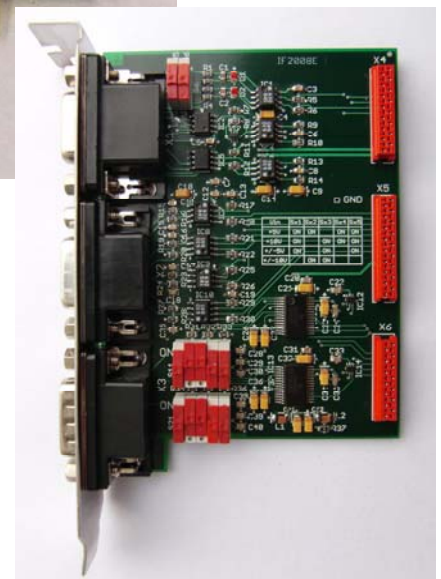
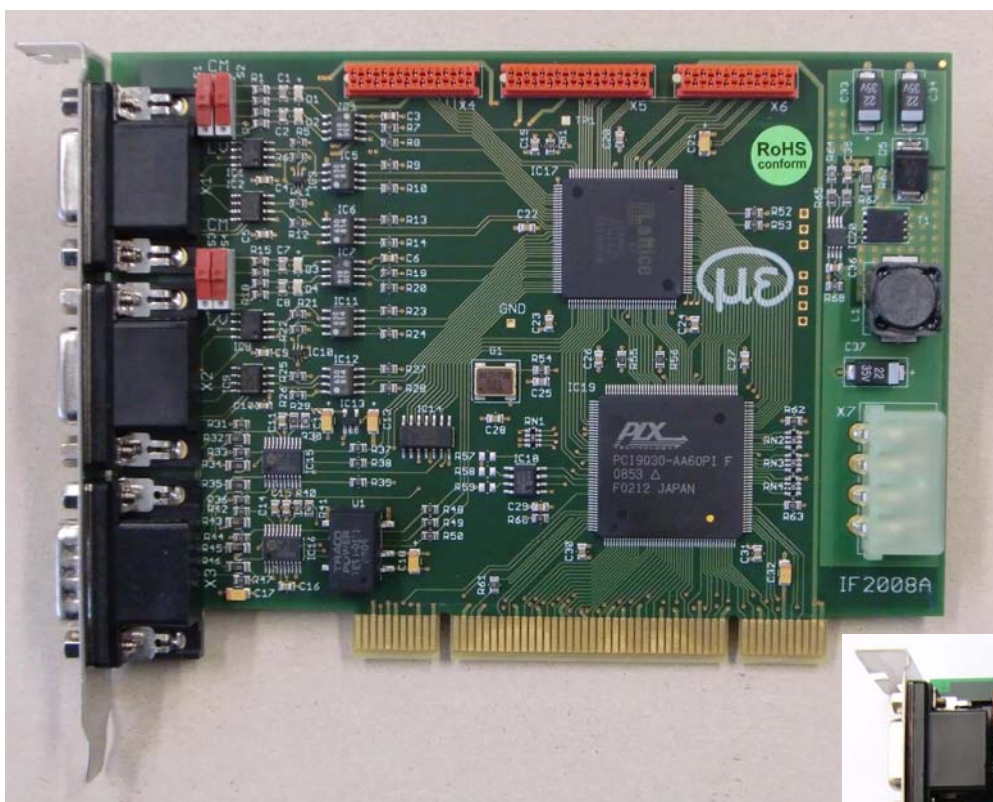


## Beschreibung

**IF2008A** PCI-Basiskarte  
**IF2008E** Option Board



# Inhaltsverzeichnis

<b>1</b>	<b>Technische Daten</b> .....	<b>4</b>
1.1	IF2008A Basisplatine .....	4
1.2	IF2008E Option Board .....	5
<b>2</b>	<b>Hardware</b> .....	<b>6</b>
2.1	Ansicht IF2008A.....	6
2.2	Ansicht IF2008E.....	7
<b>3</b>	<b>Steckerbelegung und Jumperstellung</b> .....	<b>8</b>
3.1	Sensor-Interface (IF2008A X1 und X2, IF2008E X1).....	8
3.2	Encoder-Interface (IF2008A X3) .....	8
3.3	Sensor-Power (IF2008A X7) .....	9
3.4	I/O-Interface (IF2008E X2).....	9
3.5	Analog-Interface (IF2008E X3) .....	9
3.6	Jumper-/Schalterstellung für Trigger-Level .....	10
3.7	Schalterstellung für ADC-Level .....	10
<b>4</b>	<b>Adressbelegung</b> .....	<b>11</b>
4.1	PCI-Interface .....	11
4.2	Lokale Adress-Belegung .....	11
<b>5</b>	<b>Register-Beschreibung</b> .....	<b>12</b>
5.1	Sende-Register .....	12
5.2	FIFO-Daten .....	12
5.3	Set- / Reset- / Latch-Register.....	13
5.4	FIFO-Volumen.....	13
5.5	FIFO-Enable-Register .....	14
5.6	Interrupt-Enable-Register .....	15
5.7	Interrupt-Status-Register.....	15
5.8	Sensor Baud-Rate.....	16
5.9	Zähler-Kontrollregister.....	16
5.10	Zähler Preload.....	18
5.11	Zählerwert .....	18
5.12	Timer .....	19
5.13	ADC.....	20
5.14	Status .....	21
5.15	Input .....	21
5.16	Output-Register .....	22
5.17	Mode Opto- und TxD-Ausgänge .....	23
5.18	Mode Trigger-Ausgänge .....	24
5.19	ADC-Kontrollregister .....	26
5.20	Parity-Enable-Register .....	27
5.21	Parity-Error-Register .....	27
<b>6</b>	<b>Verdrahtungsempfehlung</b> .....	<b>28</b>
6.1	Sensor ILD1302 und ILD1402.....	28
6.2	Sensor ILD1700 .....	28
6.3	Sensor ILD2200 .....	29
6.4	Sensor ILD2300 .....	29
6.5	Encoder-Interface.....	30
6.6	Optokoppler I/O.....	31
<b>7</b>	<b>Beispiele für Synchronisieren, Triggerung, Torung</b> .....	<b>32</b>
7.1	Einen Messwert vom Sensor holen, Software-Trigger .....	32

7.2	Hardware-Triggerung .....	32
7.3	Softwaretorung (Gate) am Sensor .....	32
7.4	Hardwaretorung (Gate) am Sensor .....	32
7.5	Hardwaretorung (Gate) mit IF2008 .....	33
7.6	Synchrones Messwerteinlesen mit Encoder und IF2008 .....	33

# 1 Technische Daten

## 1.1 IF2008A Basisplatine

### Mechanik und Umgebung

- Abmessungen (Leiterplattenmaße) ca. 140 x 102 mm, 1 Slot breit
- maximal zulässige Umgebungstemperatur +50 °C
- zwei D-Sub Buchsenleisten HD 15-polig für Sensoranschlüsse
- eine D-Sub Stiftleiste HD 15-polig für Encoder-Signale
- ein Tyco/AMP Commercial MATE-N-LOK Stecker (IDE-Festplattenstecker) für Versorgung DC-/DC-Wandler
- drei Buchsenleisten Tyco/AMP MicroMatch für Verbindung zur IF2008E

