

### INHALT

<b>Einleitung</b>	<b>1</b>	<b>Applikationsbeispiel 2: BiSS-Diagnosebyte zur Übertragung von Kommutierungsinformation</b>	<b>5</b>
<b>BiSS-Datenübertragung mit Diagnosebyte</b>	<b>2</b>		
<b>Erweiterte Fehler- und Warn-Meldungen</b>	<b>3</b>	<b>Applikationsbeispiel 3: BiSS-Diagnosebyte für eigensicheren Nonius-Absolutwertgeber</b>	<b>7</b>
<b>Applikationsbeispiel 1: BiSS-Diagnosebyte für Inkrementalsysteme mit Indexmarke(n)</b>	<b>5</b>	<b>REVISION HISTORY</b>	<b>8</b>

### Einleitung

Für absolute Positionsgeber ist die serielle *BiSS*-Schnittstelle ein inzwischen verbreiteter Standard zur schnellen und sicheren Übertragung absoluter Messdaten. Dank einer Laufzeitkompensation im Master sind hohe Taktfrequenzen möglich, dank einer CRC-gesicherten Übertragung werden Bitfehler aufgedeckt. Von Vorteil ist ebenfalls die permanent mögliche Registerkommunikation, die zur Parametrierung der Geräte und später zum Austausch von Diagnosedaten im Betrieb dient.

Die meisten Anwendungen erfordern nur eine Punkt-zu-Punkt Verbindung, so dass dann die prinzipiell gegebene Busfähigkeit der *BiSS*-Schnittstelle innerhalb eines Messgerätes bzw. Drehgebers genutzt werden kann, beispielsweise um mehrere redundante Sensor-Ebenen auszuwerten. Dabei müsste, von der Master-Seite her betrachtet, jede redundante Sensor-Ebene als eigene Datenquelle erscheinen und ausgewertet werden.

Bei Verwendung des *BiSS*-Diagnosebytes kann die Zykluszeit zur Datenübertragung für eine oder mehrere redundante Sensor-Ebenen deutlich verkürzt werden. Dafür prüft das Messgerät selbständig zyklisch oder auf Anforderung seine Messdaten auf Plausibilität unter Einbezug der redundant vorhandenen Sensor-Ebenen.

Ferner informiert das *BiSS*-Diagnosebyte die Steuerung über die vorliegende *Fehlerklasse in normierter Kodierung*, so dass die Steuerung auf eine Verifikation per Registerkommunikation verzichten und in kürzerer Zeit geeignete Maßnahmen einleiten kann. Oder die Steuerung hält die Applikation verfügbar, wenn ein Fehler in der speziellen Applikation als nicht sicherheitskritisch eingestuft werden kann.

Diese *BiSS* Application Note beschreibt die Funktion des *BiSS*-Diagnosebytes und stellt verschiedene Nutzungsmöglichkeiten vor.

### BiSS-Datenübertragung mit Diagnosebyte

Moderne Positionsgeber konditionieren und interpolieren ihre Messwerte und kommen daher nicht mehr ohne eine Programmierschnittstelle aus. Hier kommt die "Open-Source" *BiSS*-Schnittstelle zum Einsatz und ist weltweiter Industrie-Standard auf vielen Positions-Encodern und Interpolations-ICs verfügbar.

Das vom *BiSS*-Protokoll generell vorgesehene Error-Bit (nE) und Warning-Bit (nW) informiert die übergeordnete Steuerung über kritische und nicht-kritische Betriebszustände, die für jedes Gerät unterschiedlich definiert sein können. Ebenso unterscheiden sich die per Steuerungskommunikation (Control Data CDM, CDS) abzufragenden Adressen, unter denen Fehlermeldungen verifiziert werden müssen, um geeignete Maßnahmen einzuleiten.

Das *BiSS*-Diagnosebyte verfolgt einen alternativen und normierten Ansatz mit einer *festgelegten Codierung der Fehlerklasse* (für alle Fehler- und Warn-Meldungen) sowie mit einem festgelegten Mechanismus zur Überprüfung der Diagnosefähigkeit im Gesamtsystem.

