

ifm electronic



Gerätehandbuch
Drehgeber
mit DeviceNet Schnittstelle

DE

efector 400[®]

RM7

RN7

08/2014

706373/00

Inhalt

1	Vorbemerkung	4
1.1	Verwendete Symbole	4
1.2	Verwendete Warnhinweise	4
2	Sicherheitshinweise	4
3	Bestimmungsgemäßes Verwenden	5
3.1	Drehgeber	5
3.2	Betriebsmodi	5
4	Installation	6
4.1	Elektrischer Anschluss	6
4.2	Einstellungen in der Anschlusshaube	7
4.3	Kabel	7
4.4	Stecker	7
5	Common Industrial Protocol (CIP)	8
5.1	Objektmodell	8
6	Datenübertragung	9
6.1	Objektverzeichnis	9
6.2	CAN-ID	10
7	Konfiguration	11
7.1	Betriebsparameter	11
7.2	Auflösung pro Umdrehung	11
7.3	Gesamtauflösung	12
7.4	Presetwert	13
7.5	MAC-ID	13
7.6	Baudrate	13
8	Betriebsarten	14
8.1	Polled Mode	14
8.1.1	Allocate Master / Slave Connection Set	14
8.1.2	Release Master / Slave Connection Set	15
8.2	Change of State Mode	15
8.2.1	Allocate Master / Slave Connection Set	15
8.2.2	Expected_packet_rate der Explicit Message	16
8.2.3	Expected_packet_rate der Change of State	16
8.2.4	Release Master / Slave Connection Set	16
8.3	Speicherübernahme	17
9	Übertragung des Prozess-Istwertes	17
10	Inbetriebnahme	18
10.1	Betriebsparameter	18
10.2	Auflösung pro Umdrehung	18
10.3	Gesamtauflösung	19
10.4	Presetwert	20

10.5 Baudrate.	21
10.6 MAC-ID.	21
10.7 Speicherübernahme.	22
11 Projektierung unter RsNetworx.	23
11.1 EDS-File installieren.	23
11.2 Treiber konfigurieren.	26
11.3 Netzwerk online schalten	27
11.4 Encoder konfigurieren	28
12 Begriffe und Abkürzungen	30

Lizenzen und Warenzeichen

Microsoft®, Windows®, Windows XP® und Windows Vista® sind eingetragene Warenzeichen der Microsoft Corporation.

Alle benutzten Warenzeichen und Firmenbezeichnungen unterliegen dem Copyright der jeweiligen Firmen.

Open Source Software

Dieses Gerät enthält (ggf. veränderte) Open Source Software, die besonderen Lizenzbestimmungen unterliegt.

Urheberrechtliche Hinweise und Lizenzbestimmungen unter: www.ifm.com/int/GNU

Bei Software, die der GNU General Public License bzw. der GNU Lesser General Public License unterfällt, kann der Quelltext gegen Übernahme der Kopier- und Versandgebühren angefordert werden.

1 Vorbemerkung

1.1 Verwendete Symbole

► Handlungsanweisung

> Reaktion, Ergebnis

[...] Bezeichnung von Tasten, Schaltflächen oder Anzeigen

→ Querverweis



Wichtiger Hinweis

Fehlfunktionen oder Störungen sind bei Nichtbeachtung möglich



Information

Ergänzender Hinweis

1.2 Verwendete Warnhinweise

ACHTUNG

Warnung vor Sachschäden

2 Sicherheitshinweise

Diese Anleitung ist Bestandteil des Gerätes. Sie enthält Texte und Abbildungen zum korrekten Umgang mit dem Gerät und muss vor einer Installation oder dem Einsatz gelesen werden.

Befolgen Sie die Angaben dieser Anleitung.

Nichtbeachten der Hinweise, Verwendung außerhalb der nachstehend genannten bestimmungsgemäßen Verwendung, falsche Installation oder Handhabung können Beeinträchtigungen der Sicherheit von Menschen und Anlagen zur Folge haben.

Der Einbau und Anschluss muss den gültigen nationalen und internationalen Normen entsprechen. Die Verantwortung trägt derjenige, der das Gerät installiert.

An den Anschlüssen dürfen nur die in den technischen Daten, bzw. auf dem Geräteaufdruck angegebenen Signale eingespeist werden.

3 Bestimmungsgemäße Verwendung

3.1 Drehgeber

Drehgeber liefern für jede Winkelstellung eine Schrittweite. Die Werte sind als Codemuster auf eine oder mehrere Codescheiben abgebildet. Die Codescheiben werden mit einer Infrarot-LED durchleuchtet und das erhaltene Bitmuster wird durch ein Opto-Array detektiert. Die gewonnenen Signale werden elektronisch verstärkt und zur Verarbeitung an das Interface weitergeleitet.

Der Drehgeber hat eine maximale Grundauflösung von 65536 Schritten pro Umdrehung (16 Bit). In der Multiturn-Ausführung werden bis zu 16384 Umdrehungen (14 Bit) aufgelöst. Daraus ergibt sich eine Gesamtauflösung von maximal 30 Bit = 1.073.741.824 Schritten. Die Standard Singleturn-Ausführung hat 12 Bit, die Standard Multiturn-Ausführung 24 Bit.

3.2 Betriebsmodi

Der Drehgeber kennt die folgenden Betriebsarten:

- Polled Mode
- Change of State Mode

Über den integrierten CAN-Bus sind die folgenden Funktionen programmierbar:

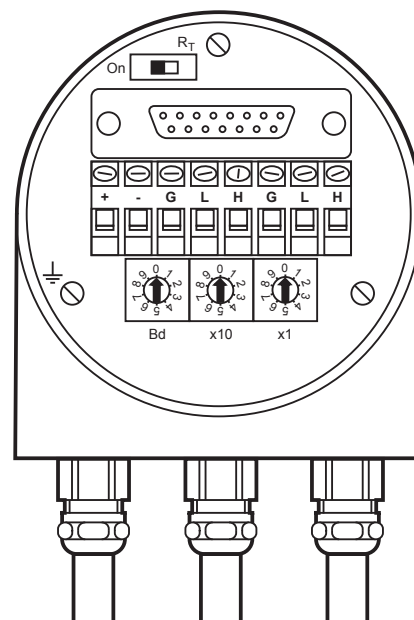
- Drehrichtung (Complement)
- Auflösung pro Umdrehung
- Gesamtauflösung
- Presetwert
- Baudrate
- MAC-ID

4 Installation

4.1 Elektrischer Anschluss

Der Winkelcodierer wird über drei Kabel angeschlossen. Die Spannungsversorgung erfolgt über ein zweiadriges Verbindungskabel durch eine PG 9. Die jeweils zweiadrig abgeschirmte Busleitung wird in bzw. aus dem Winkelcodierer über je eine PG 9 hinein- bzw. herausgeführt:

Klemme	Beschreibung
⊥	Masse
+	24 V Spannungsversorgung
-	0 V Spannungsversorgung
G	CAN Ground
L	CAN Low
H	CAN High
G	CAN Ground
L	CAN Low
H	CAN High



In der Anschlusshaube ist ein Widerstand vorgesehen, der bei Bedarf als Leitungs-Abschluss zugeschaltet werden kann. Abschlusswiderstand:

Letzter Teilnehmer



Teilnehmer X



Die Knotennummer wird über 2 Drehschalter in der Anschlusshaube eingestellt. Mögliche Adressen liegen zwischen 0 und 63, wobei jede nur einmal vorkommen darf. Die Anschlusshaube kann einfach vom Anwender durch Lösen von zwei Schrauben am Winkelcodierer zur Installation abgenommen werden. Zwei Diagnose-LED auf der Rückseite der Anschlusshaube zeigen den Betriebszustand des Drehgebers an.