### PCI-Bus

- PCI-Stecker 3,3 oder 5 Volt 32-Bit 2x60 Pin
- Target Interface (Slave) nach Spezifikation Rev. 2.1 und 2.2
- Bus-Taktfrequenz max. 40 MHz
- Stromaufnahme an +5 Volt ca. 0,5 A, ohne Sensoren und Encoder

### Sensor-Interface (X1 / X2)

- 2 RS422-Driver und zwei RS422-Receiver mit galvanischer Trennung pro Stecker (Ein- / Ausgangsfrequenz max. 5 MHz)
- 2 LVDS bzw. 3,3 Volt CMOS-Ausgänge mit galvanischer Trennung pro Stecker (Ausgangsfrequenz max. 5 MHz)
- Spannungsversorgung der Sensoren mit 24 V

### Encoder-Interface (X3)

- Interface für zwei Encoder mit 1  $V_{SS}$ , RS422- (Differenz-) oder TTL- (single-ended) Signalen
- Spannungsversorgung der Encoder mit +5 V aus PCI-Versorgung ohne galvanische Trennung (Stromaufnahme abhängig von den angeschlossenen Encodern)
- Interpolation programmierbar von 1- bis 64-fach bei Encodern mit 1  $V_{SS}$ -Signalen (Eingangsfrequenz max. = [3,2 MHz / Interpolation] ≤ 800 kHz)
- Auswertung programmierbar von 1- bis 4-fach bei Encodern mit:
  - RS422- / Differenz-Signalen (Eingangsfrequenz max. = 800 kHz)
  - TTL- / Single-Ended-Signalen (Eingangsfrequenz max. = 400 kHz)

### DC-/DC-Wandler

- Eingangsspannungsbereich 12 V ±1,0 V
- Ausgangsspannung 24 V ±0,5 V
- Ausgangsstrom max. 1,25 A für alle Sensoren gemeinsam
- Effizienz typisch 90 %

Die Versorgung des DC-DC-Wandlers erfolgt durch das Netzteil im PC. Die Verbindung zwischen PC-Netzteil und IF2008A ist bei der Installation der Karte herzustellen.

## 1.2 IF2008E Option Board

### **Mechanik und Umgebung**

- Abmessungen (Leiterplattenmaße) ca. 71 x 102 mm, 1 Slot breit
- maximal zulässige Umgebungstemperatur +50 °C
- eine D-Sub Buchsenleiste HD 15-polig für Sensoranschlüsse
- eine D-Sub Buchsenleiste 9-polig für I/O-Interface
- eine D-Sub Stiftleiste 9-polig für Analogeingänge
- drei Buchsenleisten MicroMatch für Verbindung zur IF2008A

### **Sensor-Interface (X1)**

- Identisch mit IF2008A (X1)

### **I/O-Interface (X2)**

- 4 Optokoppler-Eingänge, Eingangsstrom max. 5 mA, Eingangsfrequenz max. 1 MHz
- 4 Optokoppler-Ausgänge, Ausgangsstrom max. 20 mA, Ausgangsfrequenz max. 1 MHz

### **Analog-Interface (X3)**

- zwei ADC-Kanäle
- Eingangsspannungsbereich 0-5 V, 0-10 V,  $\pm 5$  V,  $\pm 10$  V getrennt einstellbar für jeden Kanal über DIP-Schalter
- Auflösung 16 Bit
- Offsetfehler max.  $\pm 3$  mV
- Verstärkungsfehler max.  $\pm 5$  mV
- Wandlungsrate max. 150 kHz pro Kanal

## 2 Hardware

### 2.1 Ansicht IF2008A

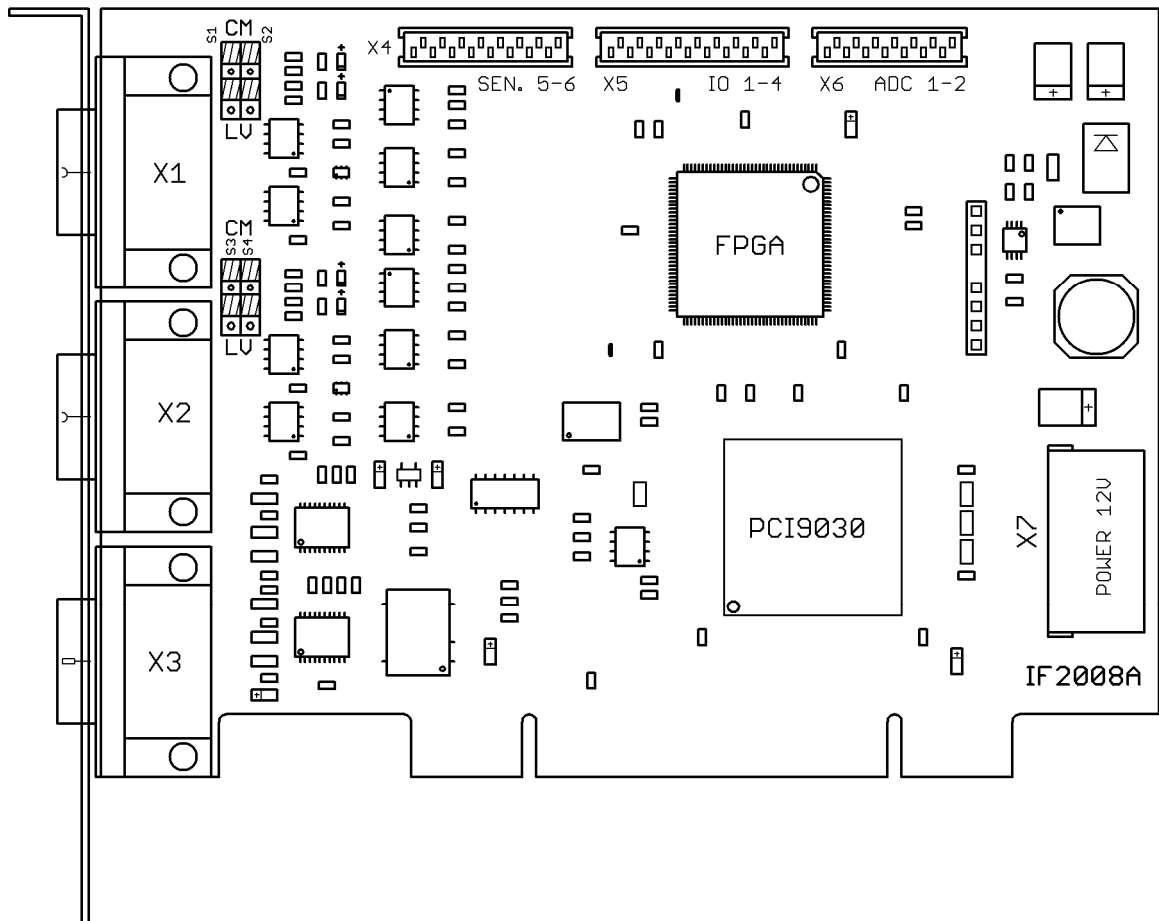


Bild 1: Platinenansicht IF2008A

- X1 = Anschluss für Sensor 1 und 2
- X2 = Anschluss für Sensor 3 und 4
- X3 = Anschluss für Encoder 1 und 2
- X4 ... X6 = Anschluss für Verbindung zur IF2008E
- X7 = Anschluss 12 V-Power, Verbindung zum PC-Netzteil erforderlich
- S1 .. S4 = Schalter für positiven Trigger-Level

