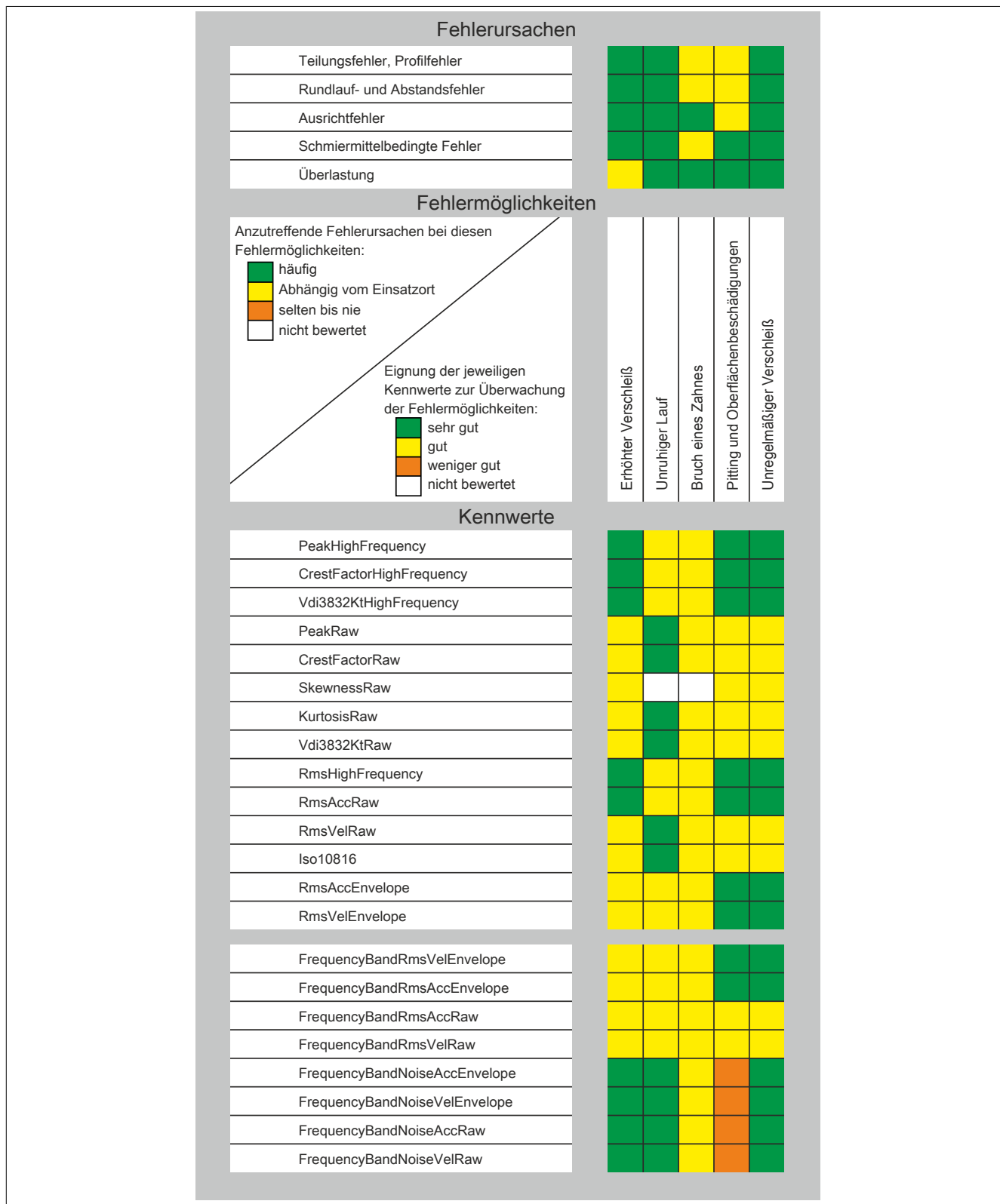


Abbildung 73: Frequenzanteile für Fehlermerkmale bei Getriebschäden

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.2.8 Elektrischer Fehler

Gelegentlich kommt es zu Brücken in den Rotorstäben oder den Kurzschlusskäfigen. Diese entstehen aufgrund von Überlastungen, aber auch durch Alterung und der laufenden Schwingungsbelastung. Dies führt zu einer sehr ungleichmäßigen Verteilung des Induktionsstromes im Rotor.

Fehlerursachen				
Überlastung				
Alterung				
Staubeintritt				
Fremdkörper				
Windungsschlüsse				
Fehlermöglichkeiten				
Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten: häufig Abhängig vom Einsatzort selten bis nie nicht bewertet				
Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet				
	Lose Stäbe, Stabbruch	Erhöhtes Laufgeräusch	Bruch der Kurzschlussringe	Magnetische Unwucht
Kennwerte				
PeakHighFrequency				
CrestFactorHighFrequency				
Vdi3832KtHighFrequency				
PeakRaw				
CrestFactorRaw				
SkewnessRaw				
KurtosisRaw				
Vdi3832KtRaw				
RmsHighFrequency				
RmsAccRaw				
RmsVelRaw				
Iso10816				
RmsAccEnvelope				
RmsVelEnvelope				
FrequencyBandRmsVelEnvelope				
FrequencyBandRmsAccEnvelope				
FrequencyBandRmsAccRaw				
FrequencyBandRmsVelRaw				
FrequencyBandNoiseAccEnvelope				
FrequencyBandNoiseVelEnvelope				
FrequencyBandNoiseAccRaw				
FrequencyBandNoiseVelRaw				

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.3 Einsatzmöglichkeiten typische Schadenserkennung

Die im Weiteren angeführten Beispiele sind typische Anwendungsfälle und sollen als Hilfestellung bei der Integration dienen. Eine Detailplanung der Anwendungen muss aber für jeden Einsatzfall einzeln erfolgen.