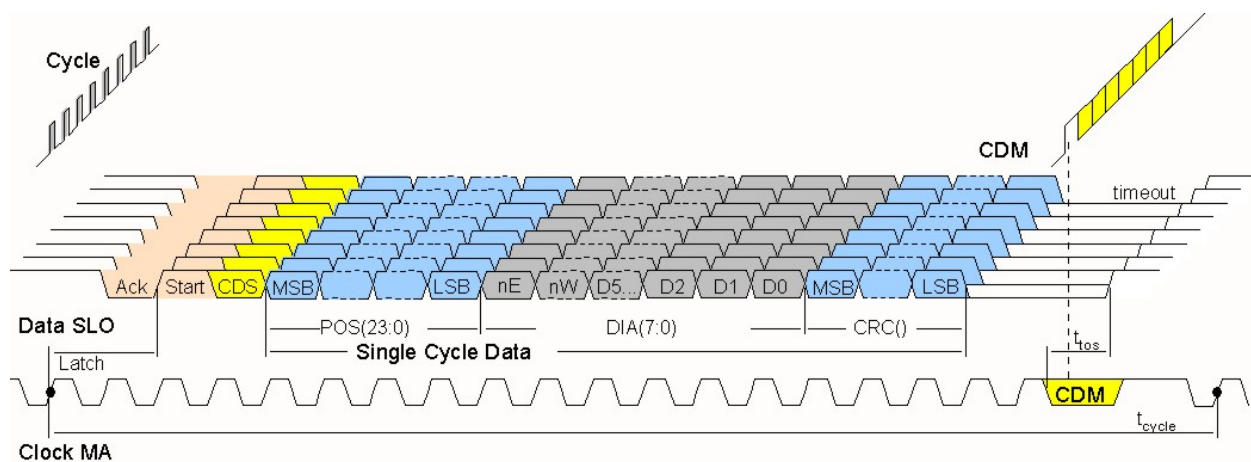


Bild 1: *BiSS*-Übertragung eines Positionswortes POS(23:0) mit zusätzlicher Diagnoseinformation DIA(7:0), beispielsweise zur Anzeige der Datengültigkeit nach systeminterner Prüfung.

### Erweiterte Fehler- und Warn-Meldungen

Dazu werden die vorhandenen nE und nW Bits um 6 Bits ergänzt: 2 Bits zur Ablaufsteuerung und 4 Bits informieren über den System-Status, über die *Fehlerklasse* (bei Fehler oder Warnung) oder beinhalten einen Test-Code. Für jede *Fehlerklasse* kann der übergeordneten Steuerung (ggf. auch auf höherer Ebene) eine vorzunehmende Maßnahme vorgeschlagen werden - der zu sendende Fehler-Code orientiert sich an der sinnvollen Behandlung.

Die Bits zur Ablaufsteuerung sind das REVOKE-Bit (RV) und das BREATH-Bit (BR).

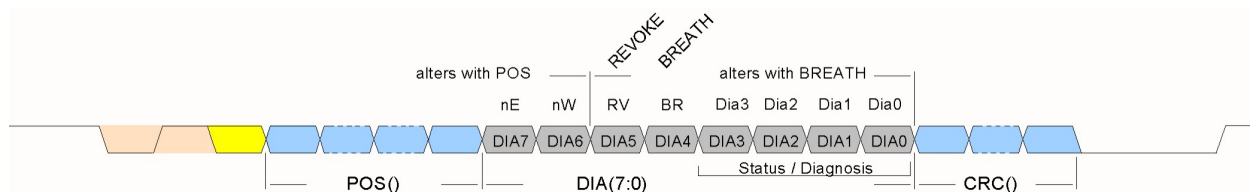


Bild 2: Diagnoseinformation im Detail. Das Error-Bit (nE) und das Warning-Bit (nW) sind zur Positionsabfrage synchron; ein Update der restlichen Diagnosebits, Dia(3:0), wird durch einen Wechsel des Breath-Bits (BR) angezeigt. Für Wirksamkeitstests markiert das Revoke-Bit (RV) die komplette Diagnoseinformation als ungültig.

Das REVOKE-Bit (RV) kann das Diagnosebyte als ungültig markieren und damit ein aktives nE oder nW (low aktiv), den gesendeten Code oder eine Statusinformation widerrufen. Dadurch können alle Meldungen und Codes im laufenden Betrieb durchgespielt werden, ohne dass die übergeordnete Steuerung Maßnahmen veranlasst.

Im Besonderen besteht für sicherheitsorientierte Anwendungen dadurch die Möglichkeit, dass ein Messsystem selbstständig sicherheitsrelevante Fehlermeldungen zyklisch einstreut, die von der übergeordneten Steuerung (ggf. in höherer Ebene) als Nachweis einer funktionierenden Diagnosefunktion erwartet werden.