## 2.2 Ansicht IF2008E

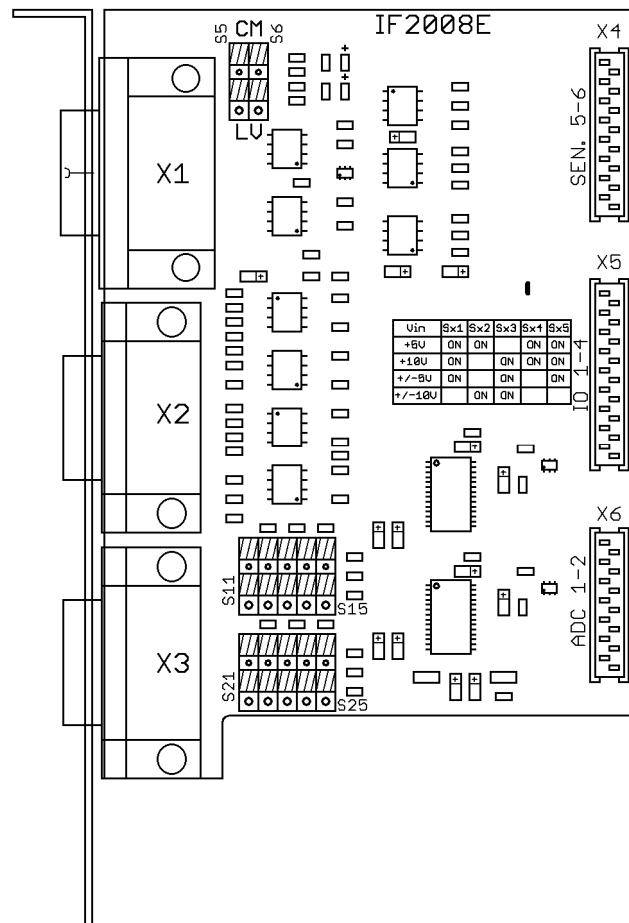


Bild 2: Platinenansicht IF2008E

- X1 = Anschluss für Sensor 5 und 6
- X2 = Anschluss für I/O-Signale
- X3 = Anschluss für Analog-Digital-Converter
- X4 ... X6 = Anschluss für Verbindung zur IF2008A
- S5 u. S6 = Schalter für positiven Trigger-Level
- S11 ... S15 = Schalter für ADC-Level 1
- S21 ... S25 = Schalter für ADC-Level 2

### 3 Steckerbelegung und Jumperstellung

#### 3.1 Sensor-Interface (IF2008A X1 und X2, IF2008E X1)

Pin	Signal
1	Sensor 1 TxD-
2	Sensor 1 TxD+
3	Sensor 1 RxD-
4	Sensor 1 RxD+
5	Spannungsversorgung 0V
6	Sensor 1 TRG+
7	Sensor 1 TRG-
8	Sensor 2 TRG+
9	Sensor 2 TRG-
10	Spannungsversorgung +24V
11	Sensor 2 TxD-
12	Sensor 2 TxD+
13	Sensor 2 RxD-
14	Sensor 2 RxD+
15	GND* (galvanisch getrennt zu PC-GND)

Tabelle 1: Steckerbelegung Sensor-Interface

#### 3.2 Encoder-Interface (IF2008A X3)

Pin	Funktion
1	Encoder 1 Spur A+
2	Encoder 1 Spur A-
3	Encoder 2 Spur A+
4	Encoder 2 Spur A-
5	VCC (+5V)
6	Encoder 1 Spur B+
7	Encoder 1 Spur B-
8	Encoder 2 Spur B+
9	Encoder 2 Spur B-
10	GND
11	Encoder 1 Spur R+
12	Encoder 1 Spur R-
13	Encoder 2 Spur R+
14	Encoder 2 Spur R-
15	GND

Tabelle 2: Steckerbelegung Encoder-Interface

**Achtung:** Die Steckerbelegung ist mit der IF2004B **nicht** kompatibel!



### 3.3 Sensor-Power (IF2008A X7)

Pin	Funktion
1	+12 V
2	GND
3	GND
4	NC

Tabelle 3: Steckerbelegung Sensor-Power

### 3.4 I/O-Interface (IF2008E X2)

Pin	Funktion
1	OUT 1
2	OUT 2
3	OUT 3
4	OUT 4
5	GND (galvanisch getrennt zu PC-GND)
6	IN 1
7	IN 2
8	IN 3
9	IN 4

Tabelle 4: Steckerbelegung I/O-Interface

### 3.5 Analog-Interface (IF2008E X3)

Pin	Funktion
1	Eingangssignal 1
2	Analog GND
3	Eingangssignal 2
4	Analog GND
5	NC
6	NC
7	NC
8	NC
9	NC

Tabelle 5: Steckerbelegung Analog-Interface

### 3.6 Jumper-/Schalterstellung für Trigger-Level

Mit den Schaltern S1 bis S4 (IF2008A) beziehungsweise den Schaltern S5 und S6 (IF2008E) kann der positive Trigger-Level für die Sensorkanäle 1 bis 4 (IF2008A) beziehungsweise Sensorkanal 5 und 6 (IF2008E) selektiert werden. Der negative Ausgang hat immer LVDS-Level.

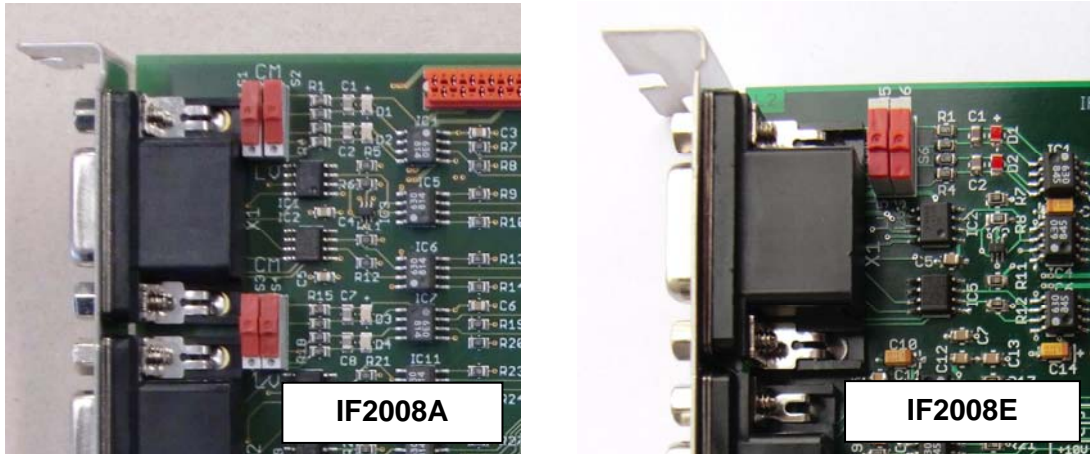


Bild 3: Schalterstellungen Trigger-Level IF2008A (links) und IF2008E (rechts)

Schalter	Stellung	Trigger-Ausgang +
S1 bis S6	CMn	3,3 V CMOS-Level für Sensor n TRG+
	LVn	LVDS-Level für Sensor n TRG+

Tabelle 6: Schalterstellungen Trigger-Level

### 3.7 Schalterstellung für ADC-Level

Mit den Schaltern S11 bis S15 beziehungsweise S21 bis S25 kann der Eingangsspannungsbereich des Analog-Digital-Wandlers für den Sensorkanal 5 und 6 auf der IF2008E selektiert werden.

