

## Betriebsanleitung für:

**Px-x-xxxxM-Pxx-1-C207**

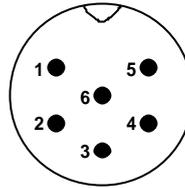
**Px-x-xxxxM-D6x-1-C207**

**Rx-x-xxxxM-Pxx-1-C207**

**Rx-x-xxxxM-D6x-1-C207**

## **Anschluß des Wegsensors:**

- 1 – grau: CAN\_L (dominant low)
- 2 – rosa: CAN\_H (dominant high)
- 5 – braun: +24V DC
- 6 – weis: 0V



## **Liste der Kommandowörter, Datenlängen und Datenformate**

Parameter	Funktion	COB-Id	DLC	Kommando/Daten
Knotenidentifizier	abfragen	2021 (7E5)	5	01, SS, SS, SS, SS
	programmieren	2021 (7E5)	6	02, SS, SS, SS, SS, XX
Positionsidentifizier	abfragen	2026 (7EA)	2	NId, 01
	programmieren	2026 (7EA)	4	NId, 02, XX, XX
Statusidentifizier	abfragen	2026 (7EA)	2	NId, 03
	programmieren	2026 (7EA)	4	NId, 04, XX, XX
Anzahl der Magnete	abfragen	2026 (7EA)	2	NId, 05
	programmieren	2026 (7EA)	3	NId, 06, XX

SS, SS, SS, SS - Seriennummer;

NId - Knotenidentifizier;

XX – Nutzdaten

## Erklärung zur Programmierung

Der Sensor arbeitet generell als Slave. Nach jedem Programmierbefehl mit dem der Sensor neue Parameter bekommt, antwortet er mit einer Quittierungsnachricht mit deren Hilfe die Steuerung erkennen kann, ob der Sensor die Daten richtig bekommen hat.

Im Normalbetrieb werden die Sensoren nach dem Einschalten der Versorgungsspannung nicht erneut parametrieren, da die Parameterdaten in einem EEPROM hinterlegt sind.

Die Positions- und Statusinformationen müssen von der Steuerung mittels eines *Remoteframes* auf den Positionsidentifizier bzw. Statusidentifizier abgeholt werden.

### Knotenidentifizier

Der Knotenidentifizier dient dem schnellen und einfachen Ansprechen der CAN-Bus Teilnehmer. Der Knotenidentifizier wird dem einzelnen CAN-Bus Teilnehmer während der Inbetriebnahme zugewiesen. Die Zuweisung erfolgt mittels der Seriennummer des Wegsensors (sie befindet sich auf dem Typenschild). Die Seriennummer muß wie folgt übertragen werden:

Seriennummer auf dem Typenschild: z.B. **FNr.:97020235**

Seriennummer für die Kommunikation: 97 02 02 35

#### *Knotenidentifizier abfragen*

Datenquelle	COB-ID	Daten	Datensenke
Steuerung	2021	01; SS; SS; SS; SS	Sensor
Sensor	2020	01; SS; SS; SS; SS; NId	Steuerung

#### *Knotenidentifizier programmieren*

Datenquelle	COB-ID	Daten	Datensenke
Steuerung	2021	02; SS; SS; SS; SS; NId	Sensor
Sensor	2020	02; SS; SS; SS; SS; NId	Steuerung

## Positionsidentifizier

Der Positionsidentifizier ist der Identifier mit dem der Wegsensor seine Positionsdaten auf den CAN-Bus sendet. Die Positionsdaten müssen mittels eines *Remote Frames* auf den Positionsidentifizier angefordert werden. Der Positionsidentifizier bestimmt die Priorität der Nachricht, wobei ein kleiner Positionsidentifizier eine höhere Priorität hat als ein großer Positionsidentifizier.

Nach einer einmaligen Anforderung der Positionswerte wird das komplette Positionspaket, bestehend aus mehreren Blöcken, übertragen.