Das BREATH-Bit (BR) ist der Atem des Diagnosebytes, ein Lebenszeichen, das alterniert, sobald ein neuer Code vorliegt. Wenn beispielsweise die Statusinformation mit jedem Single-Cycle-Zyklus aktualisiert wird, alterniert auch das Breath-Bit in jedem Zyklus. Liegt ein Fehler (nE) vor, alterniert das Breath-Bit, sobald der Fehler-Code aktualisiert wurde. Im Besonderen gibt das Breath-Bit einem im System befindlichen Mikrocontroller die erforderliche Beschaffungszeit, beispielsweise wenn komplexere Diagnosen ausgeführt werden müssen (z.B. zur Überprüfung einer Konfiguration im RAM gegen ein EEPROM).

Liegt eine Warnung (nW) vor, alterniert das Breath-Bit, sobald der Warnungs-Code aktualisiert wurde.

Ein Fehler (nE) ist immer dominant und es wird vorrangig der Fehler-Code gesendet, auch wenn eine Warnung (nW) gleichzeitig vorliegt.

Die Diagnose-Information ist grundsätzlich durch ein BiSS-Kommando rücksetzbar (über Control Data CD), und zwar im Normalbetrieb die Statusinformation, bei einem Fehler oder einer Warnung die *Fehlerklasse*. Daraufhin erneuert die MCU die Diagnose-Information.

Bit	Name	Verwendung
nE	Error Bit	Fehlermeldung (low aktiv): relevanter Systemfehler, Maßnahmen erforderlich
nW	Warning Bit	Warnmeldung (low aktiv): Betriebsinformation, nicht sicherheitskritisch, keine unmittelbaren Maßnahmen erforderlich
RV	Revoke Bit	Widerrufungsbit: markiert die Diagnoseinformation als ungültig (für Wirksamkeitstest und Fehlersimulation)
BR	Breath Bit	Atem-Bit: Lebenszeichen für Diagnoseinformation, wechselt um ein Update von Dia(3:0) anzuzeigen
Dia(3:0)	Statusinformation oder Fehlercode Warnungscode Testcode	4 Flags für Systemmeldungen oder GPIO-Portzustände (für nE= 1, nW= 1)  bei aktivem Error-Bit (nE= 0) bei aktivem Warn-Bit (nW= 0 und nE= 1) bei Widerruf durch Revoke (RV= 1)

Tab. 1: Bedeutung festgelegter Diagnosebits

nE	nW	RV	BR	DIA3	DIA2	DIA1	DIA0	Zustand
1	1	0	0→1 1→0	Dia(3:0) = Status Flags Code Update				Normalbetrieb Wechsel von BR zeigt Update an
0	d	0	0→1 1→0	Dia(3:0) = Error Code Code Update				Fehlermeldung Wechsel von BR zeigt Update an
1	0	0	0→1 1→0	Dia(3:0) = Warning Code Code Update				Warnmeldung Wechsel von BR zeigt Update an
d	d	1	d	Dia(3:0) = ungültig				Widerruf
Dia7	Dia6	1	d	Dia(3:0) = Test Code				Wirksamkeitstest
			0→1 1→0 0→1 1→0	Test Code 1 Test Code 2				Wechsel von BR aktiviert Test

Tab. 2: Verwendung des Diagnosebytes

Dia(3:0)	Bedeutung	Maßnahme
0x0	reserviert - keine Kodierung vergeben	
0xF	reserviert - keine Kodierung vergeben	
0x1	Konfigurationsfehler	Neustart, ggf. Austausch
tbd.	Datenfehler	Messwert verwerfen

Tab. 3: Definierte Fehlerklassen (für Fehler und Warnung)

### Applikationsbeispiel 1: BiSS-Diagnosebyte für Inkrementalsysteme mit Indexmarke(n)

Die vorhandene *BiSS*-Schnittstelle wird beispielsweise zur Übertragung inkrementeller Messdaten verwendet, um alternativ zur Quadraturchnittstelle die Performance zu erhöhen und gleichzeitig Kosten zu senken: ein Quadraturgeber mit ABZ-Ausgang benötigt 6 Adern für die differenzielle RS422-Übertragung, eine *BiSS*-Schnittstelle nur 4 (ebenfalls RS422).

Dabei kann die Erfassung von Indexmarken über das *BiSS*-Diagnosebyte angezeigt werden, um der Steuerung beispielsweise die Auswertung abstandskodierter Linearsysteme zu ermöglichen. Auch für pseudo-absolute Drehgeber, die durch Bewegung über eine oder mehrere Indexmarken referenzieren, kann das *BiSS*-Diagnosebyte die Nullsetzung bzw. die Absolutwert-Einstellung anzeigen.