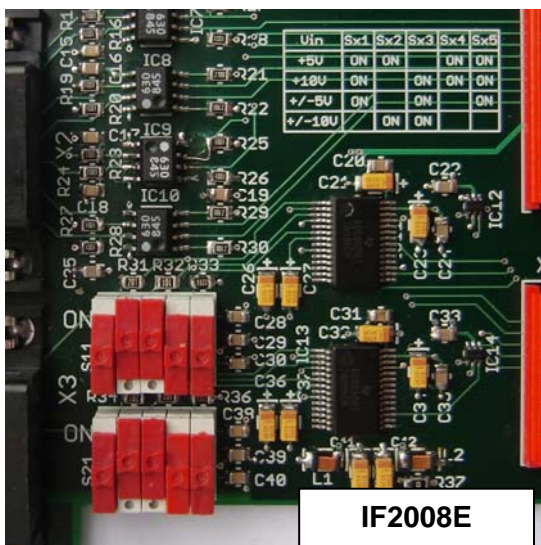


Bild 4: Schalterstellungen ADC-Level (im Bild sind ±10 V eingestellt)

VIN	Sx1	Sx2	Sx3	Sx4	Sx5
0-5 V	ON	ON		ON	ON
0-10 V	ON		ON	ON	ON
±5 V	ON		ON		ON
±10 V		ON	ON		

Tabelle 7: Schalterstellungen ADC-Level

Stellung
ON
OFF

## 4 Adressbelegung

### 4.1 PCI-Interface

Interface: 16-Bit PCI-Bus mit 3,3 oder 5-Volt-Anschluss

Zugriff: Memory-Space 40 Hex-Adressen

Basis-Adresse: Vergabe automatisch durch Betriebssystem

#### Header-Configuration

Adr.	Byte 3	Byte 2	Byte 1	Byte 0	Wert (Hex)
00h	Device ID		Vendor ID		9030 10B5
18h	Base Address Local Memory Space				xxxx xxxx
2C	Subsystem ID		Subsystem Vendor ID		2302 2810

Tabelle 8: Header-Configuration

### 4.2 Lokale Adress-Belegung

Basis-Adr. +	Schreibzugriff	Lesezugriff
00h	Sende-Register	FIFO-Daten
02h	Set- / Reset- / Latch-Register	FIFO-Volumen
04h	FIFO-Enable-Register	FIFO-Enable-Register
06h	Interrupt-Enable-Register	Interrupt-Status-Register
08h	Sensor 1 Baud-Rate	reserviert
0Ah	Sensor 2 Baud-Rate	reserviert
0Ch	Sensor 3 Baud-Rate	reserviert
0Eh	Sensor 4 Baud-Rate	reserviert
10h	Sensor 5 Baud-Rate	reserviert
12h	Sensor 6 Baud-Rate	reserviert
14h	Zähler-Kontrollregister 1	Zähler-Kontrollregister 1
16h	Zähler-Kontrollregister 2	Zähler-Kontrollregister 2
18h	Zähler 1 Preload LSW	Zählerwert 1 LSW
1Ah	Zähler 1 Preload MSW	Zählerwert 1 MSW
1Ch	Zähler 2 Preload LSW	Zählerwert 2 LSW
1Eh	Zähler 2 Preload MSW	Zählerwert 2 MSW
20h	Timer 1 Frequenz	ADC 1
22h	Timer 1 Pulsweite	ADC 2
24h	Timer 2 Frequenz	Status
26h	Timer 2 Pulsweite	Input
28h	Timer 3 Frequenz	reserviert
2Ah	Timer 3 Pulsweite	reserviert
2Ch	Timer Clock-Teiler	Timer Clock-Teiler
2Eh	Output-Register	Output-Register
30h	Mode Opto- und TxD-Ausgänge	Mode Opto- und TxD-Ausgänge
32h	Mode Trigger-Ausgänge	Mode Trigger-Ausgänge
34h	ADC-Kontrollregister	ADC-Kontrollregister
36h	Parity-Enable-Register	Parity-Error

Tabelle 9: Lokale Adress-Belegung

## 5 Register-Beschreibung

### 5.1 Sende-Register

Das Sende-Register schickt Befehle an den Sensor.

Basisadr. + 00h (Schreibzugriff)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			S6	S5	S4	S3	S2	S1	D7	D6	D5	D4	D3	D2	D1	D0
	Selektierung Sensor-Kanal							Daten-Bits								

Tabelle 10: Sende-Register

Bit 0 bis 7 beinhaltet die Daten für das Sende-Register

Bit 8 bis 15 selektieren den Sensor-Kanal

Bit 8 = 1 → Daten werden am Sensor-Kanal S1 ausgegeben

Bit 9 = 1 → Daten werden am Sensor-Kanal S2 ausgegeben

usw.

Bit 13 = 1 → Daten werden am Sensor-Kanal 6 ausgegeben

Bit 14..15 → frei

Unmittelbar nach einem Schreibzugriff auf die Adresse "0" werden die Daten zu dem unter Bit 8 bis 13 selektierten Sensor-Kanal übertragen. Die Baud-Rate für das Sende-Register wird automatisch dem selektierten Sensor-Kanal angepasst. Erfolgt die Datenausgabe gleichzeitig auf mehreren Kanälen, so wird die Baud-Rate des hochwertigsten Kanals verwendet.

### 5.2 FIFO-Daten

Antworten des Sensors, z.B. Messwerte, werden im FIFO-Speicher abgelegt und durch Funktionen der MEDAQLib an den Benutzer weitergeleitet.

Basisadr. + 00h (Lesezugriff)

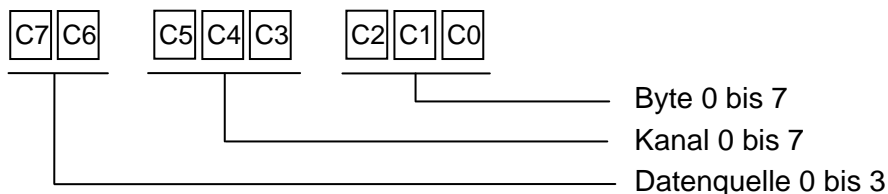
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C7	C6	C5	C4	C3	C2	C1	C0	D7	D6	D5	D4	D3	D2	D1	D0
	Code-Bits							Daten-Bits								

Tabelle 11: FIFO-Datenspeicher

Bit 0 bis 7 beinhaltet die gepufferten Daten

Bit 8 bis 15 kennzeichnen den Datencode

#### Code-Bits



C7	C6	Datenquelle
0	0	Sensor
0	1	Encoder
1	0	Schalteingang (IN 1..4 → Kanal 0, RxD 1..6 → Kanal 1)
1	1	ADC

Tabelle 12: FIFO-Datenspeicher – Datenquellen

### 5.3 Set- / Reset- / Latch-Register

Register zur Beeinflussung des Zählers.