### *Positionsidentifizier abfragen*

Datenquelle	COB-ID	Daten	Datensenke
Steuerung	2026	NId; 01	Sensor
Sensor	2025	NId; 01; XX; XX	Steuerung

### *Positionsidentifizier programmieren*

Datenquelle	COB-ID	Daten	Datensenke
Steuerung	2026	NId; 02; XX; XX	Sensor
Sensor	2025	NId; 02; XX; XX	Steuerung

## Statusidentifizier

Der Statusidentifizier ist der Identifizier mit dem der Sensor seine Statusinformation, wie z.B. falsche Magnetzahl oder Sensor defekt, sendet. Die Statusinformation muß mittels eines *Remote Frames* auf den Statusidentifizier angefordert werden. Der Statusidentifizier bestimmt die Priorität der Nachricht, wobei ein kleiner Statusidentifizier eine höhere Priorität hat als ein großer Statusidentifizier.

### *Statusidentifizier abfragen*

Datenquelle	COB-ID	Daten	Datensenke
Steuerung	2026	NId; 03	Sensor
Sensor	2025	NId; 03; XX; XX	Steuerung

### *Statusidentifizier programmieren*

Datenquelle	COB-ID	Daten	Datensenke
Steuerung	2026	NId; 04; XX; XX	Sensor
Sensor	2025	NId; 04; XX; XX	Steuerung

### **Anzahl der Magnete**

Es läßt sich die Anzahl der Magnete, die sich auf dem Sensor befinden sollen abfragen bzw. programmieren. Sollte die Anzahl der Magnete auf dem Sensor nicht identisch mit der programmierten Magneteanzahl sein, so wird dies in der Statusmeldung mit ausgegeben.

Je mehr Magnete programmiert sind, um so größer wird die zu übertragende Blocklänge werden.

#### *Anzahl der Magnete abfragen*

<b>Datenquelle</b>	<b>COB-ID</b>	<b>Daten</b>	<b>Datensenke</b>
Steuerung	2026	NId; 05	Sensor
Sensor	2025	NId; 05; XX	Steuerung

#### *Anzahl der Magnete programmieren*

<b>Datenquelle</b>	<b>COB-ID</b>	<b>Daten</b>	<b>Datensenke</b>
Steuerung	2026	NId; 06; XX	Sensor
Sensor	2025	NId; 06; XX	Steuerung

### Aufbau der Statusinformation

Die Statusinformation wird auf Anfrage mittels eines *Remoteframes* vom Sensor gesendet und enthält die Information ob die Anzahl der Magnete auf dem Sensor der programmierten Anzahl entspricht, bzw. ob mehr oder weniger Magnete auf dem Sensor sind. Außerdem wird der EEPROM Checksummenstatus und der Sensorstatus mit übertragen.

Funktion	Richtung	COB	DLC	Data
Statusmeldung	Se -> St	StatId	1	Statusbyte

Die Statusinformation ist wie folgt aufgebaut:

B7	B6	B5	B4	B3	B2	B1	B0
x	x	M+	M-	x	x	SE	SW

SW: Status Wegsensor	0 = Wegsensor in Ordnung 1 = Wegsensor defekt
SE: Status EEPROM	0 = Checksumme in Ordnung 1 = Checksumme fehlerhaft
M+ und M-: Magnetezahl	0 = Anzahl der Magnete auf dem Sensor ok 1 = zu wenig Magnete auf dem Sensor 2 = zu viel Magnete auf dem Sensor

### Aufbau der Weginformationen

Die Weginformationen befinden sich in einem Nachrichtenblock, bestehend aus x Einzel-nachrichten. Die Anzahl der Einzelnachrichten berechnet sich aus der programmierten Anzahl der Magnete auf dem Sensor. Die Blöcke werden gesendet als:

Block 1 von x, Block 2 von x, ... , Block x von x

Funktion	Richtung	COB	DLC	Data
Position Magnet 1 + 2	Se -> St	PosId	8	01,0F,M1,M1,M1,M2,M2,M2
Position Magnet 3 + 4	Se -> St	PosId	8	02,0F,Magnet 3, Magnet 4
Position Magnet 5 + 6	Se -> St	PosId	8	03,0F,Magnet 5, Magnet 6
Position Magnet 7 + 8	Se -> St	PosId	8	04,0F,Magnet 7, Magnet 8
:				
Position Magnet 29 + 30	Se -> St	PosId	8	0F,0F,Magnet 29, Magnet 30