DeviceNet Geräte BCD-Drehschalter	
x1	Geräteadresse 0..63
x10	CAN-Knotennummer
xBd	Baudrate

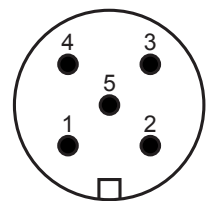
4.2 Einstellungen in der Anschlusshaube

Baudrate in kBit/s	BCD-Drehschalter
125	0
250	1
500	2
125	3
reserved	4...9

4.3 Kabel

Pin	Signal	Beschreibung	Farbkennzeichnung
1	V-	GND	Schwarz
2	CAN-L	CAN Bus Signal (dominant low)	Blau
3	CAN-H	CAN Bus Signal (dominant high)	Weiß
4	V+	Externe Spannungsversorgung Vcc	Rot

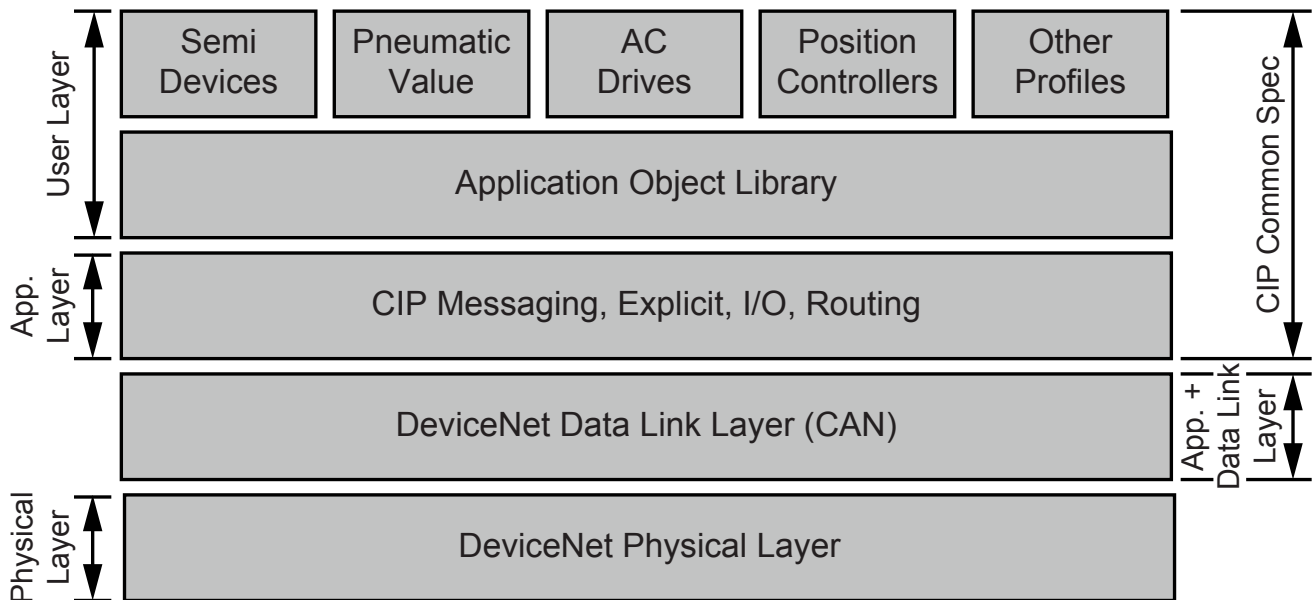
4.4 Stecker



5-poliger Stecker

Pin	Signal	Beschreibung	Farbkennzeichnung
2	V+	Externe Spannungsversorgung Vcc	Rot
3	V-	GND	Schwarz
4	CAN-H	CAN Bus Signal (dominant high)	Weiß
5	CAN-L	CAN Bus Signal (dominant low)	Blau

5 Common Industrial Protocol (CIP)



Die DeviceNet-Spezifikation definiert den Application Layer und den Physical Layer. Der Data Link Layer basiert auf der CAN-Spezifikation. Für die optimale Industriesteuerung werden zwei verschiedene Nachrichtentypen zur Verfügung gestellt. I/O Nachrichten (Implicit Messaging) und explizite Nachrichten (Explicit Messaging). Mit Implicit Messaging werden I/O Daten in Echtzeit ausgetauscht und mit Explicit Messaging werden Daten zur Konfiguration eines Gerätes ausgetauscht. CIP (Common Industrial Protocol) stellt dem Anwender vier Funktionalitäten zur Verfügung:

- Einheitliche Steuerungsdienste
- Einheitliche Kommunikationsdienste
- Einheitliche Verteilung von Nachrichten
- Gemeinsame Wissensbasis

5.1 Objektmodell

DeviceNet beschreibt alle Daten und Funktionen eines Gerätes anhand eines Objektmodells. Mit dieser objektorientierten Beschreibung kann ein Gerät mit einzelnen Objekten vollständig definiert werden.

Definition eines Objektes:

- Zusammenfassung von zugehörigen Attributen (z.B. Prozessdaten),
- nach außen bereitgestellte Funktionen (Lese- oder Schreibzugriff auf ein einzelnes Attribut),
- definiertes Verhalten.

DeviceNet unterscheidet zwischen drei Arten von Objekten:

- **Kommunikationsobjekte** definieren die über DeviceNet ausgetauschten Nachrichten und werden als Connection Objects bezeichnet (DeviceNet Object, Message Router Object, Connection Object, Acknowledge Handler Object).
- **Systemobjekte** definieren allgemeine DeviceNet-spezifische Daten und Funktionen (Identity Object, Parameter Object).
- **Applikationsspezifische Objekte** definieren gerätespezifische Daten und Funktionen (Application Object, Assembly Object).

6 Datenübertragung

Das Übertragen von Daten in DeviceNet erfolgt über Telegramme. Grundsätzlich lassen sich die Telegramme schematisch in CAN-ID und 8 Folgebytes aufteilen:

COB-ID	Message Header	Message Body
11 Bit	1 Byte	7 Byte

6.1 Objektverzeichnis

Das Objektverzeichnis beschreibt Instanz Attribute des Position Sensor Objekts.

Class Code: 23 hex

Attribut ID	Zugriff	Name	Datenlänge	Beschreibung
1 hex	Lesen	Anzahl der Attribute	USINT	Anzahl der unterstützten Attribute
2 hex	Lesen	Attribute	Array of USINT	Liste unterstützter Attribute
3 hex	Lesen	Positionswert	DINT	Ausgabe der aktuellen Position
70 hex	Lesen / Schreiben	Drehrichtungskontrolle	Boolean	Steuert die Codefolge steigend / fallend
71 hex	Lesen / Schreiben	Auflösung pro Umdrehung	INT	Auflösung für eine Umdrehung
72 hex	Lesen / Schreiben	Gesamtauflösung	DINT	Gesamt eindeutig darstellbare Auflösung
73 hex	Lesen / Schreiben	Preset Wert	DINT	Zuordnung Positionswert
6E hex	Lesen / Schreiben	Baudrate		Einstellung der Baudrate
6F hex	Lesen / Schreiben	MAC ID		Einstellung der MAC ID

6.2 CAN-ID

DeviceNet basiert auf dem Standard-CAN Protokoll und verwendet einen 11 Bit (max. 2048 Nachrichten) Nachrichtenidentifizier. Zur Kennzeichnung eines Gerätes oder Knotens in einem DeviceNet-Netzwerk reichen 6 Bit aus, da ein Netzwerk auf 64 Teilnehmer begrenzt ist. Diese Kennzeichnung wird als MAC ID (Geräte- oder Knotenadresse) bezeichnet. Der CAN-Identifizier setzt sich aus der Kennung der Message Group, der Message ID innerhalb dieser Gruppe und der MAC-ID des Gerätes zusammen.

Bei dem Absoluten Drehgeber handelt es sich um einen Group 2-Server. Die folgende Tabelle listet die wichtigsten CAN-IDs für bestimmte Kommunikationsarten auf.