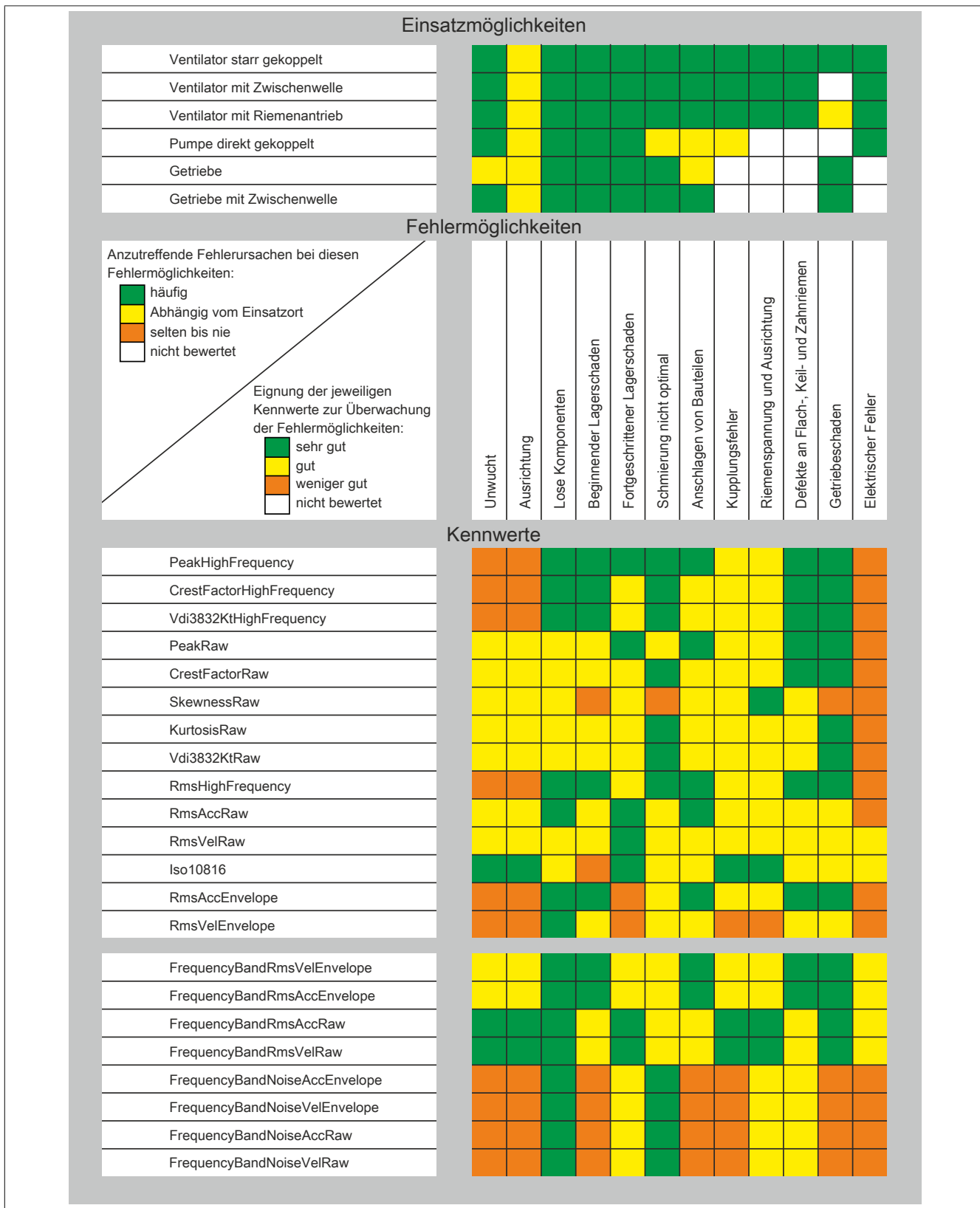


Abbildung 74: Einsatzmöglichkeiten typischer Schadenserkennung

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.3.1 Ventilator starr gekoppelt

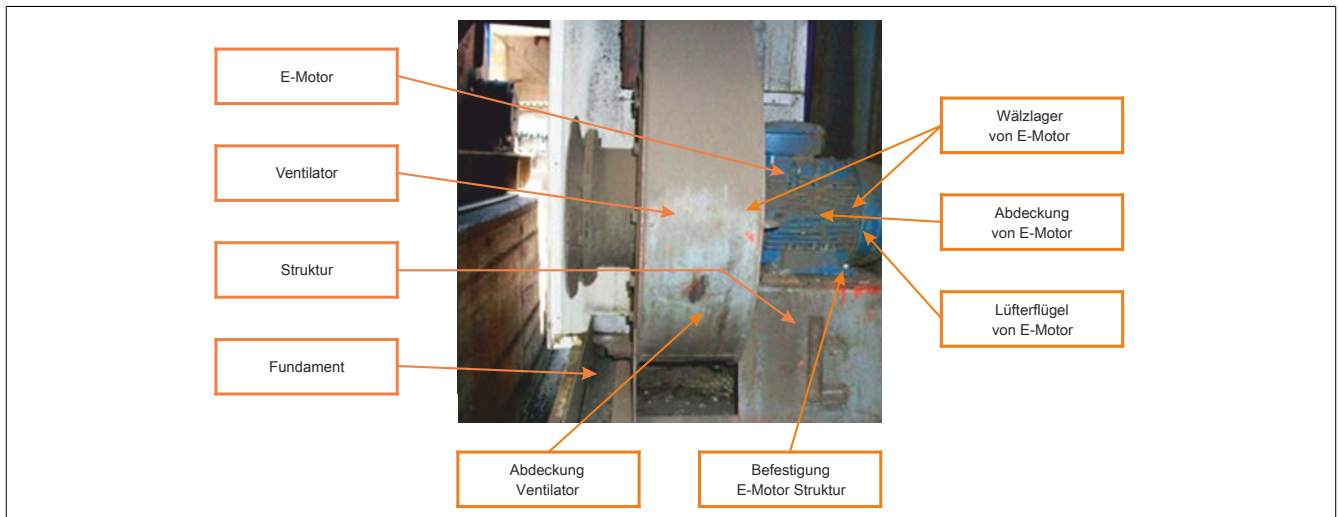


Abbildung 75: Antriebseinheit mit Ventilator

Condition Monitoring Lösung

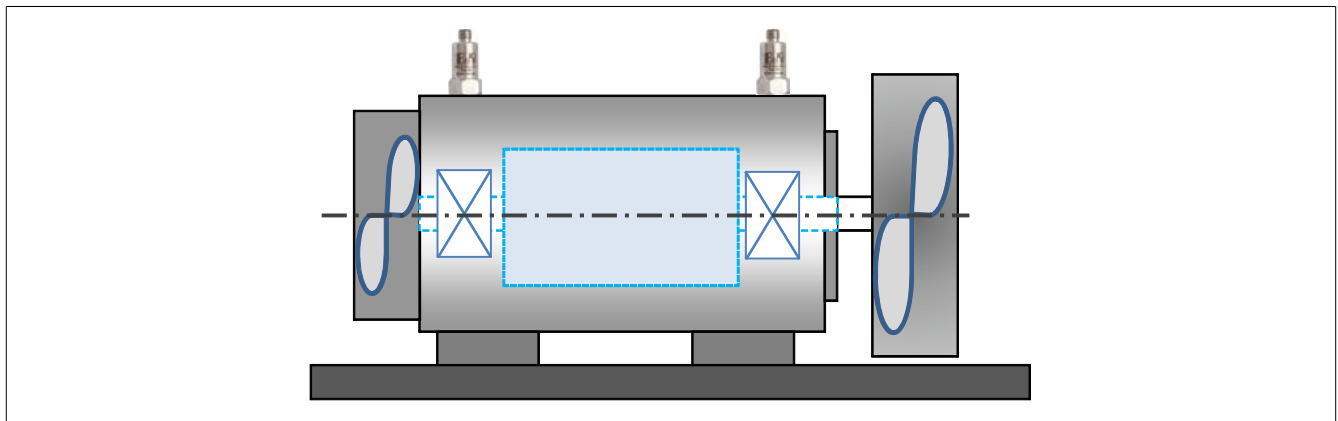


Abbildung 76: Systemskizze - Antriebseinheit mit Ventilator

Sensoreinsatz:

Sensoranzahl	Im Normalfall 2 Sensoren, bei kleineren Antriebseinheiten ist ein Sensor ausreichend.
Sensormontage	Vorzugsweise in vertikaler Richtung, bei Bedarf kann auch eine horizontale Montage erfolgen.

Ventilator starr gekoppelt - Häufige Problemstellungen:

Einsatzmöglichkeiten									
Elektromotor									
Ventilator									
Fehlermöglichkeiten									
Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten: häufig Abhängig vom Einsatzort seltener bis nie nicht bewertet									
Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet									
		Unwucht	Ausrichtung	Lose Komponenten	Lagerschaden im Frühstadium	Lagerschaden in Spätstadium	Schmierung nicht optimal	Anschlagen von Bauteilen	Elektrischer Fehler
Kennwerte									
PeakHighFrequency									
CrestFactorHighFrequency									
Vdi3832KtHighFrequency									
PeakRaw									
CrestFactorRaw									
SkewnessRaw									
KurtosisRaw									
Vdi3832KtRaw									
RmsHighFrequency									
RmsAccRaw									
RmsVelRaw									
Iso10816									
RmsAccEnvelope									
RmsVelEnvelope									
FrequencyBandRmsVelEnvelope									
FrequencyBandRmsAccEnvelope									
FrequencyBandRmsAccRaw									
FrequencyBandRmsVelRaw									
FrequencyBandNoiseAccEnvelope									
FrequencyBandNoiseVelEnvelope									
FrequencyBandNoiseAccRaw									
FrequencyBandNoiseVelRaw									

Für die Bedeutung der einzelnen Kennwerte siehe ["Kennwerte"](#) auf Seite 41 und ["Konfiguration"](#) auf Seite 64