Basisadr. + 02h (Schreibzugriff)

Bit	Funktion
0	Zähler 1 löschen, also Nullsetzen mit //“Clear_Encoder“
1	Zähler 1 laden, mit Wert vorbelegen durch //“SetEncoderPreload“ und mit //“Load_Encoder“ an IF2008 schicken.
2	Zähler 1 latchen, aktuellen Zählwert holen.
3	Zähler 1 referenzieren mit //“EnableRef_Encoder“, setzt Set_EncodeMode voraus
4	Zähler 2 löschen
5	Zähler 2 laden
6	Zähler 2 latchen
7	Zähler 2 referenzieren
8	ADC1 Konvertierung starten //“Get_ADCValue“ konvertiert und holt einen Wert
9	ADC2 Konvertierung starten
10	FIFO löschen mit //“Clear_Buffers“.
11 – 15	reserviert

Tabelle 13: Set- / Reset- / Latch-Register

#### Hinweis!

- Mit den Bits 0 bis 2 und 4 bis 6 können die Zähler unabhängig vom Zähler-Kontrollregister (Adr. 14h und Adr. 16h) per Software gelöscht, geladen oder der Zählerstand ins Latch-Register übernommen werden
- Wird mit dem Zähler-Kontrollregister (Adr. 14h und Adr. 16h) eine Zählerlatch- oder Lade-Funktion eingestellt die nur in Verbindung mit einer Referenzmarke ausgeführt werden soll, so muss diese vorher durch Setzen von Bit 3 bzw. Bit 7 freigegeben werden. Durch das Setzen von Bit 3 bzw. Bit 7 werden die Statusbits 0 und 1 bzw. 2 und 3 zurückgesetzt.
- Alle Bits müssen nur gesetzt werden, ein Rücksetzen ist nicht notwendig
- Nach einer Stromunterbrechung werden alle Bits auf “0” gesetzt
- //“ “ Name des entsprechenden Kommandos in der MEDAQLib.

### 5.4 FIFO-Volumen

Basisadr. + 02h (Lesezugriff)

Bit	Funktion
0 bis 11	FIFO-Datenvolumen (0 bis 4095)
12 bis 15	immer 0

Tabelle 14: FIFO-Volumen

Nach jedem Empfang eines Datensatzes wird dieser automatisch in den FIFO-Datenspeicher übertragen. Durch Abfrage des FIFO-Volumens kann die Datenfülle des FIFO ermittelt werden. Die Reihenfolge und Geschwindigkeit für das Puffern der empfangenen Daten ist identisch mit dem Datenstrom der Empfangs-Register. Wird der FIFO nicht schnell genug ausgelesen, so stehen in diesem die zuletzt empfangenen 4096 Datensätze.

Wird von der MEDAQLib intern genutzt; es gibt keinen separaten Befehl dazu.

## 5.5 FIFO-Enable-Register

Das FIFO-Enable-Register wird intern in der MEDAQLib behandelt.

Basisadr. + 04h (Schreib- und Lesezugriff)

Bit	Funktion
0	0 = FIFO für Sensor-Kanal 1 gesperrt 1 = FIFO für Sensor-Kanal 1 freigegeben
1	0 = FIFO für Sensor-Kanal 2 gesperrt 1 = FIFO für Sensor-Kanal 2 freigegeben
2	0 = FIFO für Sensor-Kanal 3 gesperrt 1 = FIFO für Sensor-Kanal 3 freigegeben
3	0 = FIFO für Sensor-Kanal 4 gesperrt 1 = FIFO für Sensor-Kanal 4 freigegeben
4	0 = FIFO für Sensor-Kanal 5 gesperrt 1 = FIFO für Sensor-Kanal 5 freigegeben
5	0 = FIFO für Sensor-Kanal 6 gesperrt 1 = FIFO für Sensor-Kanal 6 freigegeben
6	0 = FIFO für Encoder-Kanal 1 gesperrt 1 = FIFO für Encoder-Kanal 1 freigegeben
7	0 = FIFO für Encoder-Kanal 2 gesperrt 1 = FIFO für Encoder-Kanal 2 freigegeben
8	0 = FIFO für Status der externen Eingänge IN 1..4 gesperrt 1 = FIFO für Status der externen Eingänge IN 1..4 freigegeben
9	0 = FIFO für Status der RxD-Eingänge (Sensor 1..6) gesperrt 1 = FIFO für Status der RxD-Eingänge (Sensor 1..6) freigegeben
10	0 = FIFO für ADC 1 gesperrt 1 = FIFO für ADC 1 freigegeben
11	0 = FIFO für ADC 2 gesperrt 1 = FIFO für ADC 2 freigegeben
12	0 = FIFO wird bei aktivem, ext. Eingang IN 1 für Sensor 1 und 2 gesperrt 1 = IN 1 hat keinen Einfluss auf FIFO
13	0 = FIFO wird bei aktivem, ext. Eingang IN 2 für Sensor 3 bis 6 gesperrt 1 = IN 2 hat keinen Einfluss auf FIFO
14	0 = FIFO wird bei aktivem, ext. Eingang IN 3 für Encoder 1 und 2 gesperrt 1 = IN 3 hat keinen Einfluss auf FIFO
15	0 = FIFO wird bei aktivem, ext. Eingang IN 4 für ADC 1/2; IN 1..4; RxD 1..6 gesperrt 1 = IN 4 hat keinen Einfluss auf FIFO

Tabelle 15: FIFO-Enable-Register

Blockweises Datenlesen. Definiert, welche Daten, z. B. von den Sensoren, im FIFO gespeichert werden. Die Größe des FIFO beträgt 4095 Bytes. Ist 2/3 der Füllgröße des FIFO erreicht, liest der Treiber der IF-Karte die Daten des FIFO aus und speichert sie im Treiberpuffer mit max. 64 kByte. Die MEDAQLib holt die Daten aus dem Treiberpuffer ab und speichert sie in einem Ringpuffer mit bis zu 10 MByte.

Hinweis zu Bit 9: Die RxD-Eingänge können auch als weitere Steuereingänge, z. B. einer SPS, genutzt werden, falls die externen Eingänge (Bit 8) nicht ausreichen.

Hinweis zu Bit 12 bis 15: Damit ist eine Torung von Messdaten möglich. Der Befehl in der MEDAQLib lautet Use\_Gate.

## 5.6 Interrupt-Enable-Register

Basisadr. + 06h (Schreibzugriff)

Bit	Funktion
0	1 = Enable Interrupt-Anforderung wenn FIFO mehr als 50 % gefüllt ist
1	1 = Enable Interrupt- Anforderung wenn FIFO mehr als 75 % gefüllt ist
2	1 = Enable Interrupt- Anforderung bei Überlauf Timer 1
3	1 = Enable Interrupt- Anforderung bei Überlauf Timer 2
4	1 = Enable Interrupt- Anforderung bei Überlauf Timer 3
5	1 = Enable Interrupt- Anforderung wenn externer Eingang IN 1 aktiviert wird
6	1 = Enable Interrupt- Anforderung wenn externer Eingang IN 2 aktiviert wird
7	1 = Enable Interrupt- Anforderung wenn externer Eingang IN 3 aktiviert wird
8	1 = Enable Interrupt- Anforderung wenn externer Eingang IN 4 aktiviert wird
9 - 15	reserviert

Tabelle 16: Interrupt-Enable-Register

Die MEDAQLib nutzt Bit 1, also mehr als 75 % des FIFOs gefüllt ist. Der Interrupt veranlasst das Auslesen der Daten in den Treiberpuffer.