Dia(3)	Dia(2)	Dia(1)	Dia(0)	Meldung
-	-	M2	M1	1. Marke erkannt (Periodenzähler gespeichert) 2. Marke erkannt (Periodenzähler gespeichert)

Tab. 4: Status für Inkrementalsystem mit Indexmarke(n)

nE	nW	RV	BR	Dia(3)	Dia(2)	Dia(1)	Dia(0)	Zustand
0	1	0		tbd				System nicht referenziert

Tab. 5: Warnung für Inkrementalsystem mit Indexmarke(n)

### Applikationsbeispiel 2: BiSS-Diagnosebyte zur Übertragung von Kommutierungsinformation

Besonders preissensitive Applikationen, wie beispielsweise Geber zur Geschwindigkeitsregelung von kollektorlosen Motoren, erfordern eine Lösung für die Übertragung der Kommutierungsinformation (UVW-Signale). Quadraturgeber mit drei zusätzlichen Leitungspaaren bedeuten jedoch höhere Kosten, die man auch durch Tricks einzusparen versucht: zum Anlauf werden zunächst die UVW-Kommutierungssignale übertragen und nach dem Index wird auf die ABZ-Inkrementalsignale umgeschaltet. Zur Auswertung kommt eine besondere Steuerungslogik zum Einsatz als proprietäre und nicht standardisierte Lösung.

Die *BiSS*-Schnittstelle bietet die elegantere Lösungsmöglichkeit, indem die UVW-Kommutierungssignale als zweites absolutes Datenwort betrachtet und mitgeschickt werden als Statusinformation im Diagnosebyte. Dazu lesen 3 GPIO-Ports die UVW-Signale vom Sensor und bestimmen die Status-Flags im Diagnosebyte (Bild 3).

# BiSS Interface

AN24: BiSS DATENÜBERTRAGUNG UND DIAGNOSE



Ausgabe A1, Seite 6/8

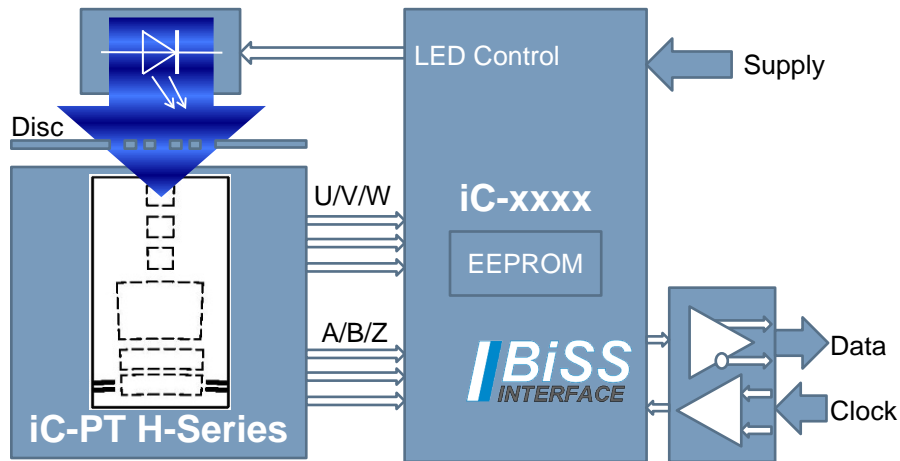


Bild 3: Systemkonzept für einen Motor-Encoder mit 6-Kanal Phased-Array-Abtastung und einer Interpolationschaltung mit *BiSS*-Schnittstelle.