5.3.3.2 Ventilator mit Zwischenwelle

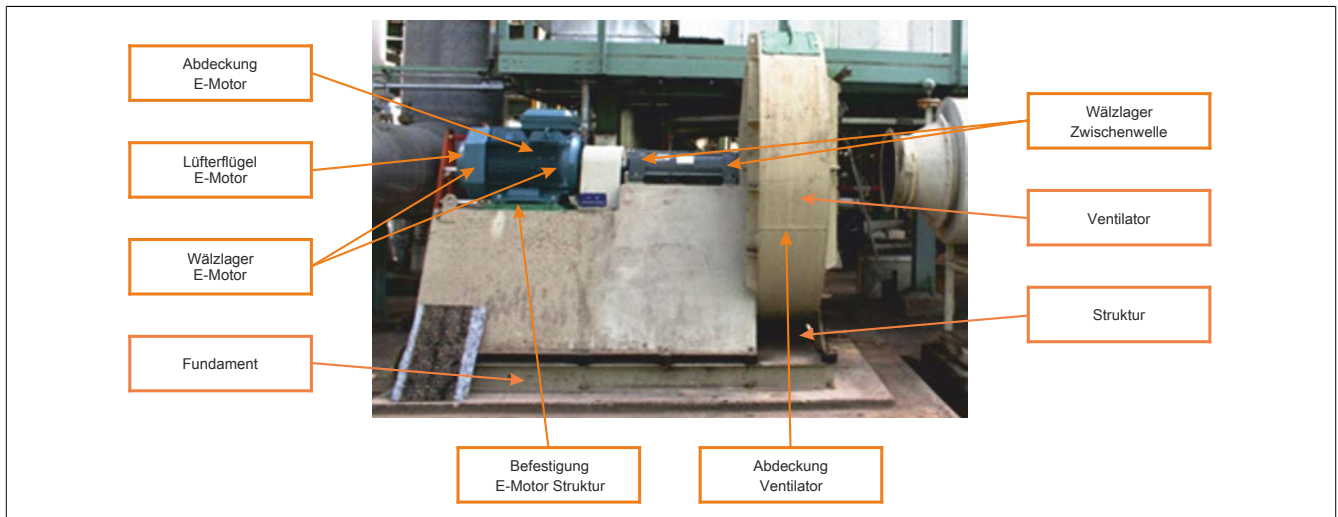


Abbildung 77: Antriebseinheit mit Ventilator

Condition Monitoring Lösung

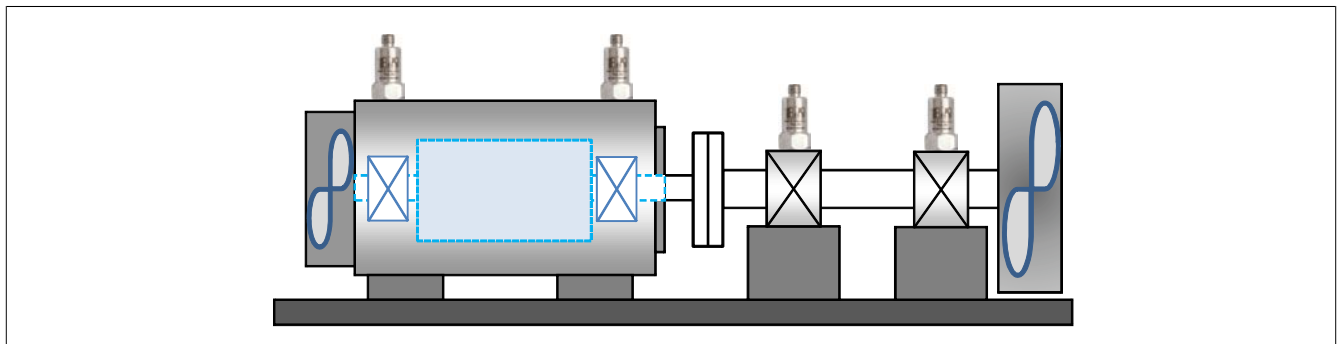


Abbildung 78: Systemskizze - Antriebseinheit mit Zwischenwelle und Ventilator

Sensoreinsatz:

Sensoranzahl	Im Normalfall 4 Sensoren, bei kleineren Antriebseinheiten ist ein Sensor ausreichend.
Sensormontage	Vorzugsweise in vertikaler Richtung, bei Bedarf kann auch eine horizontale Montage erfolgen.

Ventilator mit Zwischenwelle - Häufige Problemstellungen:

Einsatzmöglichkeiten									
Elektromotor									
Ventilator									
Kupplung									
Fehlermöglichkeiten									
Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten: häufig Abhängig vom Einsatzort selten bis nie nicht bewertet									
Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet									
					Unwucht	Ausrichtung	Lose Komponenten	Lagerschaden im Frühstadium	Lagerschaden in Spätstadium
								Schmierung nicht optimal	Anschlagen von Bauteilen
								Kupplungsfehler	Elektrischer Fehler
Kennwerte									
PeakHighFrequency									
CrestFactorHighFrequency									
Vdi3832KtHighFrequency									
PeakRaw									
CrestFactorRaw									
SkewnessRaw									
KurtosisRaw									
Vdi3832KtRaw									
RmsHighFrequency									
RmsAccRaw									
RmsVelRaw									
Iso10816									
RmsAccEnvelope									
RmsVelEnvelope									
FrequencyBandRmsVelEnvelope									
FrequencyBandRmsAccEnvelope									
FrequencyBandRmsAccRaw									
FrequencyBandRmsVelRaw									
FrequencyBandNoiseAccEnvelope									
FrequencyBandNoiseVelEnvelope									
FrequencyBandNoiseAccRaw									
FrequencyBandNoiseVelRaw									

Für die Bedeutung der einzelnen Kennwerte siehe ["Kennwerte"](#) auf Seite 41 und ["Konfiguration"](#) auf Seite 64

