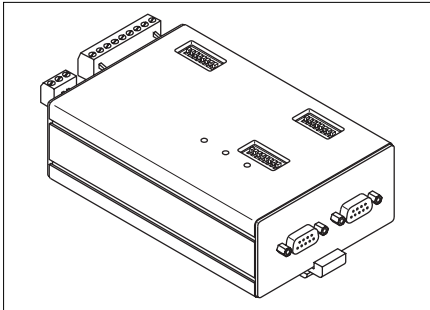


# IF09C

Interface mit CAN-Softwareprotokoll



Interface zur Anschaltung von max. 31 SIKO-Geräten mit RS485-Schnittstelle an den CAN-Bus.

## Eigenschaften

### Allgemeines

- Stromversorgung 24VDC, Stromaufnahme ca. 200 mA
- DC/DC-Wandler für galvanisch getrennte CAN-Schnittstelle
- Mikrocontroller 80C32
- 32kByte Programmspeicher
- 2kByte RAM (optional)
- Reset-Generator und Watchdogtimer

### CAN-Schnittstelle

- Full-CAN-Controller i82527
- Busankopplung mit PCA82C250T nach ISO-DIS 11898 (CAN High-Speed) mit galvanischer Trennung.
- Datenrate zwischen 1 MBit/s und 15,625 kBit/s über DIL-Schalter einstellbar.
- 11-Bit-Standard-Identifier (CAN Spezifikation 2.0 part A) oder 29-Bit Extended Identifier (CAN Spezifikation 2.0 part B) (nur bei Standardprotokoll) über DIP-Schalter umschaltbar.
- 5 Bit des Identifier sind über DIP-Schalter wählbar (Baugruppenadresse 0 ... 31).

### RS485-Schnittstelle

- Treiber zum Anschluss von max. 31 Sensoren. Protokoll der RS485-Schnittstelle auf AP09 oder SIKONETZ3 umschaltbar.
- Schnittstelleneinstellung AP09: 115200 Baud, 8 Datenbits, EVEN Parität, 1 Stopbit.  
Schnittstelleneinstellung SIKONETZ3: 19200 Baud, keine Parität, 8 Datenbits, 1 Stopbit.  
Telegrammaufbau für verschiedenen Sensoren über Tabellen einstellbar. Während der Initi-

sierungsphase werden von dem IF09C alle angeschlossenen Sensoren ermittelt und in eine Tabelle eingetragen.

## DEUTSCH

### 1. Sicherheitshinweise

- Lesen Sie vor der Montage und der Inbetriebnahme dieses Dokument sorgfältig durch. Beachten Sie zu Ihrer eigenen Sicherheit und der Betriebssicherheit alle Warnungen und Hinweise.
- Ihr Produkt hat unser Werk in geprüftem und betriebsbereitem Zustand verlassen. Für den Betrieb gelten die angegebenen Spezifikationen und die Angaben auf dem Typenschild als Bedingung.
- Garantieansprüche gelten nur für Produkte der Firma SIKO GmbH. Bei dem Einsatz in Verbindung mit Fremdprodukten besteht für das Gesamtsystem kein Garantieanspruch.
- Reparaturen dürfen nur im Werk vorgenommen werden. Für weitere Fragen steht Ihnen die Firma SIKO GmbH gerne zur Verfügung.

### 2. Identifikation

Das Typenschild zeigt den Gerätetyp mit Variantennummer. Die Lieferpapiere ordnen jeder Variantennummer eine detaillierte Bestellbezeichnung zu.

z.B. IF09C-0023



### 3. Mechanische Montage

Die Montage darf nur gemäß der angegebenen IP-Schutzart vorgenommen werden. Das System muss ggfs. zusätzlich gegen schädliche Umwelteinflüsse, wie z.B. Spritzwasser, Staub, Schläge, Temperatur geschützt werden.

Das IF09C ist zum Anbau an eine Hutschiene 35x7.5 nach DIN 50022 vorgesehen. Zur Montage ist der Halter auf die Hutschiene aufzuschnappen und darauf zu achten, dass sich die Verriegelung unten befindet. Zur Demontage wird der Riegel nach unten geschoben.

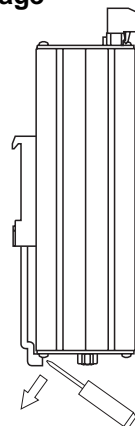


Abb. 1: Einbau

### 4. Elektrischer Anschluss

- Bei Verwendung in Antriebssystemen sind

zusätzliche Sicherheitsabschaltungen z.B. durch Endlagenschalter oder andere Verriegelungen vorzusehen.

- Verdrahtungsarbeiten dürfen nur spannungslos erfolgen!
- Litzen sind mit Aderendhülsen zu versehen.
- Vor dem Einschalten sind alle Leitungsanschlüsse und Steckverbindungen zu überprüfen.

### Hinweise zur Störsicherheit

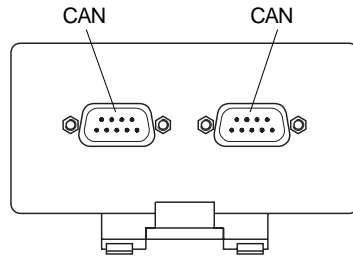
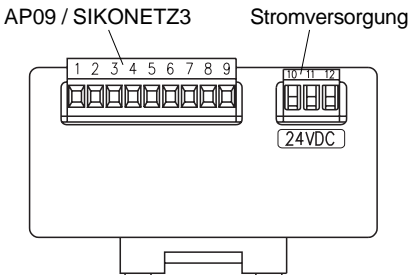
Alle Anschlüsse sind gegen äußere Störeinflüsse geschützt. **Der Einsatzort ist aber so zu wählen, dass induktive oder kapazitive Störungen nicht auf das Gerät oder dessen Anschlussleitungen einwirken können!** Durch geeignete Kabelführung und Verdrahtung können Störeinflüsse (z.B. von Schaltnetzteilen, Motoren, getakteten Reglern oder Schützen) vermindert werden.

### Erforderliche Maßnahmen:

- Nurgeschirmtes Kabel verwenden. Den Kabelschirm beidseitig auflegen. Litzenquerschnitt der Leitungen min. 0,14 mm<sup>2</sup>, max. 0,5 mm<sup>2</sup>.
- Die Verdrahtung PE Klemme 10, muss möglichst kurz und mit großem Querschnitt (2.5mm<sup>2</sup>) erfolgen.
- Das System muss in möglichst großem Abstand von Leitungen eingebaut werden, die mit Störungen belastet sind; ggfs. sind **zusätzliche Maßnahmen wie Schirmbleche oder metallisierte Gehäuse** vorzusehen. Leitungsführungen parallel zu Energieleitungen vermeiden.
- Schützpulen müssen mit Funkenlöschgliedern beschaltet sein.
- Zur Datenübertragung sind Kabellängen bis max. 200m möglich.

### Anschluss Stromversorgung

Der Anschluss erfolgt über die 3-pol. Klemmleiste an der Geräteoberseite.



### AP09 / SIKONETZ3 Anschluss

| Pin | Belegung                                                      |
|-----|---------------------------------------------------------------|
| 1   | GND                                                           |
| 2   | DÜB                                                           |
| 3   | DÜA                                                           |
| 4   | +U <sub>B</sub> 24VDC Geräteversorgung AP09 (Ausgang max. 2A) |
| 5   | PE<br>Schirm                                                  |
| 6   | GND                                                           |
| 7   | DÜB                                                           |
| 8   | DÜA                                                           |
| 9   | +U <sub>B</sub> 24VDC Geräteversorgung AP09 (Ausgang max. 2A) |

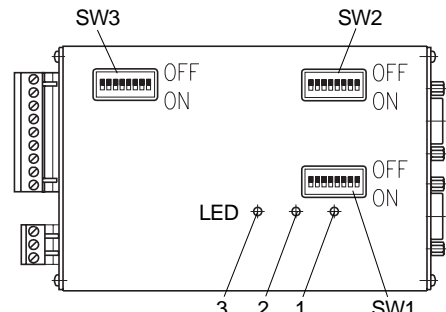
### Stromversorgung

| Pin | Belegung       |
|-----|----------------|
| 10  | PE             |
| 11  | GND            |
| 12  | +24VDC +/- 20% |

### Anschluss CAN

| Pin         | Belegung                                    |
|-------------|---------------------------------------------|
| 2           | CAN-Low                                     |
| 3           | CAN-GND ; CAN-Schnittstelle optoentkoppelt! |
| 7           | CAN-High                                    |
| 1, 4-6, 8-9 | N.C.                                        |

## 5. Funktion der DIP-Schalter



| LED-Anzeige | Farbe | Bedeutung                                                  |
|-------------|-------|------------------------------------------------------------|
| LED 1       | grün  | CAN-TX                                                     |
| LED 2       | rot   | Betriebsspannung und Betriebsspannung CAN-Schnittstelle ok |
| LED 3       | gelb  | CAN-RX                                                     |

### 5.1 Bitrate CAN-Bus

Mit den DIP-Schaltern SW3/1..SW3/4 wird die Bitrate auf dem CAN-Bus eingestellt. Sind alle Schalter offen, so ist die maximale Bitrate von 1MBit/s eingestellt. Die Einstellung für alle definierten Bitraten sind in der folgenden Tabelle dargestellt :

| SW3 |     |     |     | Bitrate [kbit/s] |
|-----|-----|-----|-----|------------------|
| 4   | 3   | 2   | 1   |                  |
| OFF | OFF | OFF | OFF | 1000             |
| OFF | OFF | OFF | ON  | 500              |
| OFF | OFF | ON  | OFF | 250              |
| OFF | OFF | ON  | ON  | 200              |
| OFF | ON  | OFF | OFF | 125              |
| OFF | ON  | OFF | ON  | 100              |
| OFF | ON  | ON  | OFF | 62,5             |
| OFF | ON  | ON  | ON  | 50               |
| ON  | OFF | OFF | OFF | 40               |
| ON  | OFF | OFF | ON  | 25               |
| ON  | OFF | ON  | OFF | 20               |
| ON  | OFF | ON  | ON  | 15,625           |
| ON  | ON  | OFF | OFF | 15,625           |
| ON  | ON  | OFF | ON  | 15,625           |
| ON  | ON  | ON  | OFF | 15,625           |
| ON  | ON  | ON  | ON  | 15,625           |

### 5.2 Standard-/Extended-Identifizier (nur Standard-Protokoll)

Mit dem DIP-Schalter SW3/5 kann zwischen 11-Bit Standard-Identifizier und 29 Bit Extended-Identifizier umgeschaltet werden. Die Einstellung gilt für alle Message-Objektes.