### Hinweis!

Die Interrupt-Generierung ist flankengetriggert, das heißt eine Interrupt-Anforderung erfolgt nur wenn im Interrupt-Enable-Register das entsprechende Bit gesetzt ist und die dazugehörige Quelle vom inaktiven in den aktiven Zustand wechselt. Es können gleichzeitig mehrere Bits gesetzt sein.

## 5.7 Interrupt-Status-Register

Basisadr. + 06h (Lesezugriff)

Bit	Funktion
0	1 = Interrupt-Anforderung durch FIFO-Füllstand mehr als 50 %
1	1 = Interrupt-Anforderung durch FIFO-Füllstand mehr als 75 %
2	1 = Interrupt- Anforderung durch Überlauf Timer 1
3	1 = Interrupt- Anforderung durch Überlauf Timer 2
4	1 = Interrupt- Anforderung durch Überlauf Timer 3
5	1 = Interrupt- Anforderung durch Aktivierung des externen Eingang IN 1
6	1 = Interrupt- Anforderung durch Aktivierung des externen Eingang IN 2
7	1 = Interrupt- Anforderung durch Aktivierung des externen Eingang IN 3
8	1 = Interrupt- Anforderung durch Aktivierung des externen Eingang IN 4
9 - 15	reserviert

Tabelle 17: Interrut-Status-Register

Ermöglicht das Abfragen, welcher Interrupt aufgetreten ist. Die MEDAQLib nutzt Bit 1. Das Register ist von außen nicht zugänglich, es wird lediglich durch den Treiber verwendet.

### Hinweis!

Das Interrupt-Status-Register gibt Auskunft, durch welche Quelle(n) die Interrupt-Anforderung erfolgte. Eine Interrupt-Anforderung kann auch gleichzeitig durch mehrere Quellen erfolgen. Ist kein Status-Bit gesetzt, so wurde die Interrupt-Anforderung nicht durch die IF2008A generiert, sondern durch eine andere Hardware.

## 5.8 Sensor Baud-Rate

Basisadr.	Sensor-Kanal	Value	Zugriff
+ 08h	1	1 bis 65.535	nur Schreibzugriff
+ 0Ah	2	1 bis 65.535	nur Schreibzugriff
+ 0Ch	3	1 bis 65.535	nur Schreibzugriff
+ 0Eh	4	1 bis 65.535	nur Schreibzugriff
+ 10h	5	1 bis 65.535	nur Schreibzugriff
+ 12h	6	1 bis 65.535	nur Schreibzugriff

Tabelle 18: Basis-Adressen für Sensor Baud-Raten

Register wird automatisch gesetzt, abhängig von der Sensorauswahl und ist dem Benutzer nicht direkt zugänglich. Bei Sensoren mit variabler Baudrate kann beim Öffnen des Sensors die Baudrate gesetzt werden. Zusätzlich setzt die MEDAQLib in der IF-Karte die richtige Baudrate.

$$\text{Value} = (40 \text{ MHz} / \text{Baud-Rate}) - 1$$

Beispiel:

gewünschte Baud-Rate = 691,2 kBaud

Value =  $(40 \text{ MHz} / 691.200) - 1 = 56,87$

Der Eingabewert muss ein ganzzahliger Wert sein d.h. das Ergebnis muss noch gerundet werden:

→ Value = 57

## 5.9 Zähler-Kontrollregister

Basisadr.	Zähler-Kanal	Bit	Zugriff
+ 14h	1	0 bis 15	Schreib- und Lesezugriff
+ 16h	2	0 bis 15	Schreib- und Lesezugriff

Tabelle 19: Basis-Adressen für Zähler-Kontrollregister

Das Zähler-Kontrollregister gibt die Arbeitsweise eines Encoders vor.

Die nachfolgenden Tabellen sind identisch für beide Zählerkanäle!

### Funktionsübersicht

Bit	Funktion
0 bis 3	Interpolation (siehe Tabelle 21: Encoder Interpolation) //Set_EncoderInterpolation bzw. Get_EncoderInterpolation
4	Zählrichtung (siehe Tabelle 22: Encoder Zählrichtung) //Set_EncoderDirection bzw. Get_EncoderDirection
5 bis 7	Zähler-Mode (siehe Tabelle 23: Zähler-Mode) //Set_EncoderMode bzw. Get_EncoderMode
8 bis 11	Latch-Source (siehe Tabelle 24: Zähler Latch-Source) //Set_EncoderLatchSource bzw. Get_EncoderLatchSource
12 bis 15	reserviert

Tabelle 20: Funktionsübersicht für Zähler-Kontrollregister



**Interpolation**

Bit 3	Bit 2	Bit 1	Bit 0	Interpolation
0	0	0	0	1
0	0	0	1	2
0	0	1	0	3
0	0	1	1	4
0	1	0	0	5
0	1	0	1	6
0	1	1	0	8
0	1	1	1	10
1	0	0	0	12
1	0	0	1	16
1	0	1	0	20
1	0	1	1	24
1	1	0	0	32
1	1	0	1	40
1	1	1	0	48
1	1	1	1	64

Tabelle 21: Encoder Interpolation

**Hinweis!**

- Für Encoder mit 1  $V_{ss}$  – Signalen eignen sich alle Interpolationen
- Für Encoder mit TTL – Signalen eignet sich nur die 1-, 2- oder 4-fach Interpolation. Z. B. 4-fach Interpolation: Zählt bei steigender und fallender Flanke und bei Spur A und Spur B; damit das 4-fache an Zählpulsen.

**Zählrichtung**

Bit 4	Zählrichtung
0	normal
1	invers

Tabelle 22: Encoder Zählrichtung

**Zähler-Mode:**

Der Zähler-Mode setzt das Verhalten des Encoders bei Erreichen einer Referenzmarke.

Bit 7	Bit 6	Bit 5	Zähler-Mode						
0	0	0	keine Zählerlade- / LösCHFunktion durch Encoder-Referenzmarken						
0	0	1	Zähler wird mit der nächsten Encoder-Referenzmarke geladen sofern Statusbit 0 bzw. Statusbit 2 „0“ ist.						
0	1	0	Zähler wird mit allen Encoder-Referenzmarken mit dem Inhalt der Laderegister geladen. Statusbit 0 bis 3 haben keine Auswirkung						
0	1	1	Zähler wird mit allen Encoder-Referenzmarken gelöscht und zusätzlich mit dem Inhalt des Laderegisters geladen, wenn der Zählerstand von -1 erreicht wurde. Diese Funktion dient zur Zählerbegrenzung, wobei das Zähler-Laderegister mit der Anzahl der zu begrenzenden Inkremente -1 vorbelegt werden muss.						
1	0	0	Zähler ohne Phasendiskriminator (Ereigniszähler) <table border="1" data-bbox="628 1794 1430 1989"> <thead> <tr> <th>Bit 4</th> <th>Funktion</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Spur A = Zählrichtungssignal Spur B = Zählertaktsignal</td> </tr> <tr> <td>1</td> <td>Spur A = Zählertaktsignal Spur B = Zählrichtungssignal</td> </tr> </tbody> </table>	Bit 4	Funktion	0	Spur A = Zählrichtungssignal Spur B = Zählertaktsignal	1	Spur A = Zählertaktsignal Spur B = Zählrichtungssignal
Bit 4	Funktion								
0	Spur A = Zählrichtungssignal Spur B = Zählertaktsignal								
1	Spur A = Zählertaktsignal Spur B = Zählrichtungssignal								
1	0	1	reserviert						
1	1	0	reserviert						
1	1	1	reserviert						

Tabelle 23: Zähler-Mode

**Latch-Source:**

Ermöglicht das synchronisierte Aufnehmen von Sensor- und Encoderwerten. Der Befehl in der MEDAQLib lautet Set\_EncoderLatchSource.