5.3.3.3 Ventilator mit Riemenantrieb

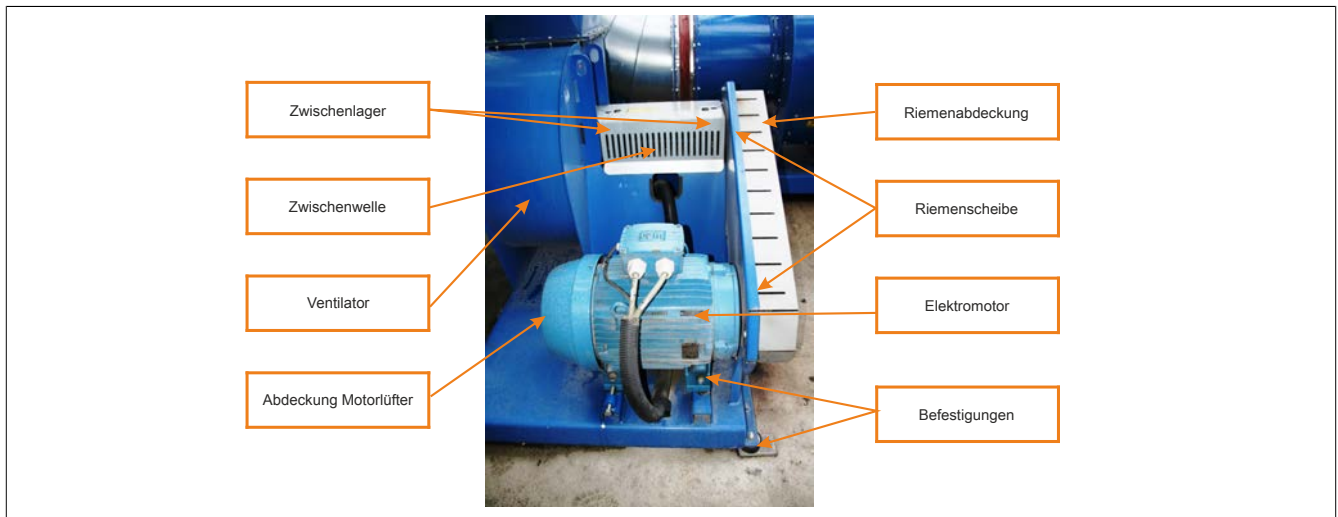


Abbildung 79: Aufbau eines Ventilators mit Riemenantrieb

Condition Monitoring Lösungen

Anordnung A

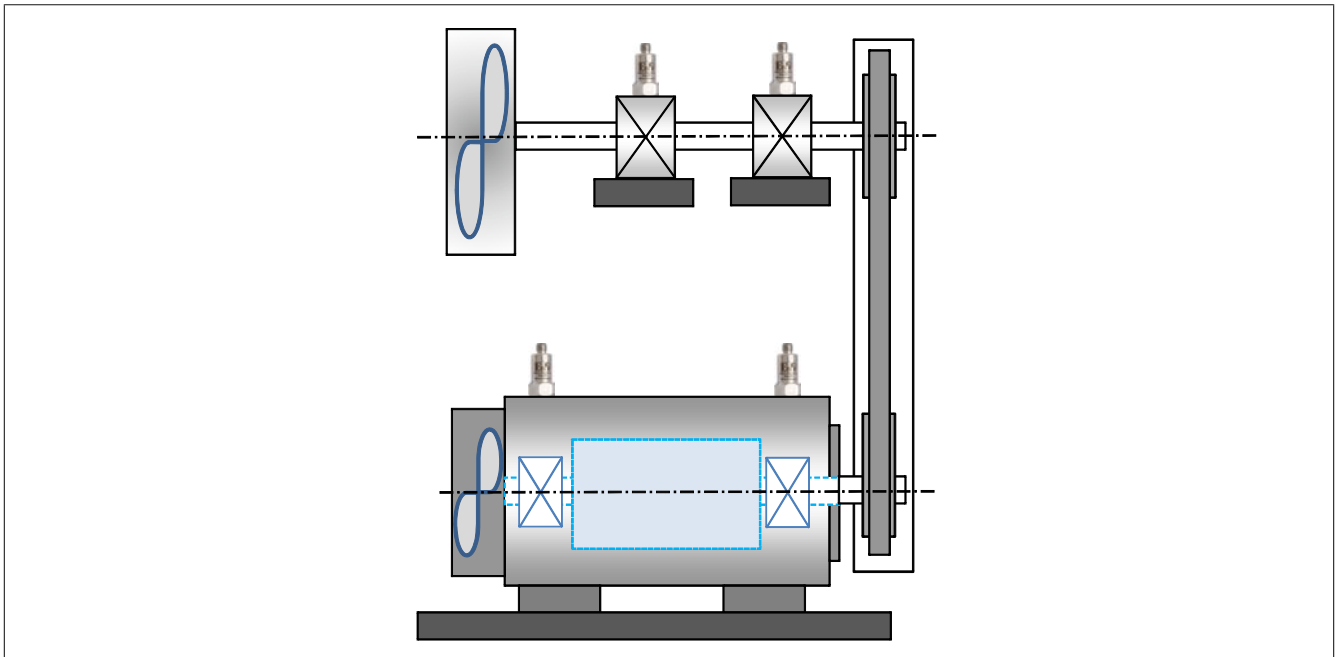


Abbildung 80: Systemskizze - Aufbau eines Ventilators mit Riemenantrieb

Anordnung B

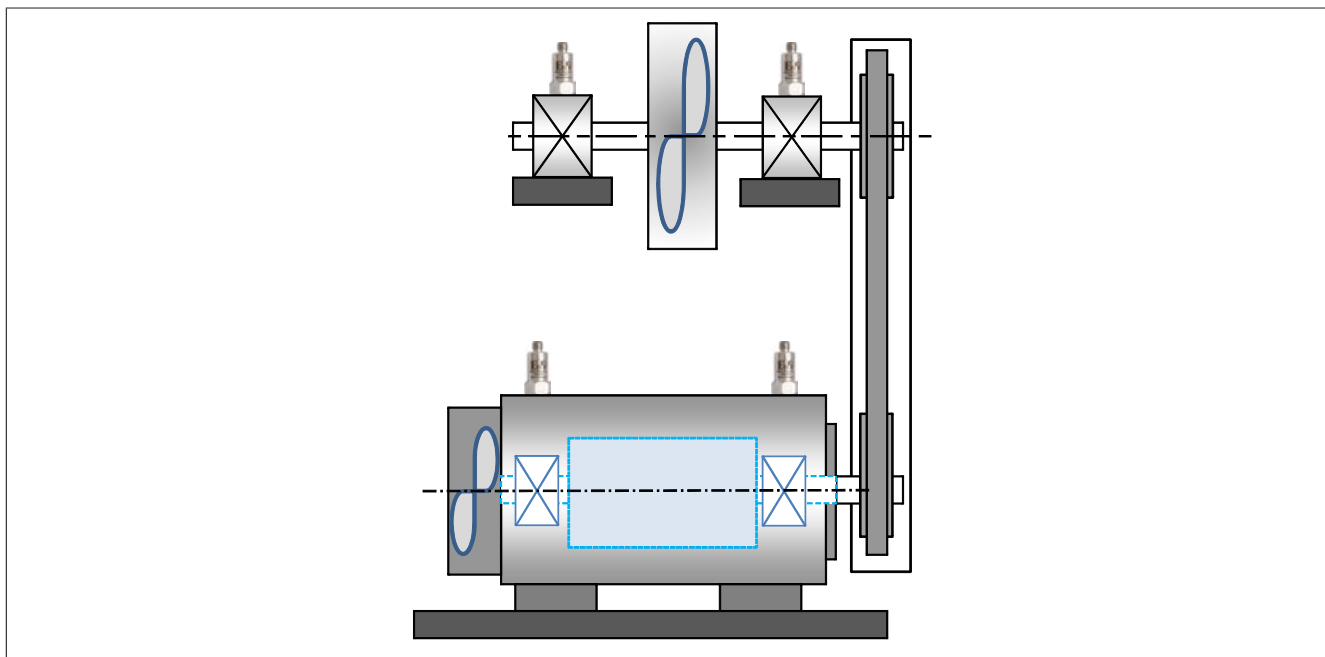


Abbildung 81: Systemskizze - Aufbau eines Ventilators mit Riemenantrieb - alternative Lagerung

Sensoreinsatz:

Sensoranzahl	im Normalfall 4 Sensoren, bei kleineren Antriebseinheiten ist ein Sensor ausreichend.
Sensormontage	Vorzugsweise in vertikaler Richtung, bei Bedarf auch eine horizontale Montage erfolgen. Eine Fehlausrichtung des Riemens zeigt sich besonders in axialer Richtung.

Ventilator mit Riemenantrieb - Häufige Problemstellungen:

Einsatzmöglichkeiten									
Elektromotor									
Ventilator									
Flach- und Keilriemen									
Zahnriemen									
Fehlermöglichkeiten									
Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten: häufig Abhängig vom Einsatzort selten bis nie nicht bewertet									
Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet									
					Unwucht	Ausrichtung	Lose Komponenten	Lagerschaden im Frühstadium	Lagerschaden in Spätstadium
									Schmierung nicht optimal
									Anschlagen von Bauteilen
									Riemenspannung und Ausrichtung
									Elektrischer Fehler
Kennwerte									
PeakHighFrequency									
CrestFactorHighFrequency									
Vdi3832KtHighFrequency									
PeakRaw									
CrestFactorRaw									
SkewnessRaw									
KurtosisRaw									
Vdi3832KtRaw									
RmsHighFrequency									
RmsAccRaw									
RmsVelRaw									
Iso10816									
RmsAccEnvelope									
RmsVelEnvelope									
FrequencyBandRmsVelEnvelope									
FrequencyBandRmsAccEnvelope									
FrequencyBandRmsAccRaw									
FrequencyBandRmsVelRaw									
FrequencyBandNoiseAccEnvelope									
FrequencyBandNoiseVelEnvelope									
FrequencyBandNoiseAccRaw									
FrequencyBandNoiseVelRaw									

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.3.4 Pumpe direkt gekoppelt



Abbildung 82: Aufbau eines Pumpenantriebs

Condition Monitoring Lösung

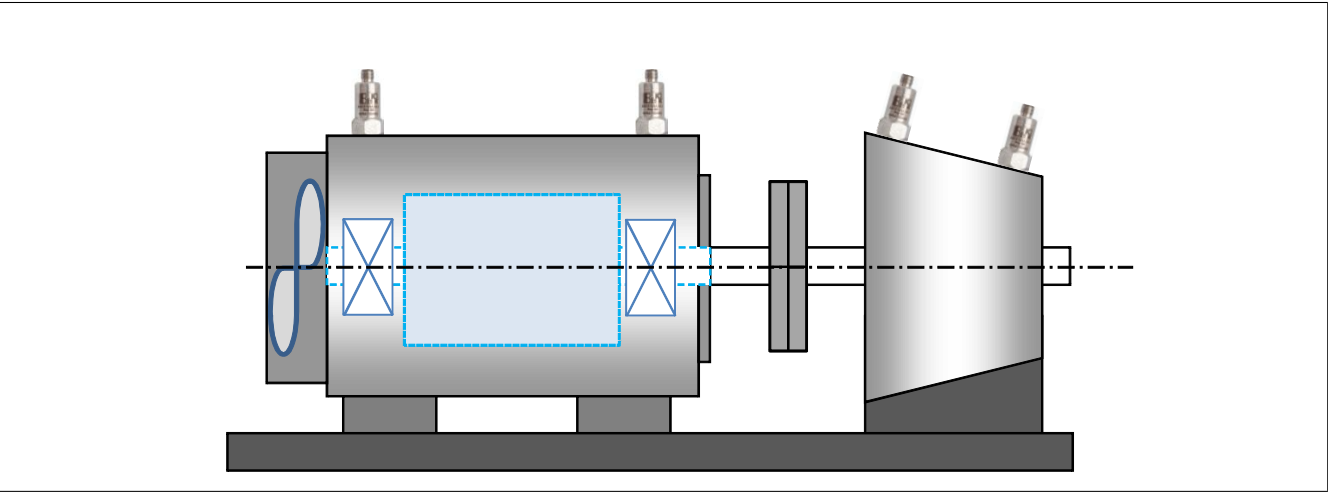


Abbildung 83: Systemskizze - Aufbau eines Pumpenantriebs

Sensoreinsatz:

Sensoranzahl	Im Normalfall 4 Sensoren, bei kleineren Antriebseinheiten sind auch 2 Sensoren ausreichend.
Sensormontage	Vorzugsweise in vertikaler Richtung, bei Bedarf kann auch eine horizontale Montage erfolgen.

5.3.3.5 Getriebe

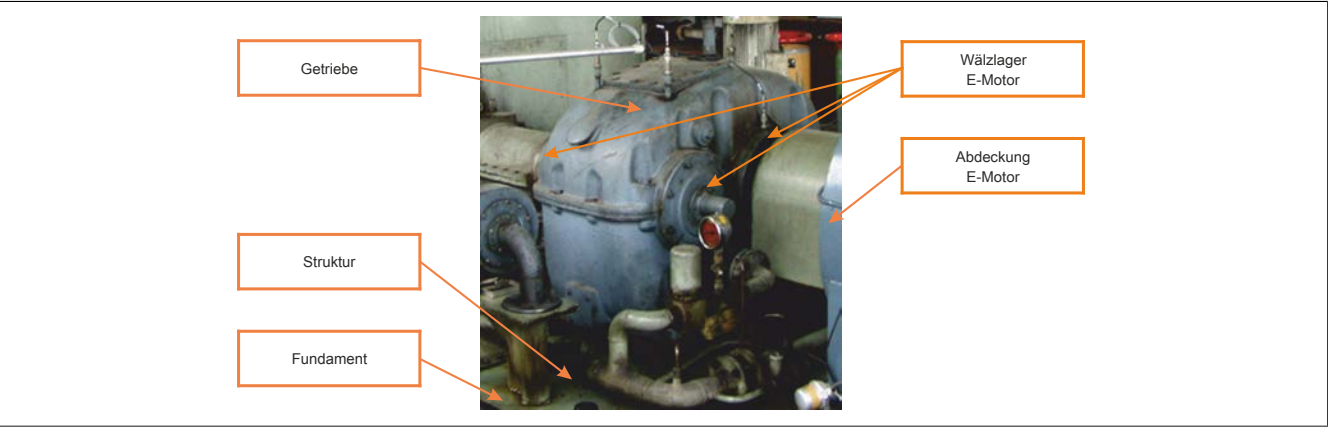


Abbildung 84: Aufbau eines Getriebes

Condition Monitoring Lösung

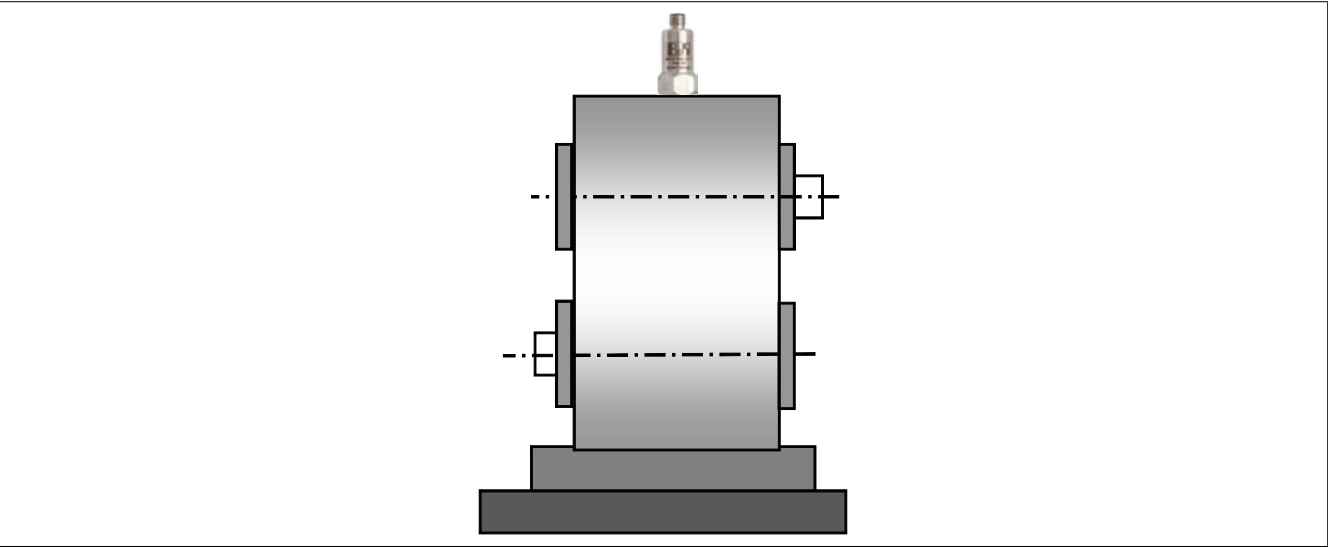


Abbildung 85: Systemskizze - Aufbau eines Getriebes

Sensoreinsatz:

Sensoranzahl	Die Sensoranzahl ist vom Typ und der Größe des Getriebes abhängig.
Sensormontage	Vorzugsweise in vertikaler Richtung, bei Bedarf kann auch eine horizontale Montage erfolgen. Die Montagerichtung ist stark abhängig von der Belastungsrichtung des Getriebes.

Getriebe - Häufige Problemstellungen:

Einsatzmöglichkeiten														
Getriebe														
Fehlermöglichkeiten														
Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten: häufig Abhängig vom Einsatzort selten bis nie nicht bewertet														
Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet														
			Unwucht	Ausrichtung	Lose Komponenten	Lagerschaden im Frühstadium	Lagerschaden in Spätstadium	Schmierung nicht optimal	Anschlagen von Bauteilen	Kupplungsfehler	Riemenspannung und Ausrichtung	Defekte an Flach-, Keil- und Zahnriemen	Getriebebeschaden	Elektrischer Fehler
Kennwerte														
PeakHighFrequency														
CrestFactorHighFrequency														
Vdi3832KtHighFrequency														
PeakRaw														
CrestFactorRaw														
SkewnessRaw														
KurtosisRaw														
Vdi3832KtRaw														
RmsHighFrequency														
RmsAccRaw														
RmsVelRaw														
Iso10816														
RmsAccEnvelope														
RmsVelEnvelope														
FrequencyBandRmsVelEnvelope														
FrequencyBandRmsAccEnvelope														
FrequencyBandRmsAccRaw														
FrequencyBandRmsVelRaw														
FrequencyBandNoiseAccEnvelope														
FrequencyBandNoiseVelEnvelope														
FrequencyBandNoiseAccRaw														
FrequencyBandNoiseVelRaw														

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.3.3.6 Getriebe mit Zwischenwelle