SW3/5 OFF: 29-Bit-Extended-Identifizier

SW3/5 ON: 11-Bit-Standard-Identifizier

### 5.3 Identifizier-Offset

Mit den DIP-Schaltern SW2/1..SW2/5 können die vom IF09C belegten Identifiziers mit einem Offset versehen werden, so dass max. 32 Gerädeadressen IF09C (0..31) verfügbar sind. Der Offset berechnet sich aus der Schalterstellung der entsprechenden Dip-Schalter.

### 11-Bit-Standard Identifizier

Aufbau des Identifiziers:

|              |              |             |             |             |
|--------------|--------------|-------------|-------------|-------------|
| ID11<br>CAN6 | ID10<br>CAN5 | ID9<br>CAN4 | ID8<br>CAN3 | ID7<br>CAN2 |
|--------------|--------------|-------------|-------------|-------------|

|             |              |              |              |              |
|-------------|--------------|--------------|--------------|--------------|
| ID6<br>CAN1 | ID5<br>SW2/5 | ID4<br>SW2/4 | ID3<br>SW2/3 | ID2<br>SW2/2 |
|-------------|--------------|--------------|--------------|--------------|

ID1  
SW2/1

CAN1..CAN6 sind die intern belegten 6 Identifizierbits.

Gültig sind hier die Schalter SW2/1..SW2/5.

### 29-Bit-Extended Identifizier (nur Standard-Protokoll)

Aufbau des Identifiziers:

|               |               |               |               |               |
|---------------|---------------|---------------|---------------|---------------|
| ID29<br>SW1/1 | ID28<br>SW1/2 | ID27<br>SW1/3 | ID26<br>SW1/4 | ID25<br>SW1/5 |
|---------------|---------------|---------------|---------------|---------------|

|               |               |               |           |           |
|---------------|---------------|---------------|-----------|-----------|
| ID24<br>SW1/6 | ID23<br>SW1/7 | ID22<br>SW1/8 | ID21<br>0 | ID20<br>0 |
|---------------|---------------|---------------|-----------|-----------|

|           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|
| ID19<br>0 | ID18<br>0 | ID17<br>0 | ID16<br>0 | ID15<br>0 |
|-----------|-----------|-----------|-----------|-----------|

|           |           |           |              |              |
|-----------|-----------|-----------|--------------|--------------|
| ID14<br>0 | ID13<br>0 | ID12<br>0 | ID11<br>CAN6 | ID10<br>CAN5 |
|-----------|-----------|-----------|--------------|--------------|

|             |             |             |             |              |
|-------------|-------------|-------------|-------------|--------------|
| ID9<br>CAN4 | ID8<br>CAN3 | ID7<br>CAN2 | ID6<br>CAN1 | ID5<br>SW2/5 |
|-------------|-------------|-------------|-------------|--------------|

|              |              |              |              |  |
|--------------|--------------|--------------|--------------|--|
| ID4<br>SW2/4 | ID3<br>SW2/3 | ID2<br>SW2/2 | ID1<br>SW2/1 |  |
|--------------|--------------|--------------|--------------|--|

Gültig sind hier die Schalter SW1/1..SW1/8 und SW2/1..SW2/5 .

CAN1..CAN6 sind die intern belegten 6 Identifizierbits.

### 5.4 Abschlusswiderstand der CAN-BUS-Leitung

Am letzten Busteilnehmer muss die Leitung mit einem Widerstand in Höhe des Wellenwiderstandes (ca. 120 Ohm) zwischen Pin 2 und 7 abgeschlossen werden. Sollte das IF09C der letzte oder erste Teilnehmer sein, so muss auf den freien Busanschluss ein Stecker mit eingebautem Abschlusswiderstand gesteckt werden.

## 5.5 Abschlusswiderstand der RS485-Leitung (nur Standard-Protokoll)

Auch die RS485 Busleitung muss mit dem Wellenwiderstand angeschlossen werden. Im IF09C ist bereits ein Busabschluss mit Pegelfestlegung integriert. Am (örtlich) letzten AP09 (bzw. SIKONETZ3 Gerät) muss ebenfalls ein Busabschluss durchgeführt werden. Im Falle der AP09 kann dies durch den Busabschlussstecker BAS09 erfolgen.

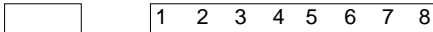
## 5.6 Umschaltung des Protokolls der RS485-Schnittstelle

An der RS485-Schnittstelle können wahlweise Sensoren mit AP09- oder SIKONETZ3-Protokoll angeschlossen werden. Mit dem Schalter SW3/6 kann das gewünschte Protokoll eingestellt werden :

SW3/6 OFF: AP09-Protokoll  
 SW3/6 ON: SIKONETZ3-Protokoll

## 6. Aufbau eines CAN-Telegramms

11 Bit Identifier bis zu 8 Byte Nutzdaten



Identifier:

Der Identifier legt die Priorität der Nachricht fest. CANopen legt mit dem Identifier außerdem die Geräteadresse, die Kanalauswahl und die Datenrichtung fest.

Nutzdaten:

Je nach Kanalauswahl durch den Identifier haben die Nutzdaten unterschiedliche Bedeutung.

## 7. Belegung der Identifier

| Kanal                       | ID (dezimal)         | ID (hexadezimal)     |
|-----------------------------|----------------------|----------------------|
| Netzwerkmanagement          | 0                    | 0                    |
| Synctelegramm               | 128                  | 80H                  |
| Prozessdatenkanal vom Gerät | 384 + Geräteadresse  | 180H + Geräteadresse |
| Parameterkanal vom Gerät    | 1408 + Geräteadresse | 580H + Geräteadresse |
| Parameterkanal zum Gerät    | 1536 + Geräteadresse | 600H + Geräteadresse |

## 8. Netzwerkmanagement

Zur Umschaltung zwischen den 3 möglichen Kommunikationszuständen INITIALISATION, PREOPERATIONAL und OPERATIONAL werden Telegramme mit dem Identifier 0 und 2 Byte

Nutzdaten verwendet.

### 8.1 INITIALISATION

In diesem Zustand ist das IF09C nicht am Geschehen auf den Bus beteiligt. Alle Hard- und Softwarekomponenten werden initialisiert. Dieser Zustand wird nach dem Einschalten des Gerätes oder nach dem Empfang des Befehls-codes 82H und der eigenen oder der globalen Adresse erreicht. Nach Abschluss der Initialisierung geht das Gerät automatisch in den Zustand PREOPERATIONAL über.

### 8.2 PREOPERATIONAL (vor betriebsbereit)

Der Austausch von Parametrierungsdaten zwischen dem IF09C und dem Busmaster ist freigegeben. Es werden jedoch keine Prozessdaten übertragen.

### 8.3 OPERATIONAL (betriebsbereit)

Der Austausch von Prozess- und Parametrierungsdaten ist freigegeben.

### 8.4 Umschaltung zwischen Kommunikationszuständen

Die Umschaltung zwischen Kommunikationszuständen kann vom Netzwerkmaster durch das Senden der folgenden Telegramme mit dem Identifier 0 ausgelöst werden.

| Zustandsänderung |                   | DATA 1 | DATA 2 |
|------------------|-------------------|--------|--------|
| von              | nach              |        |        |
| PREOPERATIONAL   | OPERATIONAL       | 01H    | xx     |
| OPERATIONAL      | PREOPERATIONAL    | 80H    | xx     |
| OPERATIONAL      | INITIALISATION    | 81H    | xx     |
| PREOPERATIONAL   | INITIALISATION    | 81H    | xx     |
| OPERATIONAL      | INITIALISATION(*) | 82H    | xx     |
| PREOPERATIONAL   | INITIALISATION(*) | 82H    | xx     |

(\*) Kaltstart wird ausgelöst.

xx = 0: Das Telegramm ist für alle Geräte am Bus bestimmt.

xx = Geräteadresse: Das Telegramm ist nur für das Gerät mit der entsprechenden Adresse bestimmt.