Bit 11	Bit 10	Bit 9	Bit 8	Latch-Source
0	0	0	0	Hardware-Latch gesperrt
0	0	0	1	Timer 1
0	0	1	0	Timer 2
0	0	1	1	Timer 3
0	1	0	0	Sensor-Kanal 1
0	1	0	1	Sensor-Kanal 2
0	1	1	0	Sensor-Kanal 3
0	1	1	1	Sensor-Kanal 4
1	0	0	0	Sensor-Kanal 5
1	0	0	1	Sensor-Kanal 6
1	0	1	0	IN 1 (nur mit IF2008E)
1	0	1	1	IN 2 (nur mit IF2008E)
1	1	0	0	IN 3 (nur mit IF2008E)
1	1	0	1	IN 4 (nur mit IF2008E)
1	1	1	0	2. Referenzmarke
1	1	1	1	alle Referenzmarken

Tabelle 24: Zähler Latch-Source

**5.10 Zähler Preload**

Vorbelegungsregister für die Encoderstartwerte, 32 Bit breit. Der Befehl in der MEDAQLib lautet Set\_EncoderPreload.

Basisadr.	Zähler-Kanal	Value	Zugriff
+ 18h	1 LSW	0 bis 65.535	nur Schreibzugriff
+ 1Ah	1 MSW	0 bis 65.535	nur Schreibzugriff
+ 1Ch	2 LSW	0 bis 65.535	nur Schreibzugriff
+ 1Eh	2 MSW	0 bis 65.535	nur Schreibzugriff

Tabelle 25: Basis-Adressen für Zähler Preload

**5.11 Zählerwert**

Befehl zum Auslesen. Der Befehl in der MEDAQLib lautet Get\_EncoderValue.

Basisadr.	Zähler-Kanal	Value	Zugriff
+ 18h	1 LSW	0 bis 65.535	nur Lesezugriff
+ 1Ah	1 MSW	0 bis 65.535	nur Lesezugriff
+ 1Ch	2 LSW	0 bis 65.535	nur Lesezugriff
+ 1Eh	2 MSW	0 bis 65.535	nur Lesezugriff

Tabelle 26: Basis-Adressen für Zählerwert

LSW = Least Significant Word

MSW = Most Significant Word

## 5.12 Timer

Befehl in der MEDAQLib: Set\_TimerFrequency.

Anwendung:

- Timer auf Digitalausgang schalten
- Datenerfassung synchronisieren
- Triggersignal erzeugen

Beispiel: Sensor zeitbasiert synchronisieren mit dem Kommando Set\_TriggerSource.

Basisadr.	Timer	Value	Zugriff
+ 20h	1 Frequenz	0 bis 65.535	nur Schreibzugriff
+ 22h	1 Pulsweite	0 bis 65.535	nur Schreibzugriff
+ 24h	2 Frequenz	0 bis 65.535	nur Schreibzugriff
+ 26h	2 Pulsweite	0 bis 65.535	nur Schreibzugriff
+ 28h	3 Frequenz	0 bis 65.535	nur Schreibzugriff
+ 2Ah	3 Pulsweite	0 bis 65.535	nur Schreibzugriff
+ 2Ch	Clock-Teiler		Schreib- und Lesezugriff

Tabelle 27: Basis-Adressen für Timer

$$\text{Value}(F) = (F_{\text{Clock}} / F_{\text{OUT}}) - 1$$

$$\text{Value}(PW) = (PW_{\text{OUT}} / T_{\text{Clock}})$$

Beispiel:

gewünschte Frequenz  $F_{\text{OUT}} = 10 \text{ kHz}$

gewünschte Pulsweite  $PW_{\text{OUT}} = 25 \mu\text{s}$

Clockteiler = 0  $\rightarrow F_{\text{Clock}} = 20 \text{ MHz}$ ,  $T_{\text{Clock}} = 50\text{ns}$  (Clockteiler siehe nachfolgende Tabelle)

$$\text{Value}(F) = (20 \text{ MHz} / 10 \text{ kHz}) - 1 = 1999$$

$$\text{Value}(PW) = (25 \mu\text{s} / 50 \text{ ns}) = 500$$

Die Eingabewerte müssen ganzzahlig sein!

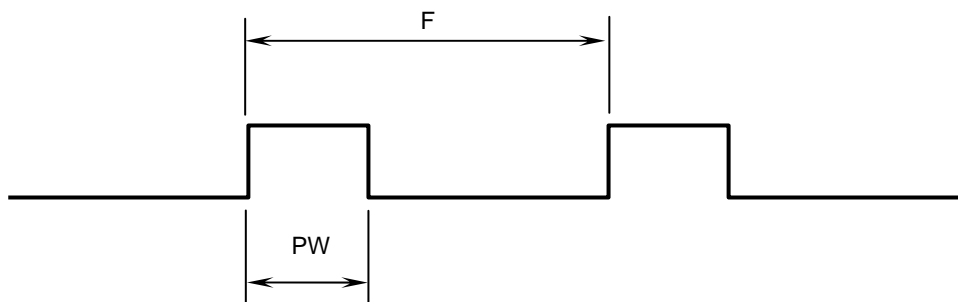


Bild 5: Timer-Frequenz und Pulsweite

### Hinweis!

Die Pulsweite hat nur Einfluss auf die Ausgänge „Sensor-Trigger“ und „Optokoppler“ und nicht auf die internen Synchronisations-Signale. Hierfür wird der Timer-Nulldurchgang verwendet.

Zum Ausschalten des Timers muss die Frequenz mit „0“ programmiert werden. Ist bei ausgeschaltetem Timer die Pulsweite  $> 0$  programmiert, so ist der Ausgang ständig auf High gesetzt. Ist die Pulsweite dagegen auch mit „0“ programmiert, so ist der Ausgang ständig auf Low gesetzt.

**Clock-Teiler:**

Bit 3	Bit 2	Bit 1	Bit 0	Clockfrequenz Timer 1
Bit 7	Bit 6	Bit 5	Bit 4	Clockfrequenz Timer 2
Bit 11	Bit 10	Bit 9	Bit 8	Clockfrequenz Timer 3
0	0	0	0	20 MHz
0	0	0	1	20 MHz / 2
0	0	1	0	20 MHz / 4
0	0	1	1	20 MHz / 8
0	1	0	0	20 MHz / 16
0	1	0	1	20 MHz / 32
0	1	1	0	20 MHz / 64
0	1	1	1	20 MHz / 128
1	0	0	0	20 MHz / 256
1	0	0	1	20 MHz / 512
1	0	1	0	20 MHz / 1024
1	0	1	1	20 MHz / 2048
1	1	0	0	20 MHz / 4096
1	1	0	1	20 MHz / 8192
1	1	1	0	20 MHz / 16384
1	1	1	1	20 MHz / 32768

Tabelle 28: Timer Clock-Teiler

**Hinweis!**

Bit 12 bis Bit 15 sind reserviert.