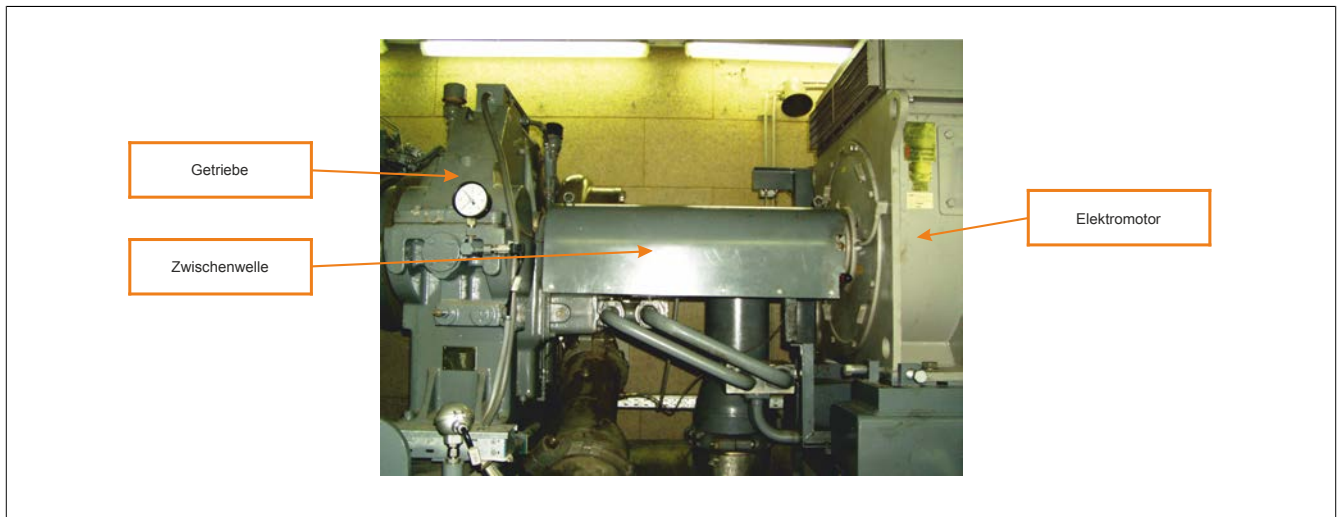


Abbildung 86: Aufbau eines Getriebes mit Zwischenwelle

Condition Monitoring Lösung

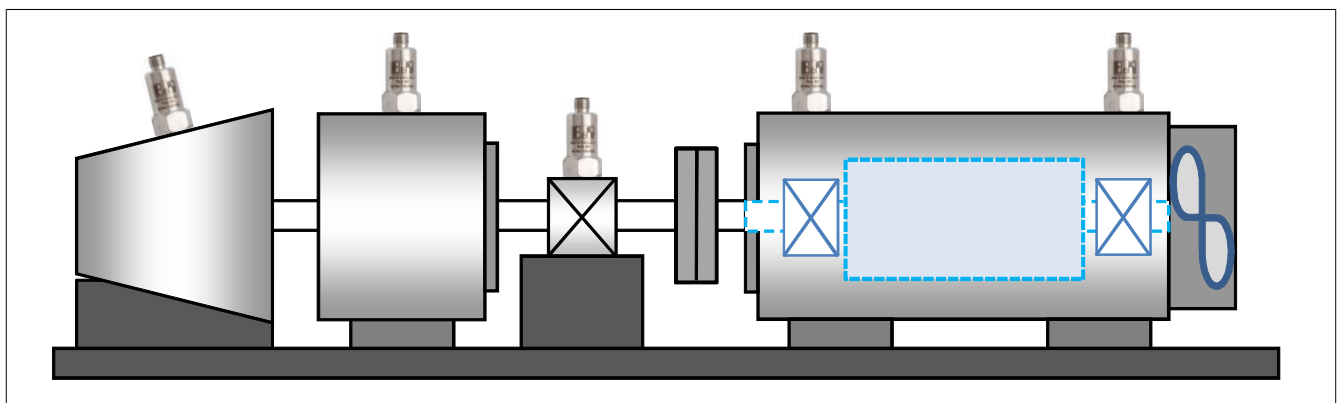


Abbildung 87: Systemskizze - Getriebe mit Zwischenwelle

Sensoreinsatz:

Sensoranzahl	Im Normalfall 5 Sensoren, bei kleineren und starr gekoppelten Antriebseinheiten sind auch 2 Sensoren ausreichend.
Sensormontage	Vorzugsweise in vertikaler Richtung, bei Bedarf kann auch eine horizontale Montage erfolgen.

Getriebe mit Zwischenwelle - Häufige Problemstellungen:

Einsatzmöglichkeiten											
Elektromotor											
Kupplung											
Pumpe											
Getriebe											
Fehlermöglichkeiten											
Anzutreffende Fehlerursachen bei diesen Fehlermöglichkeiten: häufig Abhängig vom Einsatzort selten bis nie nicht bewertet Eignung der jeweiligen Kennwerte zur Überwachung der Fehlermöglichkeiten: sehr gut gut weniger gut nicht bewertet											
	Unwucht	Ausrichtung	Lose Komponenten	Lagerschaden im Frühstadium	Lagerschaden in Spätstadium	Schmierung nicht optimal	Anschlagen von Bauteilen	Kupplungsfehler	Riemenspannung und Ausrichtung	Defekte an Flach-, Keil- und Zahnriemen	Getriebebeschaden
											Elektrischer Fehler
Kennwerte											
PeakHighFrequency											
CrestFactorHighFrequency											
Vdi3832KtHighFrequency											
PeakRaw											
CrestFactorRaw											
SkewnessRaw											
KurtosisRaw											
Vdi3832KtRaw											
RmsHighFrequency											
RmsAccRaw											
RmsVelRaw											
Iso10816											
RmsAccEnvelope											
RmsVelEnvelope											
FrequencyBandRmsVelEnvelope											
FrequencyBandRmsAccEnvelope											
FrequencyBandRmsAccRaw											
FrequencyBandRmsVelRaw											
FrequencyBandNoiseAccEnvelope											
FrequencyBandNoiseVelEnvelope											
FrequencyBandNoiseAccRaw											
FrequencyBandNoiseVelRaw											

Für die Bedeutung der einzelnen Kennwerte siehe "Kennwerte" auf Seite 41 und "Konfiguration" auf Seite 64

5.4 Weiterführende Literatur

In diesem Anwenderhandbuch kann der gesamte Themenkreis der Schwingungsanalyse wegen seines Umfanges nur in Grundzügen gestreift werden.

Wer sich tiefer in die Materie einarbeiten will, dem kann folgendes, für Einsteiger gut geeignetes Buch empfohlen werden.

Zustandsüberwachung von Maschinen

Verlag: Expert-Verlag GmbH.
Autor: Dr. Josef Kolerus und Prof. Dr. Johann Wassermann
Auflage: 5., neu bearb. Aufl. 2011
Sprache: Deutsch
Seiten: 408
ISBN-13: 978-3-8169-3080-8

6 Zubehör

6.1 Sensoren

6.1.1 0ACS100A.00-1

6.1.1.1 Bestelldaten

Bestellnummer	Kurzbeschreibung	Abbildung
	Sensoren	
0ACS100A.00-1	Beschleunigungssensor, nominale Empfindlichkeit 100 mV/g, Ausgang oben	
	Erforderliches Zubehör	
	Sensorkabel	
0ACC0020.01-1	Kabel für Beschleunigungssensor, Länge 2 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0050.01-1	Kabel für Beschleunigungssensor, Länge 5 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0100.01-1	Kabel für Beschleunigungssensor, Länge 10 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0150.01-1	Kabel für Beschleunigungssensor, Länge 15 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0200.01-1	Kabel für Beschleunigungssensor, Länge 20 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0500.01-1	Kabel für Beschleunigungssensor, Länge 50 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC1000.01-1	Kabel für Beschleunigungssensor, Länge 100 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	