## 9. Parametrierung

Aufbau des CAN-Telegramms

11 Bit Identifier bis zu 8 Byte Nutzdaten

|  |   |   |   |   |   |   |   |   |
|--|---|---|---|---|---|---|---|---|
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|--|---|---|---|---|---|---|---|---|

Byte 1: Befehlscode

| Befehl<br>(Dienst) | Befehls-<br>code<br>(hexade-<br>zimal) | Befehls-<br>code<br>(dezimal) | Bedeutung                                  |
|--------------------|----------------------------------------|-------------------------------|--------------------------------------------|
| Write Request      | 23H                                    | 35                            | Parameter zum IF09C senden                 |
| Write Response     | 60H                                    | 96                            | Antwort des IF09C auf Write Request        |
| Read Request       | 40H                                    | 64                            | Anforderung eines Parameters von dem IF09C |
| Read Response      | 42H                                    | 66                            | Antwort auf Anforderung mit aktuellem Wert |
| Error Response     | 80H                                    | 128                           | Fehlermeldung                              |

### Byte 2,3: Parameterindex

Der Parameterindex wird im Intel-Datenformat im Nutzdatenbyte 2 (low Byte) und im Nutzdatenbyte 3 (high Byte) eingetragen. Die Indextabelle wird im Kapitel 'Indextabelle' beschrieben.

### Byte 4: Subindex

Im Subindex wird die Sensornummer des anzusprechenden Sensors eingetragen. Der Wertebereich ist 1..31.

### Byte 5..8: Data 1..Data 4

Im Datenbereich wird der Wert des Parameters in linksbündiger Intel-Darstellung eingetragen. (Byte 5 = low Byte .. Byte 8 = high Byte)

### 9.1 Fehlercodes

Befehlscode = 128 = 80H

Bei einem Kommunikationsfehler wird von dem IF09C eine Error Response gesendet. Dabei wird im Nutzdatenbyte 8 (Data 4) immer eine 6 und im Nutzdatenbyte 7 (Data 3) ein Fehlercode eingetragen.

Mögliche Fehlercodes

| Befehls-<br>code | Data<br>3 | Data<br>4 | Bedeutung                                                                               |
|------------------|-----------|-----------|-----------------------------------------------------------------------------------------|
| 80H              | 6         | 6         | Falscher Index                                                                          |
| 80H              | 5         | 6         | Falscher Subindex                                                                       |
| 80H              | 3         | 6         | Zugriff verweigert z.B. Write Request auf Read-Variable oder Parameter nicht vorhanden. |

### 9.2 Beispiel für Parameter schreiben

Das IF09C hat die Geräteadresse 1. Der Sensor Nr. 5 soll mit einem Offset von 40000 beaufschlagt werden.

Berechnung des Identifiers:

Identifier des Parameterkanals zum Gerät = 1536 + Geräteadresse

Identifier = 1536 + 1 = 1537 = 601H

Befehlscode = Write Request (Parameter zum Gerät senden) = 23H

Index = 5F02H

Der Index für den Parameter Offset wurde aus der Indextabelle im Kapitel 11 'Indextabelle' übernommen.

Subindex = Sensornummer 5 = 5

Parameterwert = 40000 = 9c40H

Telegramm zum IF09C :

Identifier

|            |     |     |     |     |     |     |     |     |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Identifier | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
| 601H       | 23H | 02H | 5FH | 05H | 40H | 9CH | 00H | 00H |

Antwort des IF09C bei fehlerfreier Ausführung:

|            |     |     |     |     |     |     |     |     |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Identifier | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
| 581H       | 60H | 02H | 5FH | 05H | 00H | 00H | 00H | 00H |

Befehlscode = Write Response (Antwort auf das Write Request) = 60H

Berechnung des Identifiers:

Identifier des Parameterkanals zum Gerät = 1408 + Geräteadresse

Identifier = 1408 + 1 = 1409 = 581H

### 9.3 Beispiel für Parameter lesen

Das IF09C hat die Geräteadresse 1. Der Winkelwert des Sensors Nr. 3 soll gelesen werden.

Berechnung des Identifiers:

Identifier des Parameterkanals zum Gerät = 1536 + Geräteadresse

Identifier = 1536 + 1 = 1537 = 601H

Befehlscode = Read Request (Anforderung zum Lesen eines Parameters) = 40H

Index = 5F00H

Der Index für den Parameter Winkelwert wurde aus der Indextabelle im Kapitel 'Indextabelle' übernommen.

Subindex = Sensornummer 3 = 3

Parameterwert = 40000 = 9c40H

Telegramm zum IF09C :



### 10.3 Encoder-PDO sperren (nur CAN-open-Protokoll)

Die Übertragung der Prozessdaten (d.h. des Winkelwertes) kann für jeden angeschlossenen Encoder einzeln durch Setzen des ihm zugeordneten disable-Bits gesperrt werden. D.h. von allen Encodern, deren Disable-Bit gesetzt ist, werden bei synchronen- und beim zyklischen Prozessdatenaustausch keine Winkelwerte übertragen. Als Voreinstellung, nach power-up oder dem NMT-Kommando INITIALISATION werden alle Disable-Bits gelöscht.

| Identifizier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
|--------------|-----|-----|-----|-----|----|----|----|----|
| 581H         | 23H | 0BH | 5FH | 00H | b0 | b1 | b2 | b3 |

b0: Disable-bits für Geber mit der Adresse 1..8  
 b1: Disable-bits für Geber mit der Adresse 9..16  
 b2: Disable-bits für Geber mit der Adresse 17..24  
 b3: Disable-bits für Geber mit der Adresse 25..31

Beispiel: Geber mit den Adressen 1,8 und 10 sperren, alle anderen freigeben

|      |     |     |     |     |     |     |     |     |
|------|-----|-----|-----|-----|-----|-----|-----|-----|
| 581H | 23H | 0BH | 5FH | 00H | 81h | 02h | 00h | 00h |
|------|-----|-----|-----|-----|-----|-----|-----|-----|

Die Disable-Bits sind rücklesbar (siehe Beispiel Parameterlesen)

Verteilung der Encoder-Nummern:

Beim power-up und beim NMT-Kommando INITIALISATION des IF09C werden alle an der RS485-Schnittstelle angeschlossenen Geber ermittelt und in mit dem Index ihrer Adresse in eine Liste eingetragen.

### 10.4 Tastenstatus (nur CAN-open-Protokoll)

Der Status der Bedientaste jedes angeschlossenen AP09-Gebers wird vom Gateway zyklisch ausgelesen. Jeder Geberadresse ist ein Bit zugeordnet, das beim Betätigen der Taste gesetzt wird. Alle gesetzten Tastenstatus-Bits werden beim Lesen des zugehörigen SDOs 5F0CH zurückgesetzt. Beim power-up werden ebenfalls alle Tastenstatusbits zurückgesetzt.

| Identifizier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
|--------------|-----|-----|-----|-----|----|----|----|----|
| 5F0CH        | 23H | 0CH | 5FH | 00H | b0 | b1 | b2 | b3 |

b0: Tasten-Statusbits für AP09-Geber mit der Adresse 1..8

b1: Tasten-Statusbits für AP09-Geber mit der Adresse 9..16

b2: Tasten-Statusbits für AP09-Geber mit der Adresse 17..24

b3: Tasten-Statusbits für AP09-Geber mit der Adresse 25..31

Beispiel: Tasten der Geber mit den Adressen 1,8 und 10 wurden gedrückt.

|       |     |     |     |     |     |     |     |     |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| 5F0CH | 23H | 0CH | 5FH | 00H | 81h | 02h | 00h | 00h |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|

### 10.5 Anwesenheitsstatus (nur CAN-open-Protokoll)

Während des power-up prüft das Gateway an welchen Adressen Geber angeschlossen sind. Das Gateway speichert diese Information als Bitfeld in einer Anwesenheitsliste. Jeder Geberadresse ist ein Bit zugeordnet. Dieses wird gesetzt wenn festgestellt wurde, dass ein Geber mit dieser Adresse angeschlossen ist.

| Identifizier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
|--------------|-----|-----|-----|-----|----|----|----|----|
| 5F0DH        | 23H | 0CH | 5FH | 00H | b0 | b1 | b2 | b3 |

b0: Anwesenheitsbits für AP09-Geber mit der Adresse 1..8

b1: Anwesenheitsbits für AP09-Geber mit der Adresse 9..16

b2: Anwesenheitsbits für AP09-Geber mit der Adresse 17..24

b3: Anwesenheitsbits für AP09-Geber mit der Adresse 25..31

Beispiel: Geber mit den Adressen 1,8 und 10 sind angeschlossen.

|       |     |     |     |     |     |     |     |     |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| 5F0CH | 23H | 0CH | 5FH | 00H | 81h | 02h | 00h | 00h |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|

### 10.6 Reset CAN-Gateway (nur CAN-open-Protokoll)

Das Scheitern eines Wertes > 00 00 00 00 auf das SDO 5F0EH löst einen Reset des Gateways aus. Dabei werden alle Funktionen genau wie beim power-up ausgeführt und alle Einstellungen auf die Default-Werte gesetzt.