10	9	8	7	6	5	4	3	2	1	0	Identity Usage	Hex Range
0	Group 1 Message ID				Source MAC-ID						Group 1 Message	000-3ff
0	1	1	0	1	Source MAC-ID						Slave I/O Change of State or Cyclic Message	
0	1	1	1	1	Source MAC-ID						Slave I/O Poll Response or Change of State/Cyclic Acknowledge Message	
1	0	MAC-ID						Group 2 Message ID			Group 2 Messages	400-5ff
1	0	Destination MAC-ID						0	1	0	Master Change of State or Cyclic Acknowledge Message	
1	0	Source MAC-ID						0	1	1	Slave Explicit/Unconnected Response Messages	
1	0	Destination MAC-ID						1	0	0	Master Explicit Request Message	
1	0	Destination MAC-ID						1	0	1	Master I/O Poll Command/Change of State/Cyclic Message	
1	0	Destination MAC-ID						1	1	0	Group 2 Only Unconnected Explicit Request Message (reserved)	
1	0	Destination MAC-ID						1	1	1	Duplicate MAC-ID Check Messages	

7 Konfiguration

Im Folgenden wird das Konfigurieren des Drehgebers beschrieben.

7.1 Betriebsparameter

Als Betriebsparameter kann die Drehrichtung gewählt werden.

Attribut ID	Standardwert	Wertebereich	Datenlänge
0b hex	1 hex	0hex - 1hex	Boolean

DE

Der Parameter Drehrichtung (Complement) definiert die Zählrichtung der Ausgabe des Prozess-Istwertes bei Drehung der Welle im Uhrzeigersinn (CW) oder gegen den Uhrzeigersinn (CCW) bei Sicht auf Welle. Die Zählrichtung wird im Attribut 0bhex festgelegt:

Bit 0	Drehrichtung	Ausgabecode
1	CW	Steigend
0	CCW	Fallend

7.2 Auflösung pro Umdrehung

Der Parameter Auflösung pro Umdrehung wird dazu verwendet, den Encoder so zu programmieren, dass eine gewünschte Anzahl von Schritten bezogen auf eine Umdrehung realisiert werden kann.

Attribut ID	Standardwert	Wertebereich	Datenlänge
2C hex	siehe Typenschild	0hex - 2000hex	Unsigned Integer16

Maximale Auflösung bei

24 Bit Ausführung: 1.000 hex

25 Bit Ausführung: 2.000 hex

Wird als Auflösung pro Umdrehung ein Wert größer der Grundauflösung des Drehgebers gewählt, ist der Ausgabecode nicht mehr einschrittig. Es ist daher darauf zu achten, dass die gewünschte Auflösung die Auflösung des Drehgebers nicht übersteigt.

7.3 Gesamtauflösung

Die Gesamtauflösung gibt die gewünschte Anzahl der Messeinheiten der gesamten Verfahrenslänge an. Der Wert darf die Gesamtauflösung des Drehgebers nicht übersteigen. Die Gesamtauflösung des Drehgebers ist auf dem Typenschild angegeben.

Attribut ID	Standardwert	Wertebereich	Datenlänge
72 hex	siehe Typenschild	0h - 2.000.000h	Unsigned Integer 32

Maximale Auflösung bei

24 Bit Ausführung: 1.000.000 hex

25 Bit Ausführung: 2.000.000 hex

Folgende Formelbuchstaben werden nachfolgend verwendet:

- PGA Physikalische Gesamtauflösung des Encoders (siehe Typenschild)
- PAU Physikalische Auflösung pro Umdrehung (siehe Typenschild)
- GA Gesamtauflösung (Benutzereingabe)
- AU Auflösung pro Umdrehung (Benutzereingabe)

Wenn die gewünschte Auflösung pro Umdrehung kleiner ist als die tatsächliche physikalisch Auflösung des Encoders pro Umdrehung, dann muss die Gesamtauflösung wie folgt eingegeben werden:

Gesamtauflösung

$GA = PGA * AU / PAU$, wenn $AU < PAU$

Beispiel: Benutzervorgabe: $AU = 2048$,

Encoderwerte: $PGA = 24 \text{ Bit}$, $PAU = 12 \text{ Bit}$

$$GA = 16777216 * 2048 / 4096$$

$$GA = 8388608$$

Wird die Gesamtauflösung des Drehgebers kleiner als die physikalische Gesamtauflösung gewählt, so muss der Parameter Gesamtauflösung ein ganzzahliges Vielfaches der physikalischen Gesamtauflösung sein.

$$k = PGA / GA$$

k = ganze Zahl

7.4 Presetwert

Der Presetwert ist der gewünschte Positionswert, der bei einer bestimmten physikalischen Stellung der Achse erreicht sein soll. Über den Parameter Presetwert wird der Positions-Istwert auf den gewünschten Prozess-Istwert gesetzt. Der Presetwert darf den Parameter Gesamtauflösung nicht übersteigen.

Attribut ID	Standardwert	Wertebereich	Datenlänge
73 hex	0 hex	0hex - Gesamtauflösung	Unsigned Integer 32

DE

7.5 MAC-ID

Bei Absoluten Drehgeber ohne Anschlusshaube wird die MAC-ID mittels Explicit Messaging konfiguriert. Die DeviceNet Encoder adressieren 64 verschiedene Knoten.

Attribut ID	Standardwert	Wertebereich	Datenlänge
6F hex	(*)	0hex – 3Fhex	Unsigned Integer8

7.6 Baudrate

Bei Absoluten Drehgebern ohne Anschlusshaube wird die Baudrate mittels Explicit Messaging konfiguriert. Die DeviceNet Encoder unterstützen alle DeviceNet Baudraten der folgenden Tabelle.

Attribut ID	Standardwert	Wertebereich	Datenlänge
6E hex	(*)	0hex – 2hex	Unsigned Integer8

Byte	Baudrate
0	125kbaud
1	250kbaud
2	500kbaud

8 Betriebsarten

8.1 Polled Mode

Der Polled Mode ist eine Master-Slave Kommunikation. Der Master kann mit der Poll Command Message den aktuellen Positions-Istwert des Drehgebers abfragen. Der Drehgeber sendet anschließend den Prozess-Istwert mit einer Poll Response Message an den Master. Im Beispiel wird von einer Master MAC ID von 0A hex und einer Slave MAC ID von 03 hex ausgegangen.

8.1.1 Allocate Master / Slave Connection Set

Allocate Polling

Byte Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Frag [0]	XID	MAC ID					
1	R/R [0]	Service [4B]						
	Class ID [03]							
	Instance ID [01]							
	Allocation Choice [03]							
	0	0	Allocator MAC ID					

Definition CAN ID

10	9	8	7	6	5	4	3	2	1	0	Identity Usage	Hex Range
1	0	Destination MAC ID						1	1	0	Group 2 only unconnected explicit request message (reserved)	

Beispiel

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
41E	0A	4B	03	01	03	0A

Expected_packet_rate der Explicit Message Connection auf 0 setzen

Definition CAN ID

10	9	8	7	6	5	4	3	2	1	0	Identity Usage	Hex Range
1	0	Destination MAC ID						1	0	0	Master explicit request message	

Beispiel

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
41C	0A	10	05	01	09	00	00

Expected_packet_rate der Polling Connection auf 0 setzen

Beispiel

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
41C	0A	10	05	02	09	00	00

8.1.2 Release Master / Slave Connection Set

Release Polling

DE

Byte Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Frag [0]	XID	MAC ID					
1	R/R [0]	Service [4C]						
	Class ID [03]							
	Instance ID [01]							
	Release Choice [03]							

Beispiel

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
41E	0A	4C	03	01	03

8.2 Change of State Mode

Der Absolutwertgeber sendet ohne Aufforderung durch den Host, sobald sich der aktuelle Prozess-Istwert ändert. Solange der Prozesswert unverändert ist, wird nicht gesendet. Dadurch reduziert sich die Buslast.

8.2.1 Allocate Master / Slave Connection Set

Allocate COS

Byte Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Frag [0]	XID	MAC ID					
1	R/R [0]	Service [4B]						
	Class ID [03]							
	Instance ID [01]							
	Allocation Choice [51]							
	0	0	Allocator MAC ID					

Beispiel

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
41E	0A	4B	03	01	51	0A

8.2.2 Expected_packet_rate der Explicit Message

Beispiel für Connection auf 0 setzen

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
41C	0A	10	05	01	09	00	00

8.2.3 Expected_packet_rate der Change of State

Beispiel für Connection auf 0 setzen

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
41C	0A	10	05	04	09	00	00

8.2.4 Release Master / Slave Connection Set

Release COS

Byte Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Frag [0]	XID	MAC ID					
1	R/R [0]	Service [4C]						
	Class ID [03]							
	Instance ID [01]							
	Release Choice [51]							

Beispiel

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
41E	0A	4C	03	01	51

8.3 Speicherübernahme

Der Drehgeber ist mit einem Flash-EPROM ausgestattet, welches die enthaltenen Daten nullspannungssicher speichert. Da ein Flash-EPROM nach ca. 1000 Schreibzyklen seine Speicherfähigkeit verliert, werden geänderte Daten zuerst im Arbeitsspeicher gespeichert. Von dort werden die Daten nach einer Prüfung in das Flash-EPROM kopiert.