Tabelle 9: 0ACS100A.00-1 - Bestelldaten

6.1.1.2 Technische Daten

Bestellnummer	0ACS100A.00-1
Sensor Eigenschaften	
Eigenresonanz (montiert)	22 kHz (nominal)
Empfindlichkeit	100 mV/g $\pm 10\%$ nominal 80 Hz bei 22°C
Frequenzverhalten	2 Hz bis 10 kHz $\pm 5\%$ 0,8 Hz bis 15 kHz ± 3 dB
Isolation	Isolierte Basis
Messbereich	± 50 g
Querempfindlichkeit	<5%
Elektrische Eigenschaften	
Elektrische Störungen	max. 0,1 mg
Breitbandauflösung	0,2 mg (200 μ g) über 1 Hz bis 15 kHz
Spektralrauschen	10 Hz bis 10 μ g/Hz 100 Hz bis 4 μ g/Hz 1 kHz bis 3 μ g/Hz
Strombereich	0,5 bis 8 mA
Arbeitspunktspannung	10 bis 12 VDC
Einschwingzeit	2 s
Ausgangsimpedanz	max. 200 Ω
Gehäuseisolation	>10 ⁸ Ω bei 500 V
Einsatzbedingungen	
Schutzart nach EN 60529	IP67
Umgebungsbedingungen	
Temperatur	
Betrieb	-55 bis 140°C
max. Stoßfestigkeit	5000 g
Störaussendungen	EN61000-6-4:2001
Störfestigkeit	EN61000-6-2:1999
Mechanische Eigenschaften	
Gehäuse	
Material	Edelstahl
Montage	M8 x 1,25 x 6 mm Bolzen, am Sensor vormontiert
Gewicht	110 g
Messelement	PZT- Piezoelektronischer Kristall (Blei-Zirkonat-Titanat)
Messauführung	komprimiert
Anzugsmoment	8 Nm
Anschlussstecker	M12

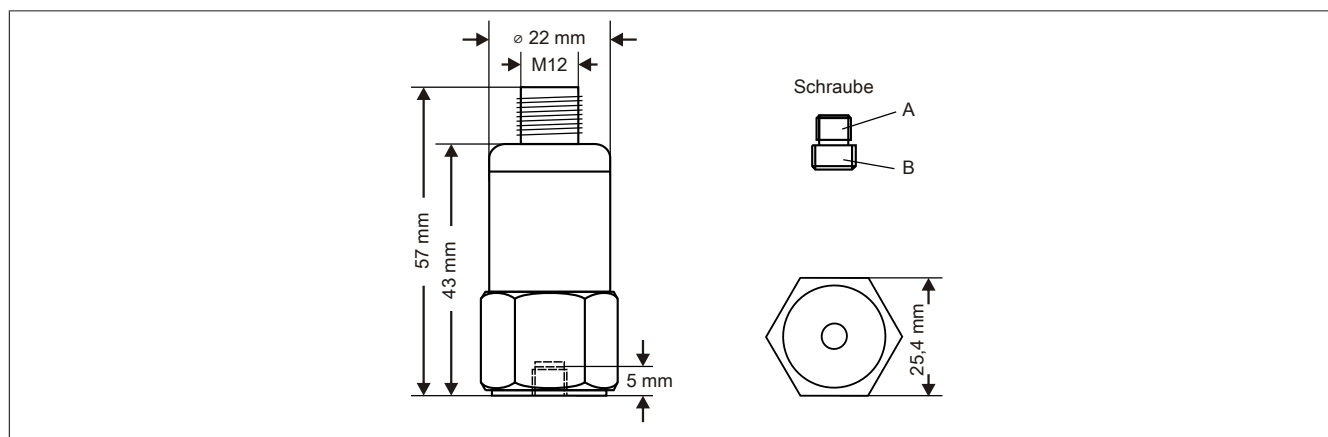
Tabelle 10: 0ACS100A.00-1 - Technische Daten

Die für den Sensor gültigen Zertifizierungen sind auf der Homepage des Herstellers zu finden.

**Zertifikate**

<http://www.hansfordsensors.com/resources/certificates/>

6.1.1.3 Abmessungen

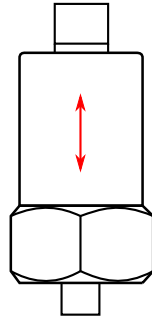


Schraube wird vormontiert mitgeliefert.

- A 1/4" - 28 UNF (Maximale Gewindelänge: 5 mm)
- B M8 x 6 x 1,25

6.1.1.4 Einbaurichtung

Der B&R Schwingungssensor 0ACS100A.00-1 ist für Messungen in der Längsachse gebaut.



6.1.2 0ACS100A.90-1

6.1.2.1 Bestelldaten

Bestellnummer	Kurzbeschreibung	Abbildung
	Sensoren	
0ACS100A.90-1	Beschleunigungssensor, nominale Empfindlichkeit 100 mV/g, Ausgang seitlich	
	Erforderliches Zubehör	
	Sensorkabel	
0ACC0020.01-1	Kabel für Beschleunigungssensor, Länge 2 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0050.01-1	Kabel für Beschleunigungssensor, Länge 5 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0100.01-1	Kabel für Beschleunigungssensor, Länge 10 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0150.01-1	Kabel für Beschleunigungssensor, Länge 15 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0200.01-1	Kabel für Beschleunigungssensor, Länge 20 m, 2x 0,34mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC0500.01-1	Kabel für Beschleunigungssensor, Länge 50 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	
0ACC1000.01-1	Kabel für Beschleunigungssensor, Länge 100 m, 2x 0,34 mm², M12 Buchse sensorseitig, schleppkettentauglich, UL zugelassen	

Tabelle 11: 0ACS100A.90-1 - Bestelldaten

6.1.2.2 Technische Daten

Bestellnummer	0ACS100A.90-1
Sensor Eigenschaften	
Eigenresonanz (montiert)	22 kHz (nominal)
Empfindlichkeit	100 mV/g $\pm 10\%$ nominal 80 Hz bei 22°C
Frequenzverhalten	2 Hz bis 10 kHz $\pm 5\%$ 0,8 Hz bis 15 kHz ± 3 dB
Isolation	Isolierte Basis
Messbereich	± 50 g
Querempfindlichkeit	<5%
Elektrische Eigenschaften	
Elektrische Störungen	max. 0,1 mg
Breitbandauflösung	0,2 mg (200 μ g) über 1 Hz bis 15 kHz
Spektralrauschen	10 Hz bis 10 μ g/Hz 100 Hz bis 4 μ g/Hz 1 kHz bis 3 μ g/Hz
Strombereich	0,5 bis 8 mA
Arbeitspunktspannung	10 bis 12 VDC
Einschwingzeit	2 s
Ausgangsimpedanz	max. 200 Ω
Gehäuseisolation	>10 ⁸ Ω bei 500 V
Einsatzbedingungen	
Schutzart nach EN 60529	IP67
Umgebungsbedingungen	
Temperatur	
Betrieb	-55 bis 140°C
max. Stoßfestigkeit	5000 g
Störaussendungen	EN61000-6-4:2001
Störfestigkeit	EN61000-6-2:1999
Mechanische Eigenschaften	
Gehäuse	
Material	Edelstahl
Montage	M8 x 1,25 x 33 mm Schraube, im Lieferumfang enthalten
Gewicht	170 g
Messelement	PZT- Piezoelektronischer Kristall (Blei-Zirkonat-Titanat)
Messauführung	komprimiert
Anzugsmoment	8 Nm
Anschlusstecker	M12

Tabelle 12: 0ACS100A.90-1 - Technische Daten

Die für den Sensor gültigen Zertifizierungen sind auf der Homepage des Herstellers zu finden.