Beispiel: Gateway-Reset ausführen

| Identifizier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
|--------------|-----|-----|-----|-----|----|----|----|----|
| 5F0EH        | 40H | 0EH | 5FH | 00H | 01 | 00 | 00 | 00 |

### 10.7 Status CAN-Gateway (nur CAN-open-Protokoll)

Über das SDO 5F0FH kann der aktuelle Betriebszustand des Gateways ausgelesen werden.

| Identifizier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
|--------------|-----|-----|-----|-----|----|----|----|----|
| 5F0FH        | 23H | 0FH | 5FH | 00H | B0 | 00 | 00 | 00 |

b0: 00H Status Initialisation

7FH Status Preoperational

05H Status Operational

Beispiel: aktueller Status Preoperational

|       |     |     |     |     |     |     |     |     |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| 5F0FH | 42H | 0FH | 5FH | 00H | 7FH | 00h | 00h | 00h |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|

## 11. Indextabelle

| Parameter                  | Index | Subindex | Datentype  | Datenlänge | Zugriff |
|----------------------------|-------|----------|------------|------------|---------|
| Winkelwert                 | 5F00H | 1 ... 31 | Integer 32 | 4          | R       |
| Kalibrierwert              | 5F01H | 1 ... 31 | Integer 32 | 4          | R/W     |
| Offset (*1)                | 5F02H | 1 ... 31 | Integer 32 | 4          | **R/W   |
| Geräteerkennung            | 5F03H | 0        | Integer 32 | 4          | R       |
| Konfiguration/Status       | 5F04H | 1 ... 31 | Integer 32 | 4          | R/W     |
| Anzeige pro Umdrehung      | 5F05H | 1 ... 31 | Integer 32 | 4          | R/W     |
| Systemstatus               | 5F06H | 0        | Integer 32 | 4          | R/W     |
| Softwarekalibrierung/Reset | 5F07H | 1 ... 31 | Integer 32 | 4          | W(*2)   |
| Prozessdatenmode           | 5F08H | 0        | Integer 32 | 4          | R/W     |
| Prozessdatenzykluszeit     | 5F09H | 0        | Integer 32 | 4          | R/W     |
| Sollwert                   | 5FOAH | 1 ... 31 | Integer 32 | 4          | W       |
| **Encoder PDO sperren(*3)  | 5FOBH | 0        | Integer 32 | 4          | R/W     |
| **Tastenstatus(*4)         | 5FOCH | 0        | Integer 32 | 4          | R       |
| **Anwesenheitsstatus(*5)   | 5FODH | 0        | Integer 32 | 4          | R       |
| **Reset CAN-Gateway        | 5FOEH | 0        | Integer 32 | 4          | W       |
| **Status CAN-Gateway       | 5FOFH | 0        | Integer 32 | 4          | R       |

\*\* Nur für CAN-open-Protokoll

**(\*1)** Dieser Parameter ist bei AP09-Drehgebern nicht vorhanden. Ein Zugriff darauf liefert als response den Fehlercode 'Zugriffsfehler'.

**(\*2)** Der Inhalt der Nutzdatenbytes sind ohne Bedeutung.

**(\*3)** Jedem angeschlossenen Geber ist ein Bit zugeordnet (siehe Abschnitt 10.3)

**(\*4)** Jedem angeschlossenen Geber ist ein Bit zugeordnet (siehe Abschnitt 10.4)

**(\*5)** Jedem angeschlossenen Geber ist ein Bit zugeordnet (siehe Abschnitt 10.5)

Subindex 0 bedeutet, dass dieser Parameter das IF09C selbst betrifft.

Subindex 1..31 adressiert den angeschlossenen Geber mit der entsprechenden Adresse.

Zugriff = R: nur Lesezugriff möglich.

Zugriff = W: nur Schreibzugriff möglich.

Zugriff = R/W: Schreib- und Lesezugriff möglich.

Die Darstellung aller Werte erfolgt im Intel-Format.

### 11.1 Parameter Geräteerkennung

liefert im

1. Datenbyte: 7 = Gerätetyp des IF09C,
2. Datenbyte: Versionsnummer des IF09C (z.B. V1.01 = 11H),
- alle anderen Datenbytes enthalten 0.

### 11.2 Parameter Konfiguration/Status

liefert beim AP09:

1. Datenbyte stehen die folgenden Informationen:

| Bit-Nr. | IF09C (CAN-Bus)<br>-> AP09                                                                       | AP09->IF09C<br>(CAN-Bus)                                                                                                            |
|---------|--------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| 7(MSB)  | keine Funktion                                                                                   | Bit =1: Batterie leer                                                                                                               |
| 6       | frei                                                                                             | -Standrad-Protokoll frei<br>-CAN-open-Protokoll<br>Tastenstatus: wird bei Tastendruck auf 1 gesetzt und beim Auslesen zurückgesetzt |
| 5-4     | Tastenmode<br>00: keine Tastenfunktion<br>01: Kettenmaß<br>10: Rücksetzen<br>11: Sollwertanzeige | Tastenmode<br>00: keine Tastenfunktion<br>01: Kettenmaß<br>10: Rücksetzen<br>11: Sollwertanzeige                                    |
| 3       | Rücksetzen                                                                                       | keine Funktion                                                                                                                      |
| 2       | Kettenmaß ausführen                                                                              | keine Funktion                                                                                                                      |
| 1       | frei                                                                                             | frei                                                                                                                                |
| 0(LSB)  | Drehrichtung:<br>Bit = 0: gegen den Uhrzeigersinn<br>Bit = 1: mit dem Uhrzeigersinn              | Drehrichtung:<br>Bit = 0: gegen den Uhrzeigersinn<br>Bit = 1: mit dem Uhrzeigersinn                                                 |

2. Datenbyte: Anzahl der Nachkommastellen
3. Datenbyte: Versionsnummer



### 11.3 Parameter Systemstatus

liefert im

1. Datenbyte: 0: IF09C nicht bereit (Sensor-identifikation läuft)  
1: IF09C bereit.
2. Datenbyte: Anzahl der angeschlossenen Sensoren (0..31)

## 12. Kommunikation-Profil-Objekte nach Encoder-Profil 406 (nur CAN-open-Protokoll)

### 12.1 Device Type

|              |                            |
|--------------|----------------------------|
| <b>Index</b> | <b>1000h</b>               |
| Name         | device type                |
| Object Code  | VAR                        |
| Data Type    | Unsigned32                 |
| Access       | Read_only                  |
| Value        | 0x96, 0x01, 0x00, 0x00 (*) |

(\*) Für dieses Gerät müsste nach dem Device Profile für Geber 406 im 3. Byte der Wert 10 für Multisensordevice eingetragen werden. Da jedoch noch nicht alle Konfigurations-Tools diesen Typ kennen, wurde dieses Byte auf 0x00 gesetzt, um ein sicheres Hochhalten während der Geräteidentifizierung zu gewährleisten.

### 12.2 Error Register

|              |                        |
|--------------|------------------------|
| <b>Index</b> | <b>1001h</b>           |
| Name         | Error register         |
| Object Code  | VAR                    |
| Data Type    | Unsigned32             |
| Access       | Read-only              |
| Value        | 0x00, 0x00, 0x00, 0x00 |

### 12.3 COB-ID Sync Message

|              |                        |
|--------------|------------------------|
| <b>Index</b> | <b>1005h</b>           |
| Name         | COB-ID SYNC            |
| Access       | Read-only              |
| Data Type    | Unsigned32             |
| Value        | 0x80, 0x00, 0x00, 0x00 |

### 12.4 Node-ID (Knotenadresse)

|              |                                  |
|--------------|----------------------------------|
| <b>Index</b> | <b>100Bh</b>                     |
| Name         | COB-ID SYNC                      |
| Access       | Read-only                        |
| Data Type    | Unsigned32                       |
| Value        | Einstellung von DIP-Schalter SW2 |

### 12.5 Guard Time

|               |              |
|---------------|--------------|
| <b>Index</b>  | <b>100Ch</b> |
| Name          | COB-ID SYNC  |
| Access        | Read-only    |
| Data Type     | Unsigned32   |
| Default-Value | 300 ms       |

### 12.6 Identity Object

|              |                |
|--------------|----------------|
| <b>Index</b> | <b>1018h</b>   |
| Name         | Error register |
| Object Code  | VAR            |
| Data Type    | Unsigned32     |

| Sub-Index | Description       | Access    | Value                |
|-----------|-------------------|-----------|----------------------|
| 00h       | number of entries | Read-only | 3                    |
| 01h       | Vendor ID         | Read-only | 0x55,0xAA, 0x00,0xFF |
| 02h       | Product Code      | Read-only | 0x07,0x00, 0x00,0x00 |
| 03h       | Revision Number   | Read-only | 0x00,0x20, 0x00,0x20 |

### 12.7 Transmit PDO-Objekte

|              |                               |
|--------------|-------------------------------|
| <b>Index</b> | <b>1800h+(Geberadresse-1)</b> |
| Name         | Transmit PDO parameter        |
| Object Code  | RECORD                        |
| Data Type    | PDO CommPar                   |

| Sub-Index | Description         | Access     | (Default)-Value                     |
|-----------|---------------------|------------|-------------------------------------|
| 00h       | number of entries   | Read-only  | 31                                  |
| 01h       | COB-ID used by PDO1 | Read/Write | ID=Datum+ Geräte-ID+ Geber- adresse |
| 02h       | Transmission type   | Read-only  | 0xFE,0x00, 0x00,0x00                |

(Geräte-ID von Schalter SW2)

|              |                               |
|--------------|-------------------------------|
| <b>Index</b> | <b>1A00h+(Geberadresse-1)</b> |
| Name         | Server SDO parameter          |
| Object Code  | RECORD                        |
| Data Type    | PDO Mapping                   |

| Sub-Index | Description         | Access    | Value                  |
|-----------|---------------------|-----------|------------------------|
| 00h       | number of entries   | Read-only | 2                      |
| 01h       | COB-ID used by PDO1 | Read-only | (0x80,0x01, 0x00,0x00) |
| 02h       | Transmission type   | Read-only | 0x01,0x00, 0x00,0x00   |

## 13. Service-Data-Objekte (SDO) nach Device-Profil für Encoder 406, Version 2.0 (nur CAN-open-Protokoll)

### 13.1 Offset (Preset\_Value)

| Index       | 6010h                                |            |            |
|-------------|--------------------------------------|------------|------------|
| Name        | Preset_value_for multisensor_devices |            |            |
| Object Code | ARRAY                                |            |            |
| Data Type   | Unsigned8                            |            |            |
| Sub-Index   | Description                          | Access     | ValueRange |
| 00h         | Number_of_available_channels         | Read/write | 00..31(*)  |
| 01h         | Preset_value_channel_1(*)            | Read/write | Integer32  |
| 02h         | Preset_value_channel_2(*)            | Read/write | Integer32  |
| 03h..1eh    | Preset_value_channel_3..30(*)        | Read/write | Integer32  |
| 1fh         | Preset_value_channel_31(*)           | Read/write | Integer32  |

(\*) Abhängig von der Anzahl der angeschlossenen Geber.