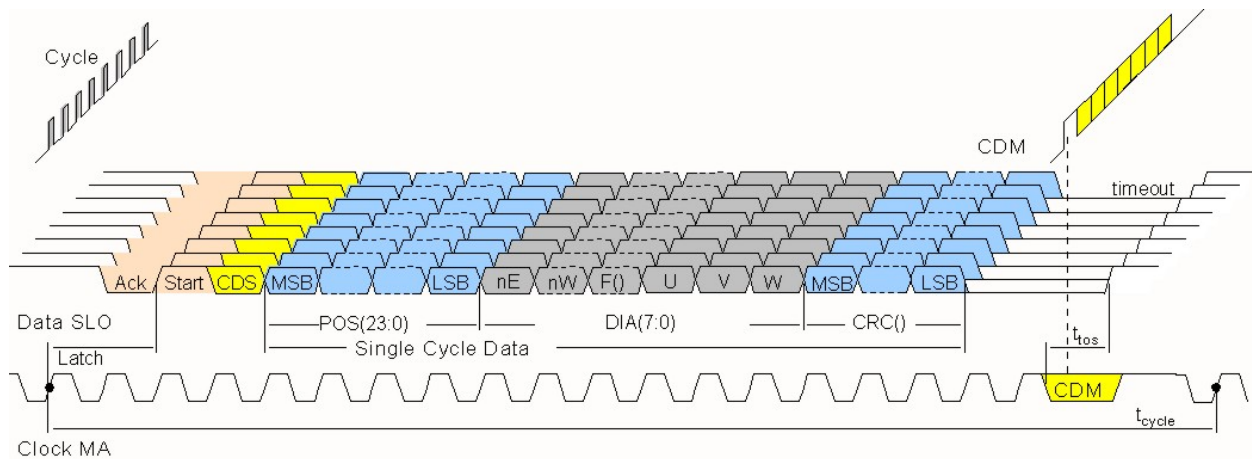


Bild 4: *BiSS*-Übertragung eines Positionswortes POS(23:0) mit zusätzlicher Diagnoseinformation DIA(7:0), beispielsweise zur Indexierung (F1) und Rotorlage (U, V, W).

Dia(3)	Dia(2)	Dia(1)	Dia(0)	Meldung
F1	U	V	W	Status der GPIO-Ports für UVW-Signale Flag 1: Indexierung erfolgt

Tab. 6: Status für Kommutierungsgeber

nE	nW	RV	BR	Dia(3)	Dia(2)	Dia(1)	Dia(0)	Zustand
0	1	0		tbd				System nicht referenziert

Tab. 7: Warnung für Kommutierungsgeber

### Applikationsbeispiel 3:

#### BiSS-Diagnosebyte für eigensicheren Nonius-Absolutwertgeber

Ein Nonius-System verwendet i.d.R. keine digital kodierte Winkelinformation, sondern ermittelt den absoluten Winkel aus der Signalphase zwischen zwei oder mehreren Inkrementalspuren, die analog durch eine Interpolationsschaltung ausgewertet werden.

Zur Abtastung der nebeneinander liegenden und in der Pulszahl unterschiedlichen Inkrementalspuren bietet der verwendete Phased-Array-Sensor mehrere Sensorfelder, deren Signale voneinander unabhängig aufbereitet und separat ausgegeben werden - eine Anordnung, die sowohl im optischen als auch im elektrischen Signalpfad eine redundante Auslegung aufweist.

Hardware-Redundanz ist ebenfalls auf Seite des auswertenden Interpolations-ICs möglich (Bild 5).

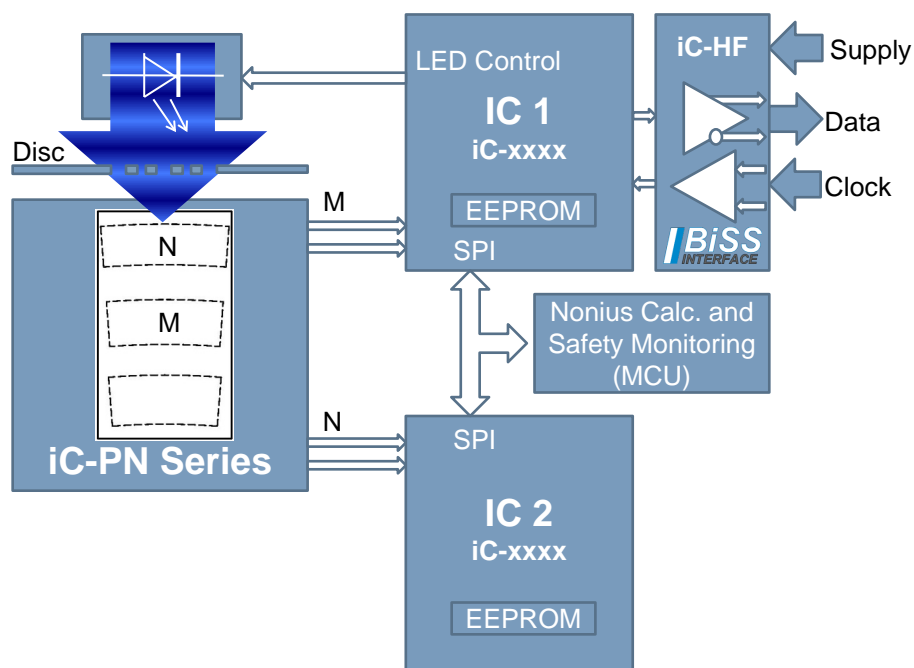


Bild 5: Systemkonzept für einen eigensicheren Absolutwert-Drehgeber mit *BiSS*-Schnittstelle. Der Mikrocontroller übernimmt die Nonius-Berechnung nach dem Einschalten und weiter die laufende Überwachung.