Wenn das Kopieren in das Flash-EPROM erfolgreich war, meldet sich der Encoder mit einem MAC ID check auf dem Bus. Um den Prozesswert abzufragen, muss der Slave erneut allocated werden.

DE

Byte Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0						
0	Frag [0]	XID	MAC ID											
1	R/R [0]	Service [32]												
	Class ID [23]													
	Instance ID [01]													

Beispiel (MAC ID Master: 0A hex, MAC ID Slave 03 hex)

CAN-ID	Byte 0	Byte 1	Byte 2	Byte 3
41C	0A	32	23	01

9 Übertragung des Prozess-Istwertes

Das folgende Telegramm beschreibt, wie der Prozess-Istwert übertragen wird.

CAN-ID	Prozess-Istwert			
11 Bit	Byte 0	Byte 1	Byte 2	Byte 3
	2^7 to 2^0	2^{15} to 2^8	2^{23} to 2^{16}	2^{31} to 2^{24}

10 Inbetriebnahme

Nach dem Einschalten der Spannungsversorgung meldet sich der Absolutwertgeber mit einem MAC ID-Check auf dem Bus.



Für die CAN ID und MAC ID wird für den Master 0A (hex) und Slave 03 (Hex) beispielhaft verwendet. Parameter die unverändert bleiben sollen, können übersprungen werden. Die Zahlenwerte werden hexadezimal angegeben.

10.1 Betriebsparameter

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Service Code	Class ID	Instance ID	Attribut ID	Data		
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
41C	0A	10	23	01	0B	X	-	-

X: 0 (hex) für CW (Standard)
1 (hex) für CCW

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID	Service Code
	Byte 0	Byte 1
41B	0A	90

10.2 Auflösung pro Umdrehung

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Service Code	Class ID	Instance ID	Attribut ID	Data		
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
41C	0A	10	23	01	2C	X	X	-

X: Auflösung pro Umdrehung

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID	Service Code
	Byte 0	Byte 1
41B	0A	90

10.3 Gesamtauflösung

Zur Übertragung der Gesamtauflösung muss eine fragmentierte Übertragung durchgeführt werden. Die folgenden Telegramme zeigen, wie die Gesamtauflösung übertragen wird.

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Fragment	Service Code	Class ID	Instance ID	Attribut ID		
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
41C	8A	00	10	23	01	2D	X	X

DE

X: Gesamtauflösung

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID		
	Byte 0	Byte 1	Byte 2
41B	8A	C0	00

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Fragment						
	Byte 0	Byte 1	Byte 2	Byte 3				
41C	8A	81	X	X				

X: Gesamtauflösung

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID		
	Byte 0	Byte 1	Byte 2
41B	8A	C1	00

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID	Service Code
	Byte 0	Byte 1
41B	0A	90

10.4 Presetwert

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Fragment	Service Code	Class ID	Instance ID	Attribut ID		
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
41C	8A	00	10	23	01	2E	X	X

X: Presetwert

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID		
	Byte 0	Byte 1	Byte 2
41B	8A	C0	00

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Fragment						
	Byte 0	Byte 1	Byte 6	Byte 7				
41C	8A	81	X	X				

X: Gesamtauflösung

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID		
	Byte 0	Byte 1	Byte 2
41B	8A	C1	00

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID	Service Code
	Byte 0	Byte 1
		90

10.5 Baudrate

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Service Code	Class ID	Instance ID	Attribut ID	Attribut ID		
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
41C	0A	10	23	01	6E	X	-	-

X: Baudrate

DE

X	Baudrate
0	125 kbaud
1	250 kbaud
2	500 kbaud

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID	Service Code
	Byte 0	Byte 1
41B	0A	90

10.6 MAC-ID

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Service Code	Class ID	Instance ID	Attribut ID	Attribut ID		
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
41C	0A	10	23	01	6F	X	-	-

X: MAC-ID

Absolutwertgeber an Master: Bestätigung

CAN ID	MAC ID	Service Code
	Byte 0	Byte 1
41B	0A	90

10.7 Speicherübernahme

Master an Absolutwertgeber: Set-Parameter

CAN ID	MAC ID	Service Code	Class ID	Instance ID
	Byte 0	Byte 1	Byte 2	Byte 3
41C	0A	32	23	01

Ist die Übertragung erfolgreich, meldet sich der Absolutwertgeber nach 2 s mit einer vervielfältigten MAC ID zurück. Der Master muss anschließend den Slave neu zuweisen.

Ist die Übertragung nicht erfolgreich, wird eine Fehlermeldung ausgegeben.

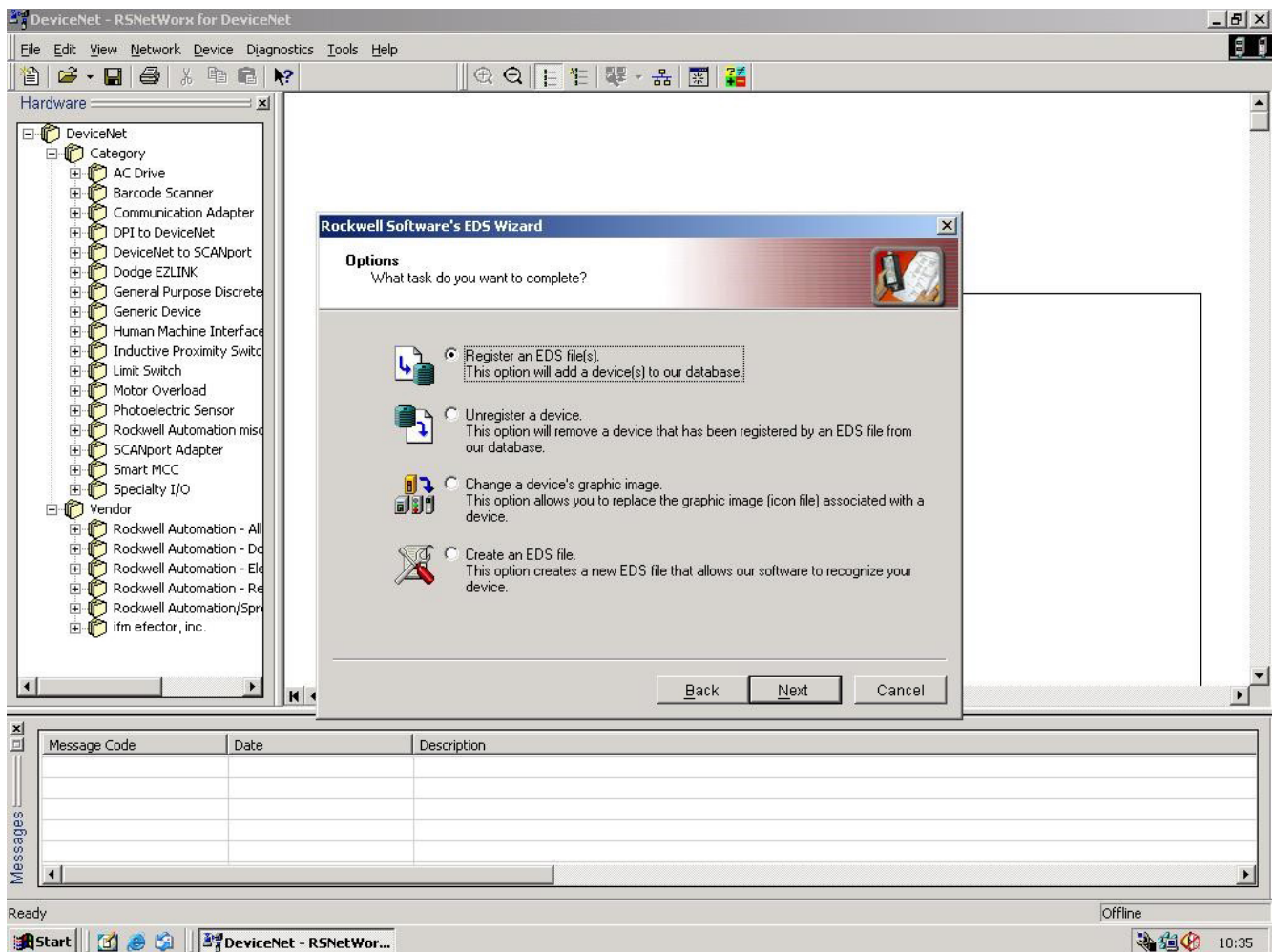
Der verwendete Service-Code zum Speichern der Werte ist herstellerspezifisch.