### 13.2 Winkelwert (Position\_value)

| Index       | 6020h                                  |           |            |
|-------------|----------------------------------------|-----------|------------|
| Name        | Position_value_for multisensor_devices |           |            |
| Object Code | ARRAY                                  |           |            |
| Data Type   | Unsigned8                              |           |            |
| Sub-Index   | Description                            | Access    | ValueRange |
| 00h         | Number_of_available_channels           | Read-only | 00..31(*)  |
| 01h         | Position_value_channel_1(*)            | Read-only | Integer32  |
| 02h         | Position_value_channel_2(*)            | Read-only | Integer32  |
| 03h..1eh    | Position_value_channel_3..30(*)        | Read-only | Integer32  |
| 1fh         | Position_value_channel_31(*)           | Read-only | Integer32  |

(\*) Abhängig von der Anzahl der angeschlossenen Geber.

### 13.3 Cyclic Timer

| Index         | 6200h                   |
|---------------|-------------------------|
| Name          | Cyclic Timer            |
| Object Code   | Var                     |
| Data Type     | Unsigned16              |
| Access        | Read/write              |
| Value Range   | Unsigned16 (0..65535ms) |
| Default Value | 100 (100 ms)            |

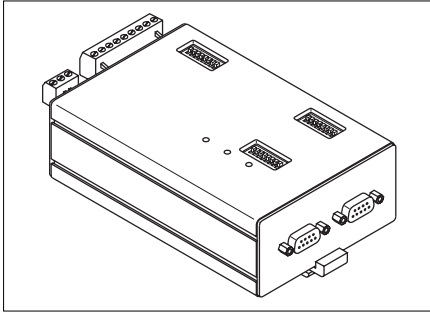
Zykluszeit für die Übertragung der Prozessdaten im Modus 'zyklische Prozessdatenübertragung' (die erreichbare minimale Zykluszeit wird von der Anzahl der angeschlossenen Geber und deren Kommunikationszeit mit dem Gateway über die RS485-Schnittstelle begrenzt).

## 14. EDS-Datei (nur CAN-open-Protokoll)

Für das IF09C steht die EDS-Datei 'SIKOCGW.EDS' (electronic-data-sheet) zur Verfügung, die im Lieferumfang der Baugruppe enthalten ist. Mit Hilfe dieser Datei ist die einfache Einbindung und Konfigurierung des IF09C in ein CANopen-Netzwerks mittels handelsüblicher CANopen-Konfiguratoren möglich.

# IF09C

Interface with CAN software protocol



Interface for connecting max. 31 SIKO devices with RS485 interface to the CAN bus.

## Features

### General

- Power supply 24VDC, power consumption appr. 200 mA
- DC/DC converter for DC isolated CAN interface
- Micro-controller 80C32
- 32kbyte program storage
- 2kbyte RAM (optional)
- Reset-generator and watchdog timer

### CAN interface

- Full-CAN controller i82527
- Bus linking with PCA82C250T acc. to ISO-DIS 11898 (CAN high speed) with DC isolation.
- Data rate between 1 Mbits/s and 15.625 kbit/s, adjustable via DIL switch.
- 11 bits standard identifier (CAN specification 2.0 part A) or 29 bits extended identifier (CAN specification 2.0 part B) (only with standard protocol) switchable by DIP switch.
- 5 bits of the identifier may be selected by DIP switch (assembly address 0 ... 31).

### RS485 Interface

- Driver for connecting max. 31 sensors. Protocol of the RS485 interface can be switched to AP09 or SIKONETZ3.

Interface setting AP09: 115200 baud, 8 data bits, EVEN parity, 1 stop bit.

Interface setting SIKONETZ3: 19200 baud, no parity, 8 data bits, 1 stop bit.

Telegram setup adjustable for different sensors via tables. During initialization the IF09C identifies each connected sensor and enters them in a table.

## 1. Safety information

In order to carry out installation correctly, we strongly recommend this document is read very carefully. This will ensure your own safety and the operating reliability of the device.

- Your device has been quality controlled, tested and is ready for use. Please respect all warnings and information which are marked either directly on the device or in this document.
- Warranty can only be claimed for components supplied by SIKO GmbH. If the system is used together with other products, the warranty for the complete system is invalid.
- Repairs should be carried out only at our works. If any information is missing or unclear, please contact the SIKO sales staff.

## 2. Identification

Please check particular type of unit and type number from the identification plate. Type number and the corresponding execution are indicated in the delivery documentation.

e.g. IF09C-0023  
└───┬───┘ type number  
└───┘ type of unit

## 3. Installation

The unit should be used only according to the protection level provided. Protect the unit, if necessary, against environmental influences such as sprayed water, dust, knocks, extreme temperatures.

IF09P has been designed for mounting on standard rails 35x7.5 accord. to DIN 50022. For mounting snap the retainer onto the rail and ensure that the lock is at the device's underside. For dismounting, the lock is pushed downward.

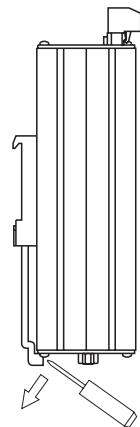


Fig. 1: Installation

## 4. Electrical connection

- If the unit is used in combination with drive systems, additional safety protection must be used (eg. limit switches or other interlocking systems).
- Wiring must only be carried out with power off.

- Provide stranded wires with ferrules.
- Check all lines and connections before switching on the equipment.

### Interference and distortion

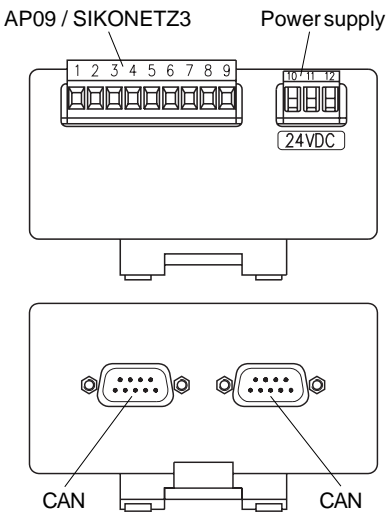
All connections are protected against the effects of interference. **The location should be selected to ensure that no capacitive or inductive interferences can affect the encoder or the connection lines!** Suitable wiring layout and choice of cable can minimise the effects of interference (eg. interference caused by switching power supplies, motors, cyclic controls and contactors).

### Necessary steps:

- Only screened cable should be used. Screen should be connected to earth at both ends. Wire cross section is to be at least 0,14 mm<sup>2</sup>, max. 0,5 mm<sup>2</sup>.
- Wiring of PE terminal 10 should be as short as possible and with a large cross section (2.5mm<sup>2</sup>).
- The unit should be positioned well away from cables with interference; if necessary a **protective screen or metal housing must be provided**. The running of wiring parallel to the mains supply should be avoided.
- Contactor coils must be linked with spark suppression.
- With data transmission max. allowable cable length is 200m.

### Connection of power supply:

via the 3-pin terminal strip on front of the device.



### AP09 / SIKONETZ3 connection

| Pin | Description                                                  |
|-----|--------------------------------------------------------------|
| 1   | GND                                                          |
| 2   | DÜB                                                          |
| 3   | DÜA                                                          |
| 4   | +U <sub>B</sub> 24VDC device supply<br>AP09 (output max. 2A) |
| 5   | PE<br>screen                                                 |
| 6   | GND                                                          |
| 7   | DÜB                                                          |
| 8   | DÜA                                                          |
| 9   | +U <sub>B</sub> 24VDC device supply<br>AP09 (output max. 2A) |

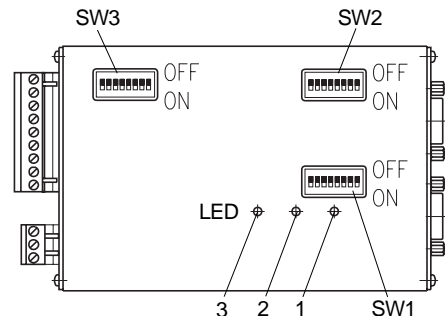
### Power supply

| Pin | Description    |
|-----|----------------|
| 10  | PE             |
| 11  | GND            |
| 12  | +24VDC +/- 20% |

### Connection CAN

| Pin         | Description                                |
|-------------|--------------------------------------------|
| 2           | CAN-low                                    |
| 3           | CAN-GND ; CAN interface<br>opto-decoupled! |
| 7           | CAN-high                                   |
| 1, 4-6, 8-9 | N.C.                                       |

## 5. Funktion der DIP-Schalter



| LED-display | Color  | Description                                                   |
|-------------|--------|---------------------------------------------------------------|
| LED 1       | green  | CAN-TX                                                        |
| LED 2       | red    | operating voltage and operating voltage of CAN interface okay |
| LED 3       | yellow | CAN-RX                                                        |

### 5.1 Bit Rate CAN Bus

The bit rate on the CAN bus is set by means of the DIP switches SW3/1..SW3/4. The maximum bit rate of 1 Mbit/s is set if all switches are open. The settings for all defined bit rates are represented in the following table :

| SW3 |     |     |     | Bitrate [kbit/s] |
|-----|-----|-----|-----|------------------|
| 4   | 3   | 2   | 1   |                  |
| OFF | OFF | OFF | OFF | 1000             |
| OFF | OFF | OFF | ON  | 500              |
| OFF | OFF | ON  | OFF | 250              |
| OFF | OFF | ON  | ON  | 200              |
| OFF | ON  | OFF | OFF | 125              |
| OFF | ON  | OFF | ON  | 100              |
| OFF | ON  | ON  | OFF | 62,5             |
| OFF | ON  | ON  | ON  | 50               |
| ON  | OFF | OFF | OFF | 40               |
| ON  | OFF | OFF | ON  | 25               |
| ON  | OFF | ON  | OFF | 20               |
| ON  | OFF | ON  | ON  | 15,625           |
| ON  | ON  | OFF | OFF | 15,625           |
| ON  | ON  | OFF | ON  | 15,625           |
| ON  | ON  | ON  | OFF | 15,625           |
| ON  | ON  | ON  | ON  | 15,625           |

### 5.2 Standard / Extended Identifier (only standard protocol)

The DIP switch SW3/5 serves for switching between the 11 bits standard identifier and the 29 bits extended identifier. The setting applies to all message objects.