Nach dem Einschalten des Systems übernimmt der Mikrocontroller (MCU) die Nonius-Berechnung und setzt die Periodenzähler in beiden Interpolations-ICs IC1 und IC2 auf die errechnete absolute Startposition. Danach laufen beide Interpolations-Kanäle als Inkremental-Systeme unabhängig voneinander, wobei der Überlauf beider Periodenzähler auf die jeweils ausgewertete Linienzahl eingestellt ist. Beispielsweise nullt IC1 nach 64 Perioden / Umdrehung und IC2 nach 63 Perioden / Umdrehung.

Die MCU übernimmt die Überwachungsfunktionen und füllt das *BiSS*-Diagnosebyte im ausgehenden Baustein (IC 1); ferner dient die MCU als Host für *BiSS*-Registerabfragen außerhalb des Adressbereichs von IC1, beispielsweise für ein EDS und OEM-Daten.

Durch Verwendung des *BiSS*-Diagnosebytes können mit jeder Datenübertragung Informationen zum aktuellen Systemstatus mitgesendet werden, beispielsweise zur Datensicherheit.

Dafür prüft das System selbständig zyklisch oder auf Anforderung seine Messdaten auf Plausibilität unter Einbezug der redundant vorhandenen Sensor-Ebene (hier N-Kanal über IC2). Das Konzept ist auf eine 3. Sensor-Ebene (S-Kanal) erweiterbar.

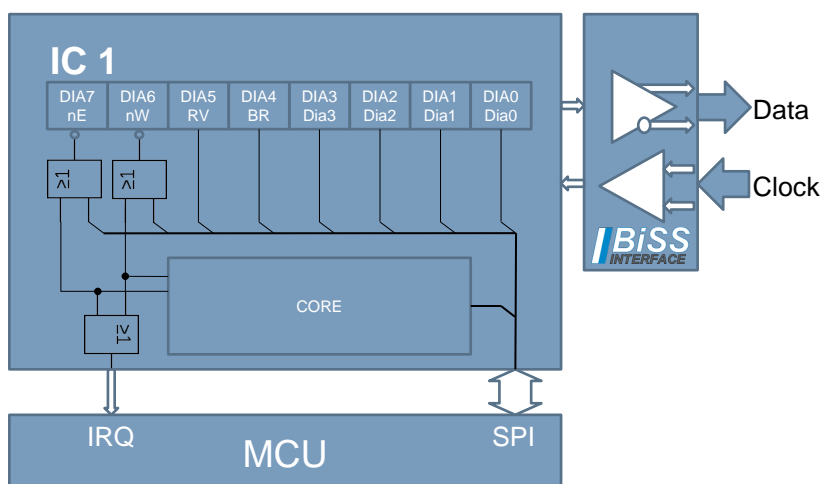


Bild 6: Konzept zur Ansteuerung des Diagnosebytes über eine MCU. Eine Fehler- oder Warnmeldung der IC1-Kernschaltung (CORE) bewirkt einen Interrupt, worauf die MCU die Fehler- oder Warn-Meldungen des COREs auswertet und die Fehlerklasse Code auf Dia3...0 einstellt.

Dia(3)	Dia(2)	Dia(1)	Dia(0)	Meldung
-	-	DV2	DV1	Daten CH1 gültig Daten CH2 = CH1

Tab. 8: Status (Nonius-Absolutwertgeber)

nE	nW	RV	BR	Dia(3)	Dia(2)	Dia(1)	Dia(0)	Zustand
0	1	0		tbd				DV2 = 0 (Noniuskontrolle)
1	0	0		tbd				DV2 = 0 (Noniuskontrolle) NOK

Tab. 9: Fehler und Warnung (Nonius-Absolutwertgeber)

### REVISION HISTORY

Rel.	Rel. Date	Chapter	Modification	Page
A1	2015-06-15		Erstes Release	all