11 Projektierung unter RsNetworkx

11.1 EDS-File installieren

Das EDS-File enthält Informationen über gerätespezifische Parameter sowie mögliche Betriebsarten des Encoders. Hiermit steht in elektronischer Form eine Art Datenblatt zur Verfügung, das zur Projektierung unter z.B. RsNetworkx verwendet werden kann.

DE



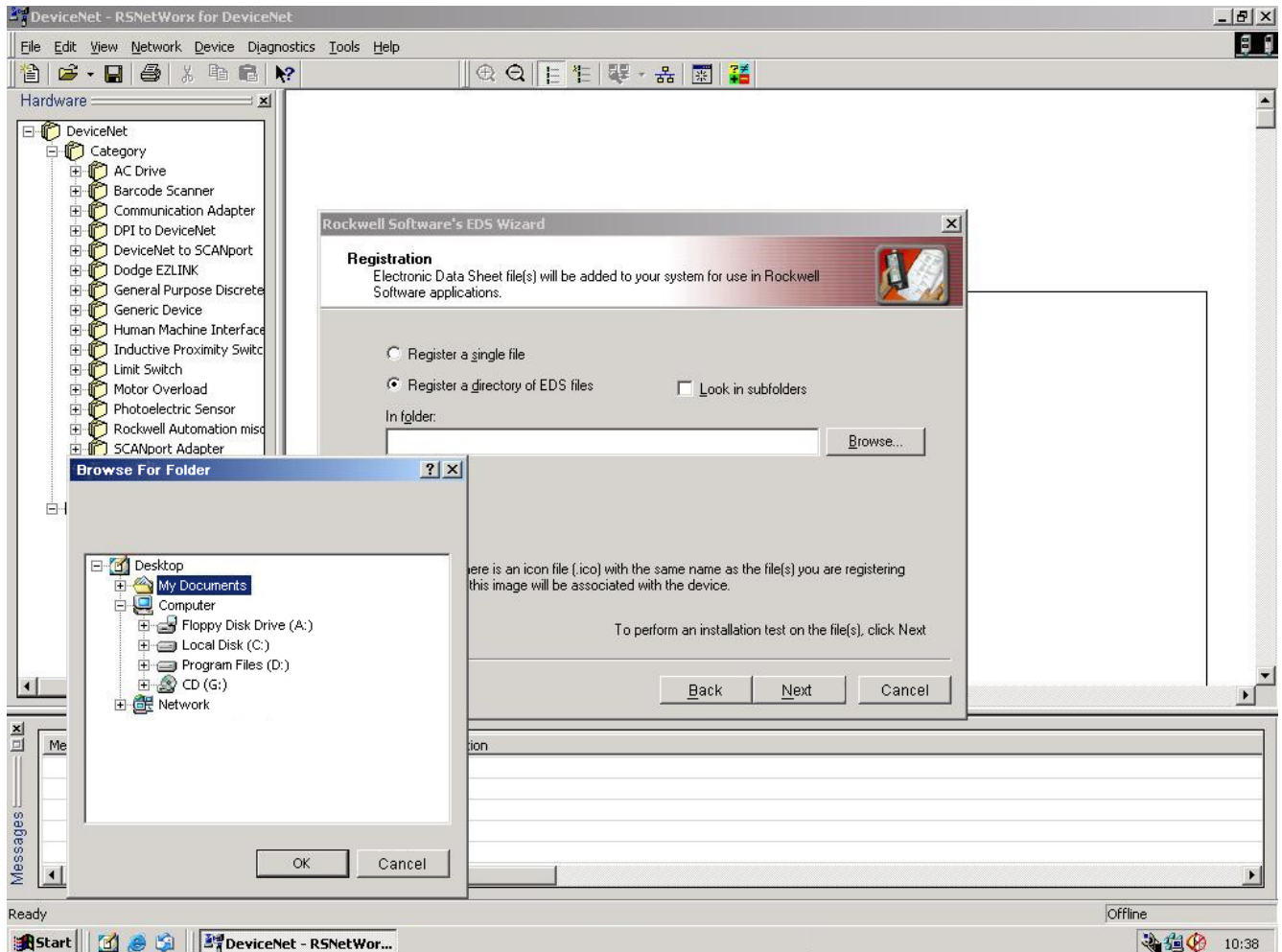
Bevor ein Encoder an den Bus angeschlossen werden kann, muss das EDS-File installiert werden. Das EDS-File wird mit dem EDS-Wizard installiert.

Zum Starten des EDS-Wizard in der Menüleiste "Tools / EDS-Wizard" anklicken. Nach erfolgreichen Start erscheint der EDS-Wizard (siehe Screenshot oben).

Anschließend den Punkt "Register an EDS File(s)" wählen und "Next" anklicken.

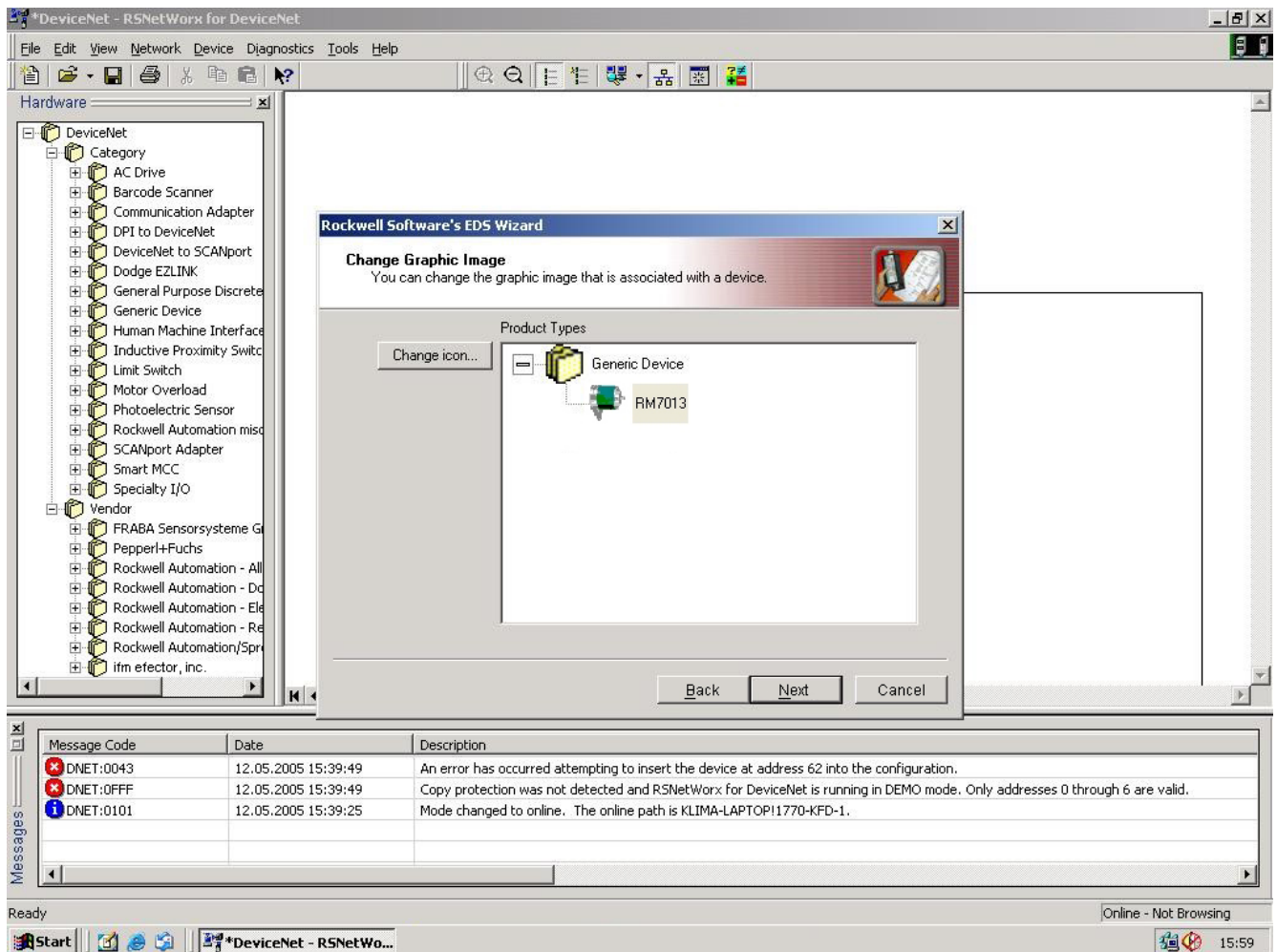
Im nächsten Schritt "Register a directory of EDS Files" wählen und "Browse" anklicken, um den Pfad des EDS-Files anzugeben (siehe Screenshot unten).