SW3/5 OFF: 29 bits extended identifier

SW3/5 ON: 11 bits standard identifier

### 5.3 Identifier offset

By means of the DIP switches SW2/1..SW2/5 the identifiers assigned by IF09C may be provided with an offset, so that a maximum of 32 device addresses IF09C (0..31) will be available. The offset is calculated from the switch positioning of the respective DIP switch.

#### 11 bits standard identifier

Setup of the identifier:

|              |              |              |              |              |
|--------------|--------------|--------------|--------------|--------------|
| ID11<br>CAN6 | ID10<br>CAN5 | ID9<br>CAN4  | ID8<br>CAN3  | ID7<br>CAN2  |
| ID6<br>CAN1  | ID5<br>SW2/5 | ID4<br>SW2/4 | ID3<br>SW2/3 | ID2<br>SW2/2 |
| ID1<br>SW2/1 |              |              |              |              |

CAN1..CAN6 are the internally assigned 6 identifier bits.

Switches SW2/1..SW2/5 are valid here.

### 29 bits extended identifier (only standard protocol)

Setup of the identifier:

|               |               |               |               |               |
|---------------|---------------|---------------|---------------|---------------|
| ID29<br>SW1/1 | ID28<br>SW1/2 | ID27<br>SW1/3 | ID26<br>SW1/4 | ID25<br>SW1/5 |
| ID24<br>SW1/6 | ID23<br>SW1/7 | ID22<br>SW1/8 | ID21<br>0     | ID20<br>0     |
| ID19<br>0     | ID18<br>0     | ID17<br>0     | ID16<br>0     | ID15<br>0     |
| ID14<br>0     | ID13<br>0     | ID12<br>0     | ID11<br>CAN6  | ID10<br>CAN5  |
| ID9<br>CAN4   | ID8<br>CAN3   | ID7<br>CAN2   | ID6<br>CAN1   | ID5<br>SW2/5  |
| ID4<br>SW2/4  | ID3<br>SW2/3  | ID2<br>SW2/2  | ID1<br>SW2/1  |               |

Here, the switches SW1/1..SW1/8 and SW2/1..SW2/5 are valid .

CAN1..CAN6 are the internally assigned 6 identifier bits.

### 5.4 Terminator of the CAN BUS line

The line must be terminated at the last bus station by means of a resistance equal to the characteristic impedance (appr. 120 ohm) between pins 2 and 7. If the IF09C is the last or the first station, then a connector with an integrated terminator must be plugged into the free bus connection.

### 5.5 Terminator of the RS485 line (only standard protocol)

The RS485 bus line, too, must be terminated by the characteristic impedance. In the IF09C a bus terminator with level specification is already integrated. At the (locally) last AP09 (or SIKONETZ3 device, respectively) a bus termination must be performed as well. In the case of AP09 the bus terminator connector BAS09 can be used.

### 5.6 Switching of the RS485 interface protocol

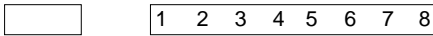
Sensors with AP09 or SIKONETZ3 protocol may be optionally connected to the RS485 interface. The desired protocol may be set by means of switch SW3/6 :

SW3/6 OFF: AP09 protocol

SW3/6 ON: SIKONETZ3 protocol

## 6. Setup of a CAN telegram

11 bits identifier up to 8 bytes user data



Identifier:

The identifier defines the priority of the message. Furthermore, CANopen determines the device address, the channel selection and the data direction by means of the identifier.

User data:

Depending on the channel selected by the identifier, the user data have a different meaning.

## 7. Seizure of the identifiers

| Channel                         | ID<br>(decimal)       | ID<br>(hexadecimal)   |
|---------------------------------|-----------------------|-----------------------|
| Network management              | 0                     | 0                     |
| Sync telegram                   | 128                   | 80H                   |
| Process data channel fr. device | 384 + device address  | 180H + device address |
| Parameter channel from device   | 1408 + device address | 580H + device address |
| Parameter channel to device     | 1536 + device address | 600H + device address |

## 8. Network management

For switching between the 3 possible communication states INITIALISATION, PREOPERATIONAL and OPERATIONAL, telegrams with the identifier 0 and 2 bytes user data are used.

### 8.1 INITIALIZATION

In this state the IF09C is not involved in the processes on the bus. All hardware and software components are being initialized. This status is achieved after switching the device on or after receiving the command code 82H and one's own or the global address. After completion of initialization, the device changes automatically to the state PREOPERATIONAL.

### 8.2 PREOPERATIONAL (before ready)

The exchange of parameterization data between the IF09C and the bus master is enabled. There is, however, no transmission of process data.

### 8.3 OPERATIONAL (ready)

The exchange of process and parameterization data is enabled.

## 8.4 Switching between communication states

Switching between communication states can be induced by the network master through sending the following telegrams with the identifier 0.

| Change of state |                   | DATA 1 | DATA 2 |
|-----------------|-------------------|--------|--------|
| from            | to                |        |        |
| PRE-OPERATIONAL | OPERATIONAL       | 01H    | xx     |
| OPERATIONAL     | PRE-OPERATIONAL   | 80H    | xx     |
| OPERATIONAL     | INITIALIZATION    | 81H    | xx     |
| PRE-OPERATIONAL | INITIALIZATION    | 81H    | xx     |
| OPERATIONAL     | INITIALIZATION(*) | 82H    | xx     |
| PRE-OPERATIONAL | INITIALIZATION(*) | 82H    | xx     |

(\*) cold start is induced.

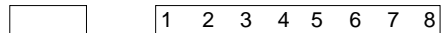
xx = 0: the telegram applies to all devices on the bus.

xx = device address: the telegram applies only to the device with the corresponding address.

## 9. Parameterization

Setup of the CAN telegram

11 bits identifier up to 8 bytes user data



Byte 1: Command code

| Command<br>(service) | Command<br>code<br>(hexadecimal) | Command<br>code<br>(decimal) | Description                                 |
|----------------------|----------------------------------|------------------------------|---------------------------------------------|
| Write Request        | 23H                              | 35                           | Send parameter to IF09C                     |
| Write                | 60H                              | 96                           | Response of IF09C Response to Write Request |
| Read Request         | 40H                              | 64                           | Request of a parameter from IF09C           |
| Read Response        | 42H                              | 66                           | Response to request with current value      |
| Error Response       | 80H                              | 128                          | Error message                               |

### Bytes 2,3: Parameter index

The parameter index is entered in the user data byte 2 (low byte) and in the user data byte 3 (high byte) in the Intel data format. For a description of the index table see chapter 'Index Table'.

#### Byte 4: Sub-index

In the sub-index the sensor number of the sensor to be addressed is entered. The value range is 1..31.

#### Bytes 5..8: Data 1..Data 4

In the data domain the value of the parameter is entered in the left-aligned Intel format. (byte 5 = low byte .. byte 8 = high byte)

#### 9.1 Error codes

Error code = 128 = 80H

In the case of a communication error, the IF09C sends an error response In the user data byte 8 (data 4) a 6 is always entered, and in the user data byte 7 (data 3) an error message is entered.

Possible error codes

| Command code | Data 3 | Data 4 | Description                                                               |
|--------------|--------|--------|---------------------------------------------------------------------------|
| 80H          | 6      | 6      | Wrong index                                                               |
| 80H          | 5      | 6      | Wrong sub-index                                                           |
| 80H          | 3      | 6      | Access refused, e.g. write request on read variable or parameter missing. |

#### 9.2 Example for Write Parameter

The IF09C has the device address 1. An Offset of 40000 is to be added to sensor no. 5.

Calculation of the identifier:

Identifier of the parameter channel to the device = 1536 + device address

Identifier = 1536 + 1 = 1537 = 601H

Command code = Write Request (send parameter to the device) = 23H

Index = 5F02H

The index for the parameter offset was taken over from the index table in chapter 11 'index table'.

Sub-index = sensor number 5 = 5

Parameter value = 40000 = 9c40H

Telegram to the IF09C :

Identifier

| Identifier | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|
| 601H       | 23H | 02H | 5FH | 05H | 40H | 9CH | 00H | 00H |

Response of the IF09C in error-free execution:

| Identifier | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|
| 581H       | 60H | 02H | 5FH | 05H | 00H | 00H | 00H | 00H |

Command code = Write Response (response to the Write Request) = 60H

Calculation of the identifier:

Identifier of the parameter channel to the device = 1408 + device address

Identifier = 1408 + 1 = 1409 = 581H

#### 9.3 Example for Read Parameter

The IF09C has device address 1. The angle value of the sensor no. 3 is to be read.