Zertifikate

<http://www.hansfordsensors.com/resources/certificates/>

6.1.2.3 Abmessungen

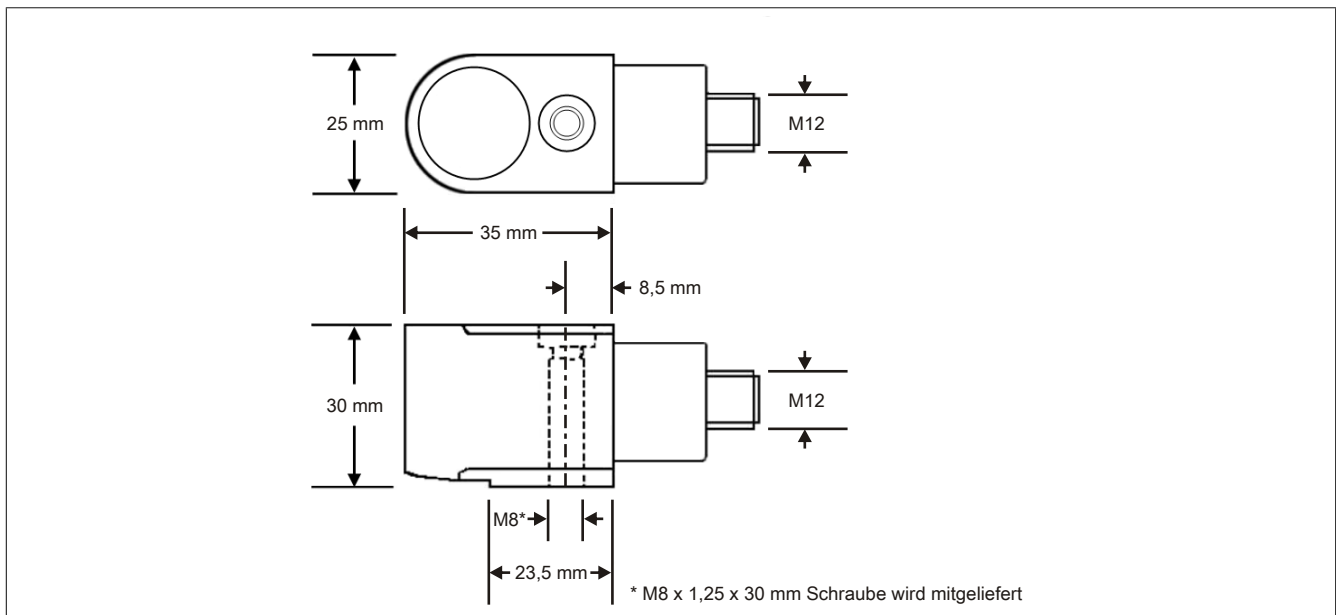
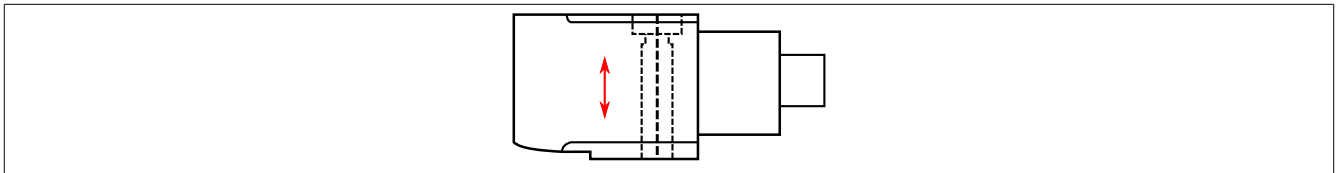


Abbildung 88: 0ACS100A.90-1 - Abmessungen

6.1.2.4 Einbaurichtung

Der B&R Schwingungssensor 0ACS100A.90-1 ist für Messungen in der Querachse gebaut.



6.1.3 Allgemeines

6.1.3.1 Steckerbelegung

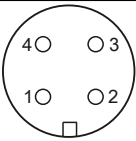
	Pin	Beschreibung
	1	Nicht belegt
	2	18 bis 30 V (braun)
	3	Nicht belegt
	4	0 V (blau)

Tabelle 13: 0ACS100A.x0-1 - Steckerbelegung

6.1.3.2 Frequenzverhalten

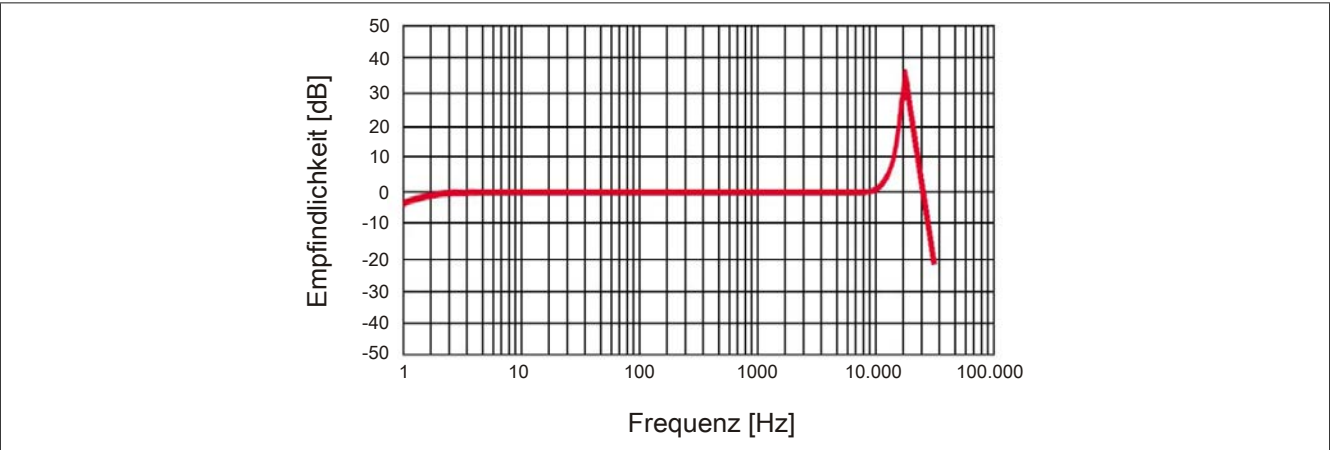


Abbildung 89: 0ACS100A.x0-1 - Frequenzverhalten

6.2 Sensorkabel

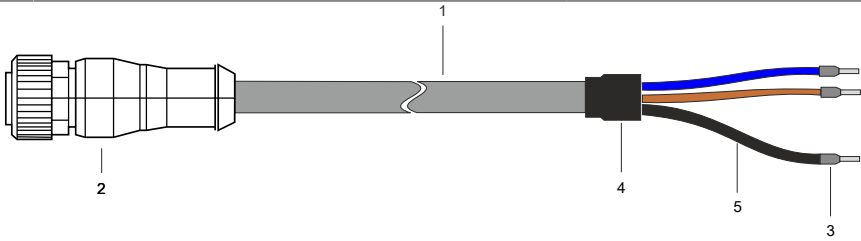
6.2.1 Bestelldaten

Bestellnummer	Länge	Kurzbeschreibung
0ACC0020.01-1	2 m	Kabel für Beschleunigungssensor, 2x 0,34 mm ² , 1x 0,25 mm ² , M12 Buchse sensorseitig, 1x 25mm ² Schirmanbindung, schleppkettentauglich, UL/CSA zugelassen
0ACC0050.01-1	5 m	
0ACC0100.01-1	10 m	
0ACC0150.01-1	15 m	
0ACC0200.01-1	20 m	
0ACC0500.01-1	50 m	
0ACC1000.01-1	100 m	

6.2.2 Technische Daten

Bezeichnung	0ACC0xx0.01-1
Allgemeines	
Polzahl	3
Kabellänge	x
Kennwerte Leitung	
Kabeltyp	PUR halogenfrei schwarz geschirmt
AWG Signalleitung	22
Leiteraufbau Signalleitung	42x 0,10 mm
Aderdurchmesser inkl. Isolierung	1,27 mm ±0,02 mm
Wandstärke Isolierung	≥ 0,21 mm (Aderisolierung) ca. 1,1 mm (Außenmantel)
Kabelaußendurchmesser	5,9 mm ±0,15 mm
Isolationswiderstand	≥ 100 GΩ*km (bei 20 °C)
Leiterwiderstand	max. 58 Ω/km (bei 20 °C)
Schirmung	Geflecht aus Kupferdrähten
Kabelgewicht	44 kg/km
Kleinsten Biegeradius, fest verlegt	29,5 mm
Kleinsten Biegeradius, beweglich verlegt	59 mm
Anzahl der Biegezyklen	4000000
Biegeradius	59 mm
Verfahrweg	10 m
Verfahrgeschwindigkeit	3 m/s
Beschleunigung	10 m/s ²
Umgebungstemperatur (Betrieb)	-40 °C ... 80 °C (Kabel, feste Verlegung) -25 °C ... 80 °C (Kabel, bewegliche Verlegung)
Schutzart nach EN 60529	IP67

6.2.3 Sensorkabel mit Stecker M12 Buchse

			
Pos.	Stück	Bezeichnung	Anmerkung
1	1	Sensorleitung	2x 0,34 mm ² (1501702 3x 0,34)
2	1	M12 Buchse (axial)	M12 Buchse (M12x1 A Codiert)
3	2	Aderendhülse (2x Sensorleitung)	3203066 AI 0,34-8 TQ
4	1	Schrumpfschlauch	
5	1	Schirmanbindung	1x 0,25 mm ² schwarz
6	1	Aderendhülse (1x Schirmanbindung)	3200632 AI 0,25-12 BU

6.2.4 Kabelplan