Der EDS-Wizard findet alle EDS-Files, die sich unter dem Pfad befinden und prüft die EDS-Files auf Fehler.



Die Schaltfläche "Next" anklicken, um zum Change Graphic Image-Fenster weitergeleitet zu werden (siehe nächsten Screenshot).

Im Change Graphic Image-Fenster werden den verwendeten Knoten Bilder zugeordnet.



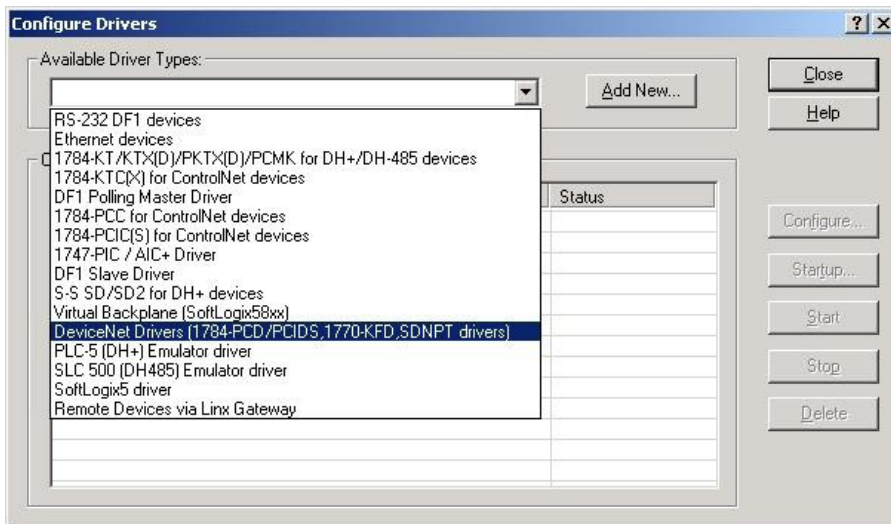
DE

Die Schaltfläche "Next" anklicken, um die Installation des EDS-File abzuschließen.

11.2 Treiber konfigurieren

Nachdem das EDS-File installiert ist, wird der passende Treiber konfiguriert. Im Folgenden wird der Treiber 1770-KFD konfiguriert.

Mit RSLinx wird der passende Treiber ausgewählt. Über "Start / Programme / Rockwell Software / RSLinx" das Programm RSLinx starten. In RSLinx in der Menüleiste "Communications / Configure Drivers" anklicken. Das Fenster "Configure Drivers" öffnet sich (siehe Screenshot unten).



Das Dropdown-Menü "Available Driver Types" anklicken und den Treiber 1770-KFD auswählen. Die Schaltfläche "Add New" anklicken, um den Treiber hinzuzufügen.

Der hinzugefügte Treiber kann jetzt konfiguriert werden. Den Treiber auswählen und die Schaltfläche "Configure" anklicken. Das Fenster "Driver Configuration" öffnet sich (siehe Screenshot unten).



Im Menü "Data Rate" die Baudrate des DeviceNet-Netzwerkes einstellen und die Schaltfläche "OK" anklicken. Das Fenster "Configure Drivers" durch Klicken auf die Schaltfläche "Close" schließen. Anschließend kann der konfigurierte Treiber verwendet werden.



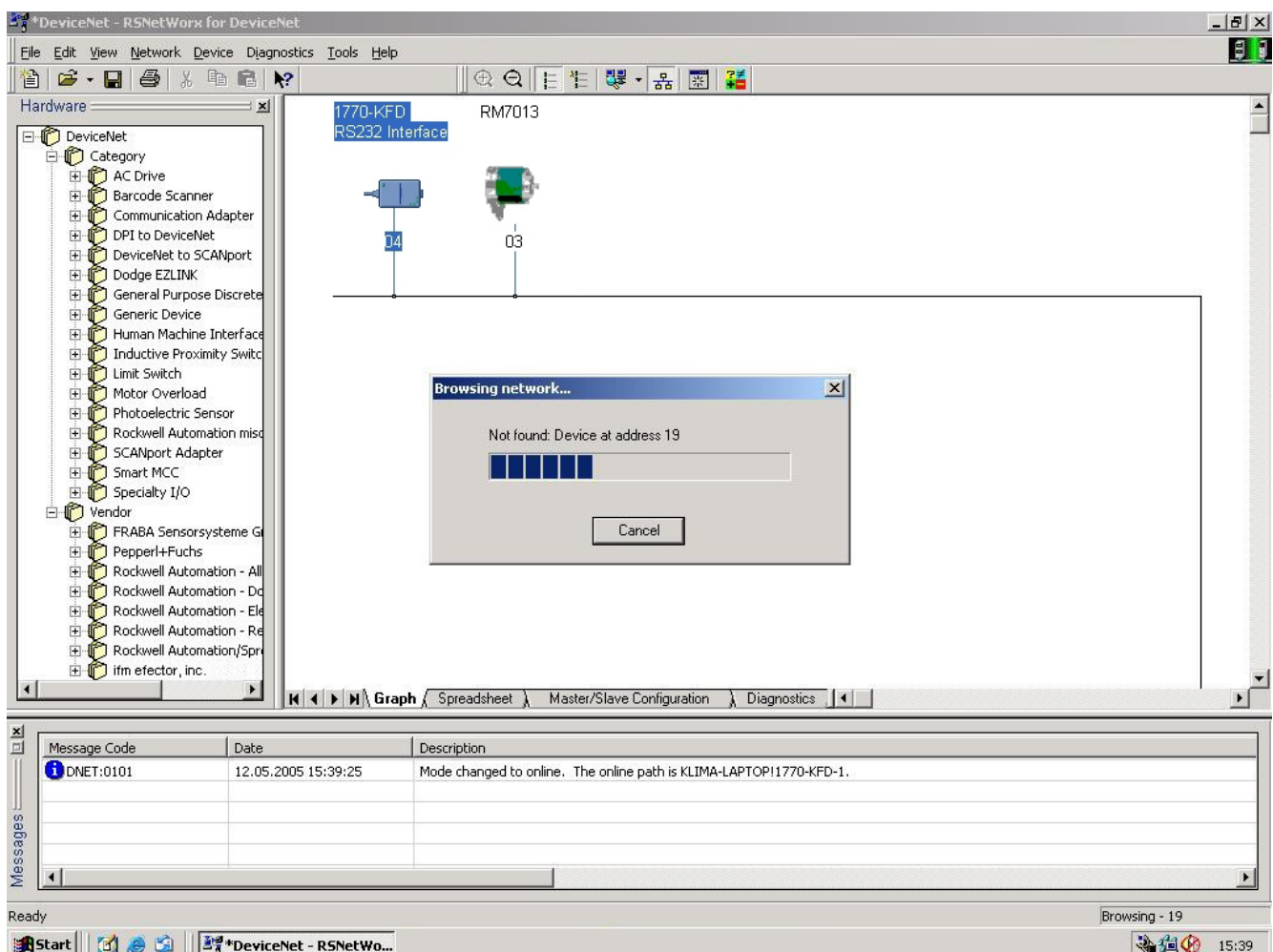
Damit der Treiber erfolgreich mit dem DeviceNet-Netzwerk kommunizieren kann, müssen der Treiber und das Netzwerk die selbe Baudrate verwenden. Verwenden Sie für den Treiber die Baudrate des DeviceNet-Netzwerkes.