Calculation of the identifier:

Identifier of the parameter channel to the device = 1536 + device address

Identifier = 1536 + 1 = 1537 = 601H

Command code = Read Request (request to read a parameter) = 40H

Index = 5F00H

The index for the angle value parameter was taken over from the index table in the chapter 'Index Table'.

Sub-index = Sensor number 3 = 3

Parameter value = 40000 = 9c40H

Telegram to the IF09C :

| Identifier | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|
| 601H       | 40H | 02H | 5FH | 03H | 00H | 00H | 00H | 00H |

Response of the IF09C with error-free execution:

| Identifier | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|
| 581H       | 42H | 02H | 5FH | 03H | 9CH | 40H | 00H | 00H |

Command code = Read Response (response to the Read Request with current value) = 42H

Calculation of the identifier:

Identifier of the parameter channel to the device = 1408 + device address

Identifier = 1408 + 1 = 1409 = 581H

### 10. Process data

The 'process data mode' parameter (index 5F08H) serves for specifying the mode of the process data exchange:

Process data mode = 0 (default): synchronous process data exchange

Process data mode = 1: cyclic process data exchange

Value range: 1 ms .. 65535 ms. The default value is 100 ms.

The process data channel is activated when the network master of the IF09C switches to the OPERATIONAL state.

### Setup of the PDOs (only CANopen protocol):

A process data channel is assigned to each connected encoder: The identifier of the encoder with the address 1 set to 180h+device address by default. The PDOs of all further encoders are calculated according to 180+device address +encoder address-1 (\*).

|            |     |     |     |     |     |     |     |     |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Identifier | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   |
| (*)        | LSB | NSB | NSB | MSB | 00H | 00H | 00H | 00H |

### 10.1 Synchronous process data exchange

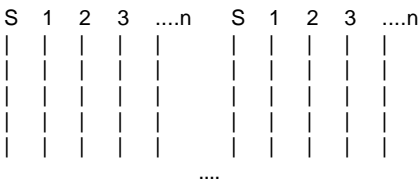
With the Sync telegram, the sending of the process data from all sensors is induced. This telegram contains no user data, it consists only of the identifier 80H. For cyclic process data transmission, the sync telegram must be repeatedly generated by the network master.

#### Scanning time:

Program flow in the IF09C :

The background program of the IF09C reads in cyclically the angle values of all connected sensors, it takes appr. 2 ms for the transducer to do this (AP09). The scanning time is, therefore 2 ms x number of connected sensors.

The following schema illustrates the synchronous process data exchange:



S: sync telegram

1: PDO rx for sensor 1

2: PDO rx for sensor 2

3: PDO rx for sensor 3

n: PDO rx for sensor n

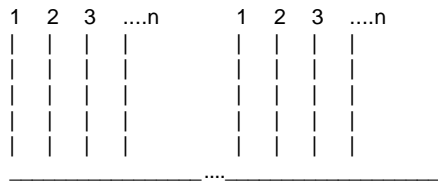
n = number of connected sensors

All further telegrams, e.g. parameter transfer, are taken over asynchronously after transfer from the IF09C. The asynchronous data is ignored in the above representation.

### 10.2 Cyclic process data exchange

In cyclic process data exchange, the actual angle values (process data PDO) of all connected sensors are transmitted in defined intervals by the IF09C in the stand-alone mode. The intervals are determined in the process cycle time variable.

The following schema illustrates the cyclic process data exchange:



1: PDO rx for sensor 1

2: PDO rx for sensor 2

3: PDO rx for sensor 3

n: PDO rx for sensor n

n = number of connected sensors

All further telegrams, e.g. parameter transfer, are taken over asynchronously after transfer from the IF09C. The asynchronous data is ignored in the above representation.

### 10.3 Disable encoder PDO (only CANopen protocol)

The transfer of the process data (i.e. of the angle value) can be disabled for each connected encoder by setting the assigned disable bit. This means that in synchronous and cyclic process data exchange no angle values are transferred from all encoders whose enable bit is set. By default, after power-up or the NMT command INITIALIZATION all disable bits are deleted.

|            |     |     |     |     |    |    |    |    |
|------------|-----|-----|-----|-----|----|----|----|----|
| Identifier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
| 581H       | 23H | 0BH | 5FH | 00H | b0 | b1 | b2 | b3 |

b0: Disable bits for encoders with addresses 1..8

b1: Disable bits for encoders with addresses 9..16

b2: Disable bits for encoders with addresses 17..24

b3: Disable bits for encoders with addresses 25..31

Example: disable encoders with the addresses 1, 8 and 10, enable all other encoders.

|      |     |     |     |     |     |     |     |     |
|------|-----|-----|-----|-----|-----|-----|-----|-----|
| 581H | 23H | 0BH | 5FH | 00H | 81h | 02h | 00h | 00h |
|------|-----|-----|-----|-----|-----|-----|-----|-----|

The disable bits are traceable (see example read parameter)

Assignment of encoder numbers:

During power-up and the NMT command INITIALIZATION of the IF09C all encoders connected to the RS485 interface are identified and entered into a list, together with the indexes of their addresses.

### 10.4 Key status (only CANopen protocol)

The status of the operator key of each connected AP09 encoder is cyclically read out by the gateway. By pressing the key, a bit assigned to each encoder address is set. Each set key status bit is reset when the respective SDO 5F0CH is read. During power-up all key status bits are reset as well.



|            |     |     |     |     |    |    |    |    |
|------------|-----|-----|-----|-----|----|----|----|----|
| Identifier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
| 5F0CH      | 23H | 0CH | 5FH | 00H | b0 | b1 | b2 | b3 |

b0: Key status bits for AP09 encoders with the addresses 1..8

b1: Key status bits for AP09 encoders with the addresses 9..16

b2: Key status bits for AP09 encoders with the addresses 17..24

b3: Key status bits for AP09 encoders with the addresses 25..31

Example: Keys of the encoders with the addresses 1,8 and 10 were pressed.

|       |     |     |     |     |     |     |     |     |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| 5F0CH | 23H | 0CH | 5FH | 00H | 81h | 02h | 00h | 00h |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|

### 10.5 Online status (only CANopen protocol)

During power-up the gateway checks at which addresses encoders are connected. The gateway stores this information as a bit field in an online list. A bit is assigned to each encoder address. This bit is set after identification of a connected encoder with this address.

|            |     |     |     |     |    |    |    |    |
|------------|-----|-----|-----|-----|----|----|----|----|
| Identifier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
| 5F0DH      | 23H | 0CH | 5FH | 00H | b0 | b1 | b2 | b3 |

b0: Online bits for AP09 encoders with the addresses 1..8

b1: Online bits for AP09 encoders with the addresses 9..16

b2: Online bits for AP09 encoders with the addresses 17..24

b3: Online bits for AP09 encoders with the addresses 25..31

Example: Encoders with the addresses 1,8 and 10 are connected.

|       |     |     |     |     |     |     |     |     |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| 5F0CH | 23H | 0CH | 5FH | 00H | 81h | 02h | 00h | 00h |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|

### 10.6 Reset CAN gateway (only CANopen protocol)

Writing a value > 00 00 00 00 on the SDO 5F0EH induces the reset of the gateway. All functions are executed exactly the same way as with power up and all settings are set to the default values.

Example: Execute gateway reset

|            |     |     |     |     |    |    |    |    |
|------------|-----|-----|-----|-----|----|----|----|----|
| Identifier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
| 5F0EH      | 40H | 0EH | 5FH | 00H | 01 | 00 | 00 | 00 |

### 10.7 CAN gateway status (only CANopen protocol)

Via the SDO 5F0FH the current operational mode of the gateway can be read out.

|            |     |     |     |     |    |    |    |    |
|------------|-----|-----|-----|-----|----|----|----|----|
| Identifier | 1   | 2   | 3   | 4   | 5  | 6  | 7  | 8  |
| 5F0FH      | 23H | 0FH | 5FH | 00H | B0 | 00 | 00 | 00 |

b0: 00H status initialization

7FH status pre-operational

05H status operational

Example: Current status: pre-operational.

|       |     |     |     |     |     |     |     |     |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|
| 5F0FH | 42H | 0FH | 5FH | 00H | 7Fh | 00h | 00h | 00h |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|

## 11. Index table

| Parameter                  | Index | Subindex | Data type  | Data length | Access |
|----------------------------|-------|----------|------------|-------------|--------|
| Angle value                | 5F00H | 1 ... 31 | Integer 32 | 4           | R      |
| Calibration value          | 5F01H | 1 ... 31 | Integer 32 | 4           | R/W    |
| Offset (*1)                | 5F02H | 1 ... 31 | Integer 32 | 4           | **R/W  |
| Device identification      | 5F03H | 0        | Integer 32 | 4           | R      |
| Configuration/status       | 5F04H | 1 ... 31 | Integer 32 | 4           | R/W    |
| Display per revolution     | 5F05H | 1 ... 31 | Integer 32 | 4           | R/W    |
| System status              | 5F06H | 0        | Integer 32 | 4           | R/W    |
| Software calibration/reset | 5F07H | 1 ... 31 | Integer 32 | 4           | W(*2)  |
| Process data mode          | 5F08H | 0        | Integer 32 | 4           | R/W    |
| Process data cycle time    | 5F09H | 0        | Integer 32 | 4           | R/W    |
| Desired value              | 5F0AH | 1 ... 31 | Integer 32 | 4           | W      |
| **Disable encoder PDO(*3)  | 5F0BH | 0        | Integer 32 | 4           | R/W    |
| **Key status(*4)           | 5F0CH | 0        | Integer 32 | 4           | R      |
| **Online status(*5)        | 5F0DH | 0        | Integer 32 | 4           | R      |
| **Reset CAN gateway        | 5F0EH | 0        | Integer 32 | 4           | W      |
| **Status CAN gateway       | 5F0FH | 0        | Integer 32 | 4           | R      |

\*\* Only for CANopen protocol

(\*1) This parameter is not available for AP09 rotary encoders. Access attempt results in the error code response 'access error'.