DE

11.3 Netzwerk online schalten

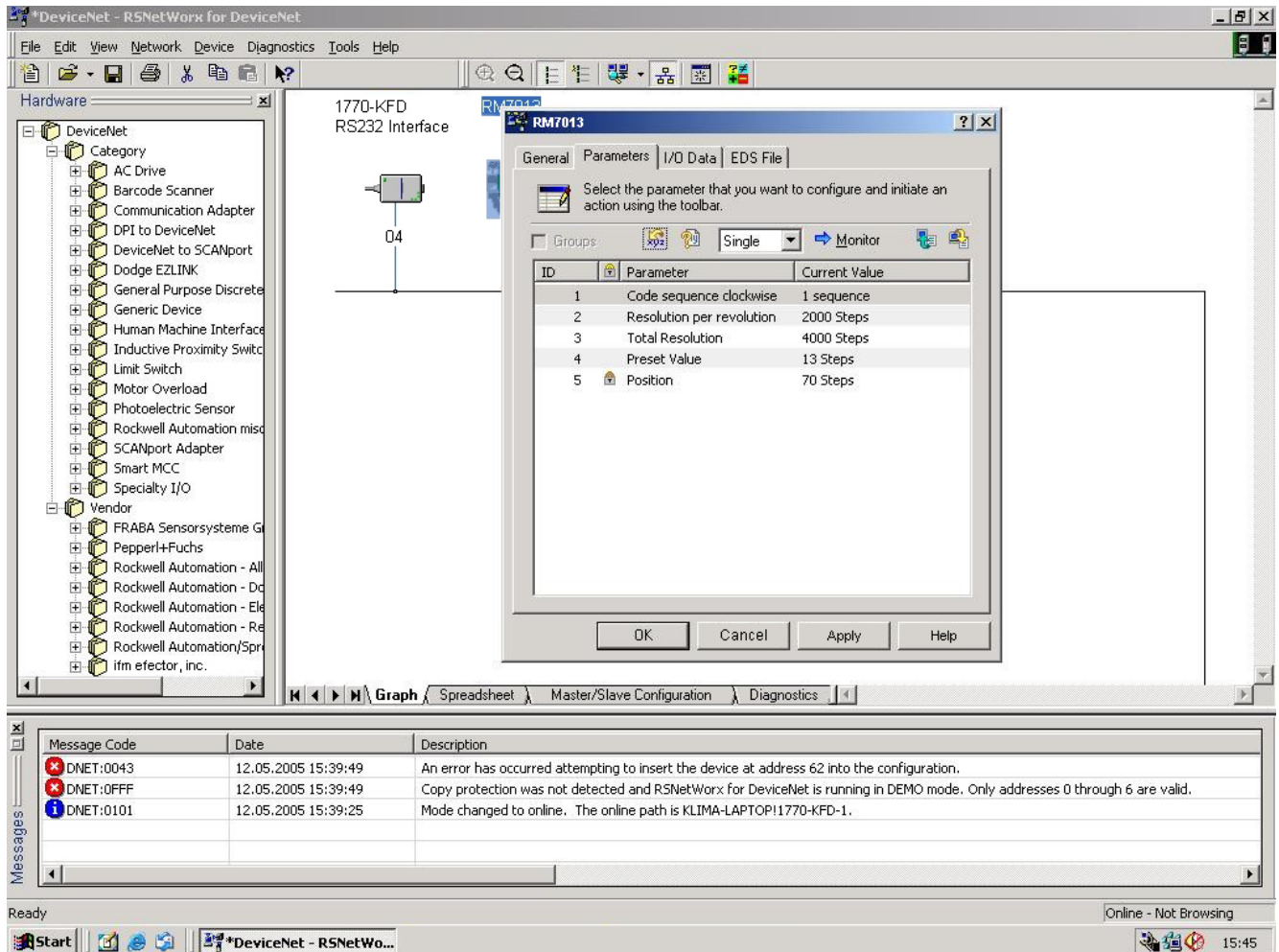
Zum Online schalten von Netzwerken in RsNetworx in der Menüleiste "Network / Online" anklicken. Das Fenster "Browse for network" öffnet sich und es wird nach verbundenen Netzwerken gesucht. Anschließend den konfigurierten Treiber auswählen (z.B. 1770-KFD), um das Netzwerk online zu schalten.

Nachdem das Netzwerk online geschaltet ist, sucht RsNetworx nach verfügbaren Knoten im Netzwerk (siehe Screenshot unten).



11.4 Encoder konfigurieren

Zum Konfigurieren von Encodern in RsNetworx in der Menüleiste "Device / Properties" anklicken. Das Konfigurationsfenster des Encoders öffnet sich (siehe Screenshot unten). Über den Tab "Parameters" können die Parameter des Encoders konfiguriert werden.

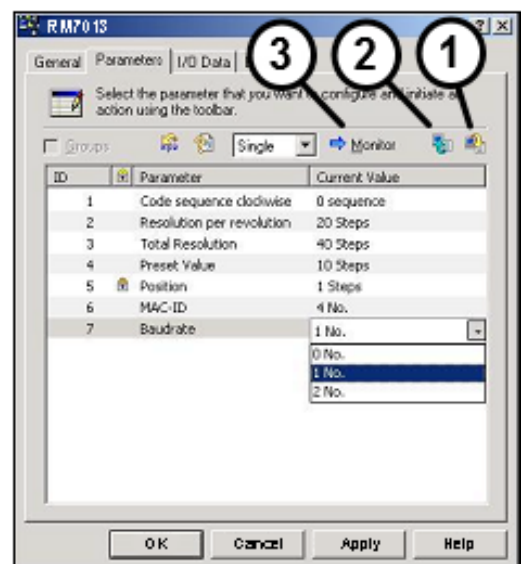


Nach dem Konfigurieren können die Parameter hochgeladen werden (siehe Screenshot rechts):

- ① Parameter herunterladen
- ② Parameter hochladen
- ③ Aktuellen Positionswert anzeigen



Die MAC-ID und Baudrate sind nur bei Encodern ohne Anschlusskabel änderbar.



Nach dem Hochladen müssen die Parameter im EEPROM gespeichert werden, um verwendet werden zu können. Zum Speichern der Parameter in RsNetwork in der Menüleiste "Device / Class Instance Editor" anklicken. Das Fenster "Service Class Instance Attribute Editor" öffnet sich (siehe Screenshot unten).

Die im Screenshot angegebenen Werte übernehmen. Anschließend die Schaltfläche "Execute" anklicken.

Die Werte sind jetzt im EEPROM gespeichert und können verwendet werden.

Service Class Instance Attribute Editor - [Node 32]

Unrecognized Device

Execute Transaction Arguments

Service Code

Value	Description
32	Other

Object Address

Class	Instance	Attribute
23	1	1

☒ Send the attribute ID

Transmit Data Size: Byte

Data sent to the device:

☐ Values in decimal

Execute

Receive Data

Size: Byte

Radix: Decimal

Data received from the device:

The execution was completed.