(\*2) The contents of the user data bytes are insignificant.

(\*3) A bit is assigned to each connected encoder (see section 10.3)

(\*4) A bit is assigned to each connected encoder (see section 10.4)

(\*5) A bit is assigned to each connected encoder (see section 10.5)

Sub-index 0 means that this parameter concerns the IF09C itself.

Sub-index 1..31 addresses the connected encoder with the corresponding address.

Access = R: only read access is possible.  
 Access = W: only write access is possible.  
 Access = R/W: both read and write access are possible.

All values are represented in the Intel format.

### 11.1 Device identification parameter

provides in the

1st data byte: 7 = device type of the IF09C,  
 2nd data byte: version number of the IF09C (e.g. V1.01 = 11H),  
 all other data bytes contain 0.

### 11.2 Configuration/status parameter

provides with the AP09:

the 1st contains the following information:

| Bit no. | IF09C (CAN bus)<br>-> AP09                                                                               | AP09->IF09C<br>(CAN bus)                                                                                            |
|---------|----------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------|
| 7(MSB)  | no function                                                                                              | Bit =1: battery empty                                                                                               |
| 6       | free                                                                                                     | -standard protocol free<br>-CANopen protocol key status: is set to 1 upon pressing the key and reset during readout |
| 5-4     | key mode<br>00: no key function<br>01: incremental measure<br>10: reset<br>11: display of required value | key mode<br>00: no key function<br>01: incremental measure<br>10: reset<br>11: display of required value            |
| 3       | reset                                                                                                    | no function                                                                                                         |
| 2       | execute incremental measure                                                                              | no function                                                                                                         |

| 1      | free                                                                       | free                                                                       |
|--------|----------------------------------------------------------------------------|----------------------------------------------------------------------------|
| 0(LSB) | direction of rotation:<br>Bit = 0: counter-clockwise<br>Bit = 1: clockwise | direction of rotation:<br>Bit = 0: counter-clockwise<br>Bit = 1: clockwise |

2nd data byte: number of digits after the decimal point

3rd data byte: version number

### 11.3 System Status Parameter

provides in the

1st data byte: 0: IF09C not ready (ongoing sensor identification)  
 1: IF09C ready.

2nd data byte: number of connected sensors (0..31)

## 12. Communication-Profile Objects acc. to encoder profile 406 (only CANopen protocol)

### 12.1 Device Type

| Index       | 1000h                      |
|-------------|----------------------------|
| Name        | device type                |
| Object Code | VAR                        |
| Data Type   | Unsigned32                 |
| Access      | Read_only                  |
| Value       | 0x96, 0x01, 0x00, 0x00 (*) |

(\*) For this device the value 10 would have to be entered in the 3rd byte for multi-sensor device according to the device profile for encoder 406. However, this byte was set to 0x00 to guarantee for keeping up during device identification since not all configuration tools can recognize this type.

### 12.2 Error Register

| Index       | 1001h                  |
|-------------|------------------------|
| Name        | Error register         |
| Object Code | VAR                    |
| Data Type   | Unsigned32             |
| Access      | Read-only              |
| Value       | 0x00, 0x00, 0x00, 0x00 |

### 12.3 COB-ID Sync Message

| Index     | 1005h                  |
|-----------|------------------------|
| Name      | COB-ID SYNC            |
| Access    | Read-only              |
| Data Type | Unsigned 32            |
| Value     | 0x80, 0x00, 0x00, 0x00 |

## 12.4 Node-ID (node address)

|              |                         |
|--------------|-------------------------|
| <b>Index</b> | <b>100Bh</b>            |
| Name         | COB-ID SYNC             |
| Access       | Read-only               |
| Data Type    | Unsigned32              |
| Value        | setup of DIP switch SW2 |

## 12.5 Guard Time

|               |              |
|---------------|--------------|
| <b>Index</b>  | <b>100Ch</b> |
| Name          | COB-ID SYNC  |
| Access        | Read-only    |
| Data Type     | Unsigned32   |
| Default-Value | 300 ms       |

## 12.6 Identity Object

|              |                |
|--------------|----------------|
| <b>Index</b> | <b>1018h</b>   |
| Name         | Error register |
| Object Code  | VAR            |
| Data Type    | Unsigned32     |

| Sub-Index | Description       | Access    | Value               |
|-----------|-------------------|-----------|---------------------|
| 00h       | number of entries | Read-only | 3                   |
| 01h       | Vendor ID         | Read-only | 0x55,0xAA,0x00,0xFF |
| 02h       | Product Code      | Read-only | 0x07,0x00,0x00,0x00 |
| 03h       | Revision Number   | Read-only | 0x00,0x20,0x00,0x20 |

## 12.7 Transmit PDO objects

|              |                                 |
|--------------|---------------------------------|
| <b>Index</b> | <b>1800h+(Encoderaddress-1)</b> |
| Name         | Transmit PDO parameter          |
| Object Code  | RECORD                          |
| Data Type    | PDO CommPar                     |

| Sub-Index | Description         | Access     | (Default)-Value                   |
|-----------|---------------------|------------|-----------------------------------|
| 00h       | number of entries   | Read-only  | 31                                |
| 01h       | COB-ID used by PDO1 | Read/Write | ID=date+device ID+encoder-address |
| 02h       | Transmission type   | Read-only  | 0xFE,0x00,0x00,0x00               |

(device ID of switch SW2)

|              |                                 |
|--------------|---------------------------------|
| <b>Index</b> | <b>1A00h+(Encoderaddress-1)</b> |
| Name         | Server SDO parameter            |
| Object Code  | RECORD                          |
| Data Type    | PDO Mapping                     |

| Sub-Index | Description         | Access    | Value                 |
|-----------|---------------------|-----------|-----------------------|
| 00h       | number of entries   | Read-only | 2                     |
| 01h       | COB-ID used by PDO1 | Read-only | (0x80,0x01,0x00,0x00) |
| 02h       | Transmission type   | Read-only | 0x01,0x00,0x00,0x00   |

## 13. Service Data Objects (SDO) acc. to device profile for encoder 406, version 2.0 (only CANopen protocol)

### 13.1 Offset (Preset\_Value)

|              |                                      |
|--------------|--------------------------------------|
| <b>Index</b> | <b>6010h</b>                         |
| Name         | Preset_value_for multisensor_devices |
| Object Code  | ARRAY                                |
| Data Type    | Unsigned8                            |

| Sub-Index | Description                   | Access     | ValueRange |
|-----------|-------------------------------|------------|------------|
| 00h       | Number_of_available_channels  | Read/write | 00..31(*)  |
| 01h       | Preset_value_channel_1(*)     | Read/write | Integer32  |
| 02h       | Preset_value_channel_2(*)     | Read/write | Integer32  |
| 03h..1eh  | Preset_value_channel_3..30(*) | Read/write | Integer32  |
| 1fh       | Preset_value_channel_31(*)    | Read/write | Integer32  |

(\*) depending on the number of connected encoders.

### 13.2 Angle value (Position\_value)

|              |                                        |
|--------------|----------------------------------------|
| <b>Index</b> | <b>6020h</b>                           |
| Name         | Position_value_for multisensor_devices |
| Object Code  | ARRAY                                  |
| Data Type    | Unsigned8                              |

| Sub-Index | Description                     | Access    | ValueRange |
|-----------|---------------------------------|-----------|------------|
| 00h       | Number_of_available_channels    | Read-only | 00..31(*)  |
| 01h       | Position_value_channel_1(*)     | Read-only | Integer32  |
| 02h       | Position_value_channel_2(*)     | Read-only | Integer32  |
| 03h..1eh  | Position_value_channel_3..30(*) | Read-only | Integer32  |
| 1fh       | Position_value_channel_31(*)    | Read-only | Integer32  |

(\*) depending on the number of connected encoders.

### 13.3 Cyclic Timer

|               |                         |
|---------------|-------------------------|
| <b>Index</b>  | <b>6200h</b>            |
| Name          | Cyclic Timer            |
| Object Code   | Var                     |
| Data Type     | Unsigned16              |
| Access        | Read/write              |
| Value Range   | Unsigned16 (0..65535ms) |
| Default Value | 100 (100 ms)            |

Cycle time for transferring the process data in the 'cyclic process data transfer' mode (the attainable minimal cycle time is limited by the number of connected encoders and their communication time with the gateway via the RS485-interface).

### 14. EDS file (only CANopen protocol)

For the IF09C the EDS file 'SIKOCGW.EDS' (electronic-data-sheet) is available, which is delivered together with the assembly. This file enables easy integration and configuration of the IF09C in a CANopen network by means of commercial CANopen configurators.

**SIKO** GmbH  
DR.-ING. G. WANDRES

Postanschrift / Postal address:  
Postfach 1106  
D-79195 Kirchzarten

Werk / Factory:  
Weihermattenweg 2  
D-79256 Buchenbach

Telefon / Phone 0 76 61 / 3 94 - 0  
Telefax / Fax 0 76 61 / 3 94 - 388  
Internet [www.siko.de](http://www.siko.de)