Close Help

DE

12 Begriffe und Abkürzungen

0b ...	binärer Zahlenwert (zur Bitcodierung), z.B. 0b0001 0000
0d ...	dezimaler Zahlenwert, z.B. 0d100
0x ...	hexadezimaler Zahlenwert, z.B. 0x64 (= 100 dezimal)
Baudrate	Übertragungsgeschwindigkeit (1 Baud = 1 Bit/sec.)
CAL	CAN Application Layer CAN basierendes Netzwerkprotokoll auf Applikationsebene
CAN	Controller Area Network (Bussystem für den Einsatz im Mobilbereich)
CAN_H	CAN-High; CAN-Anschluss/-Leitung mit dem hohen Spannungspegel
CAN_L	CAN-Low; CAN-Anschluss/-Leitung mit dem niederen Spannungspegel
CANopen	CAN basierendes Netzwerkprotokoll auf Applikationsebene mit einer offenen Konfigurationsschnittstelle (Objektverzeichnis).
CiA	"CAN in Automation e.V." (Anwender- und Herstellerorganisation in Deutschland/Erlangen) Definitions- und Kontrollorgan für CAN und CAN-basierende Netzwerkprotokolle
CiA DS	Draft Standard (veröffentlichte CiA-Spezifikation, die in der Regel ein Jahr nicht geändert und erweitert wurde)
CiA DSP	Draft Standard Proposal (veröffentlichter CiA-Spezifikationsentwurf)
CiA WD	Work Draft (CiA-intern zur Diskussion akzeptiertes Arbeitspapier)
CiA DS 301	Spezifikation zum CANopen Kommunikationsprofil; beschreibt die grundlegenden Kommunikationsmechanismen zwischen den Netzwerkteilnehmern, wie z.B. die Übertragung von Prozessdaten in Echtzeit, den Datenaustausch zwischen Geräten oder die Konfigurationsphase. Entspr. der Applikation ergänzt mit den nachfolgenden CiA-Spezifikationen:
CiA DS 401	Geräteprofil für digitale und analoge E/A-Baugruppen
CiA DS 402	Geräteprofil für Antriebe
CiA DS 403	Geräteprofil für Bediengeräte
CiA DS 404	Geräteprofil für Messtechnik und Regler
CiA DS 405	Spezifikation zur Schnittstelle zu programmierbaren Systemen (IEC 61131-3)
CiA DS 406	Geräteprofil für Drehgeber/Encoder
CiA DS 407	Applikationsprofil für den öffentlichen Nahverkehr
COB	CANopen Communication Object (PDO, SDO, EMCY, ...)
COB-ID	CANopen Identifier eines Communication Objects
Communication cycle	Die zu überwachende Synchronisationszeit; max. Zeit zwischen 2 Sync-Objekten
EMCY Object	Emergency Object (Alarmbotschaft; Gerät signalisiert einen Fehler)
Error Reg	Error Register (Eintrag mit einer Fehlerkennung)
Guarding Error	Knoten bzw. Netzwerkteilnehmer wurde bzw. wird nicht mehr gefunden Guard-MASTER: Einer oder mehrere SLAVES melden sich nicht mehr. Guard-SLAVE: Das Gerät (SLAVE) wird nicht mehr abgefragt.

Guard Time	Innerhalb dieser Zeit erwartet der Netzwerkteilnehmer ein "Node Guarding" des Netz-Masters
Heartbeat	Parametrierbare zyklische Überwachung von Netzwerkteilnehmern untereinander. Im Gegensatz zum „Node Guarding“ wird kein übergeordneter NMT-Master benötigt.
ID (auch Identifier)	Identifier; kennzeichnet eine CAN-Nachricht. Der numerische Wert des ID beinhaltet gleichzeitig eine Priorität bezüglich des Bus-Zugriffes. ID 0 = höchste Priorität.
Idx	Index; bildet zusammen mit dem S-Index die Adresse eines Eintrages im Objektverzeichnis
Life Time Factor	Anzahl der Versuche bei fehlender Guarding Antwort
Monitoring	Wird verwendet um die Fehlerklasse (Guarding-Überwachung, Sync-, etc.) zu beschreiben.
NMT	Netzwerk-Management
NMT-Master/-Slaves	Der NMT-Master steuert die Betriebszustände der NMT-Slaves
Node Guarding	Parametrierbare zyklische Überwachung von Slave-Netzwerkteilnehmern durch einen übergeordneten Master-noten, sowie die Überwachung dieses Abfragemechanismus durch die Slave-Teilnehmer.
Node-ID	Knotenpunkt-Identifier (Kennung eines Teilnehmers im CANopen Netz)
Objekt (auch OBJ)	Oberbegriff für austauschbare Daten/Botschaften innerhalb des CANopen-Netzwerks
Objektverzeichnis	enthält alle CANopen-Kommunikationsparameter eines Gerätes, sowie gerätespezifische Parameter und Daten. Auf die einzelnen Einträge wird über den Index und S-Index zugegriffen.
Operational	Betriebszustand eines CANopen Teilnehmers. In diesem Modus können SDOs, NMT-Kommandos und PDOs übertragen werden.
PDO	Process Data Object; im CANopen Netz zur Übertragung von Prozessdaten in Echtzeit, wie z.B. Drehzahl eines Motors. PDOs besitzen eine höhere Priorität als SDOs; im Gegensatz zu SDOs werden sie unbestätigt übertragen. PDOs bestehen aus einer CAN-Nachricht mit Identifier und bis zu 8 Byte Nutzdaten.
PDO Mapping	Beschreibt die Applikationsdaten, die mit einem PDO übertragen werden.
Pre-Op	Pre-Operational; Betriebszustand eines CANopen Teilnehmers. Nach dem Einschalten der Spannungsversorgung geht jeder Teilnehmer automatisch in diesen Zustand. Im CANopen-Netz können in diesem Modus nur SDOs und NMT-Kommandos übertragen werden, jedoch keine Prozessdaten
Prepared	(auch stopped) Betriebszustand eines CANopen Teilnehmers. In diesem Modus werden nur NMT- Kommandos übertragen.
Rec PDO (auch Rx PDO)	(Receive) Empfangs Process Data Object
ro	read only (unidirektional; nur Lesen)
rw	read-write (bidirektional; Lesen-Schreiben)
Rx-Queue	Empfangspuffer
s16	Datentyp signed 16 bit (mit Vorzeichen, 16 Bit-Format)

SDO	Service Data Object. Mit diesem Objekt wird gezielt auf das Objektverzeichnis eines Netzwerkteilnehmers zugegriffen (lesen/schreiben). Ein SDO kann aus mehreren CAN-Nachrichten bestehen. Die Übertragung der einzelnen Nachrichten wird von dem angesprochenen Teilnehmer bestätigt. Mit den SDOs lassen sich Geräte konfigurieren und parametrieren.
Server SDO	Mechanismus und Parametersatz um das "eigene" Objektverzeichnis eines Netzwerkteilnehmers anderen Teilnehmern (Clients) zugänglich zu machen.
S-Idx (auch SIdx)	Subindex innerhalb d. Objektverzeichnisses eines CANopen fähigen Gerätes
Start Guarding	Start der Knotenüberwachung
str	Datentyp String (Variable für Zeichenketten, wie z.B. Text "load")
Sync Error	Ausbleiben des Sync OBJ innerhalb der parametrierbaren Synchronisationszeit
Sync OBJ	Synchronisationsobjekt zur netzwerkweit gleichzeitigen Aktualisierung bzw. Übernahme der Prozessdaten der entsprechend parametrierten PDOs.
Sync Windows	Zeitfenster in dem die synchronen PDOs übertragen werden müssen.
Time Stamp	Zeitstempel zum Abgleich evtl. vorhandener Uhren in Netzwerkteilnehmern
Trans Type	Art der Prozess-Datenübertragung; synchron, asynchron
Trans PDO (auch Tx PDO)	(Transmit) Sende Process Data Object
Trans SDO (auch Tx SDO)	(Transmit) Sende Service Data Object
Tx-Queue (Transmit)	Sendepuffer
u8 (16, 32)	Datentyp unsigned 8 (16, 32) bit (ohne Vorzeichen, 8 (16, 32) Bit-Format)
wo	write only (nur schreiben)