**5.13 ADC**

Zwei Register für die aktuell konvertierten Werte des A/D-Wandlers. MEDAQLib liest die Werte mit dem Befehl Get\_ADCValue oder durch Schreiben der Werte in den FIFO.

Basisadr.	ADC-Kanal	Value	Zugriff
+ 20h	1	0 bis 65.535	nur Lesezugriff
+ 22h	2	0 bis 65.535	nur Lesezugriff

Tabelle 29: Basis-Adressen für ADC

## 5.14 Status

Basisadr. + 24h (nur Lesezugriff)

Bit	Funktion
0	1 = Encoder 1: 1. Referenzmarke überfahren //Get_EncoderReference
1	1 = Encoder 1: 2. Referenzmarke überfahren
2	1 = Encoder 2: 1. Referenzmarke überfahren
3	1 = Encoder 2: 2. Referenzmarke überfahren
4	0 = Transmitter bereit zur Übertragung neuer Daten 1 = Transmitter ist beschäftigt
5	0 = kein Erweiterungsmodul mit Sensor 5 / 6 vorhanden 1 = Erweiterungsmodul mit Sensor 5 / 6 vorhanden
6	0 = kein Erweiterungsmodul für externe I/O vorhanden 1 = Erweiterungsmodul für externe I/O vorhanden
7	0 = kein Erweiterungsmodul mit ADC vorhanden 1 = Erweiterungsmodul mit ADC vorhanden
8 – 15	FPGA-Version

Tabelle 30: Status

Hinweis zu Bit 4: Interne Verwendung. Die IF-Karte braucht für das Senden von Kommandos an Sensoren länger als die MEDAQLib an die IF-Karte. Mit diesem Bit prüft die MEDAQLib, ob die IF-Karte bereit ist für weitere Aktionen.

Hinweis zu Bit 5, 6, 7: Interne Verwendung. Zeigt an, ob die IF2008E angesteckt ist. Kommandos für das Abfragen der Bits: Is\_Channel56Available, Is\_ADCAvailable oder IS\_DigitalIOAvailable.

## 5.15 Input

Basisadr. + 26h (nur Lesezugriff)

Bit	Funktion
0	1 = ext. Eingang IN 1 aktiv
1	1 = ext. Eingang IN 2 aktiv
2	1 = ext. Eingang IN 3 aktiv
3	1 = ext. Eingang IN 4 aktiv
4	1 = RxD Eingang am Sensoreingang 1 aktiv
5	1 = RxD Eingang am Sensoreingang 2 aktiv
6	1 = RxD Eingang am Sensoreingang 3 aktiv
7	1 = RxD Eingang am Sensoreingang 4 aktiv
8	1 = RxD Eingang am Sensoreingang 5 aktiv
9	1 = RxD Eingang am Sensoreingang 6 aktiv
8 – 15	reserviert

Tabelle 31: Input

Hinweis zu Bit 0 bis 3: Interne Verwendung. Zeigt an, ob die Digitaleingänge High oder Low sind. Kommando: Get\_DigitalInValue.

Hinweis zu Bit 4 bis 9: Interne Verwendung. Zeigt an, ob die RxD-Eingänge High oder Low sind. Kommando: Get\_RxDValue.

## 5.16 Output-Register

Basisadr. + 2Eh (Schreib- und Lesezugriff)

Bit	Funktion	Ausgangssignal
0	0 = OUT 1 OFF    Optokoppler gesperrt 1 = OUT 1 ON     Optokoppler leitend	Output 1 = High Output 1 = Low  //Set_DigitalOutValue bzw. Get_DigitalOutValue
1	0 = OUT 2 OFF    Optokoppler gesperrt 1 = OUT 2 ON     Optokoppler leitend	Output 2 = High Output 2 = Low
2	0 = OUT 3 OFF    Optokoppler gesperrt 1 = OUT 3 ON     Optokoppler leitend	Output 3 = High Output 3 = Low
3	0 = OUT 4 OFF    Optokoppler gesperrt 1 = OUT 4 ON     Optokoppler leitend	Output 4 = High Output 4 = Low
4	0 = TxD 1         inaktiv 1 = TxD 1         aktiv	TxD 1+ = High    TxD 1- = Low TxD 1+ = Low     TxD 1- = High  //Set_TxDValue bzw. Get_TxDValue
5	0 = TxD 2         inaktiv 1 = TxD 2         aktiv	TxD 2+ = High    TxD 2- = Low TxD 2+ = Low     TxD 2- = High
6	0 = TxD 3         inaktiv 1 = TxD 3         aktiv	TxD 3+ = High    TxD 3- = Low TxD 3+ = Low     TxD 3- = High
7	0 = TxD 4         inaktiv 1 = TxD 4         aktiv	TxD 4+ = High    TxD 4- = Low TxD 4+ = Low     TxD 4- = High
8	0 = TxD 5         inaktiv 1 = TxD 5         aktiv	TxD 5+ = High    TxD 5- = Low TxD 5+ = Low     TxD 5- = High
9	0 = TxD 6         inaktiv 1 = TxD 6         aktiv	TxD 6+ = High    TxD 6- = Low TxD 6+ = Low     TxD 6- = High
10	0 = TRG 1         inaktiv 1 = TRG 1         aktiv	TRG 1+ = Low     TRG 1- = High TRG 1+ = High    TRG 1- = Low  //Set_TrgValue bzw. Get_TrgValue
11	0 = TRG 2         inaktiv 1 = TRG 2         aktiv	TRG 2+ = Low     TRG 2- = High TRG 2+ = High    TRG 2- = Low
12	0 = TRG 3         inaktiv 1 = TRG 3         aktiv	TRG 3+ = Low     TRG 3- = High TRG 3+ = High    TRG 3- = Low
13	0 = TRG 4         inaktiv 1 = TRG 4         aktiv	TRG 4+ = Low     TRG 4- = High TRG 4+ = High    TRG 4- = Low
14	0 = TRG 5         inaktiv 1 = TRG 5         aktiv	TRG 5+ = Low     TRG 5- = High TRG 5+ = High    TRG 5- = Low
15	0 = TRG 6         inaktiv 1 = TRG 6         aktiv	TRG 6+ = Low     TRG 6- = High TRG 6+ = High    TRG 6- = Low

Tabelle 32: Output-Register

### Hinweise!

Für alle Ausgänge stehen mehrere Signalquellen zur Verfügung. Oben aufgeführte Bits werden nur dann durchgeschaltet, wenn der entsprechende Mode eingestellt ist (siehe Tabelle 33: Mode Opto- und TxD-Ausgänge auf Seite 23).

<sup>1)</sup> Nur mit Option Board möglich

## 5.17 Mode Opto- und TxD-Ausgänge

Basisadr. + 30h (Schreib- und Lesezugriff)

Bit	Funktion		
0 und 1	<b>Bit 1</b>	<b>Bit 0</b>	<b>Funktion</b>
	0	0	Output 1 schaltet mit Adr. 2Eh Bit 0 //Set_DigitalOutSource bzw. Get_DigitalOutSource
	0	1	Output 1 schaltet mit Timer 1 Pulsweite
	1	0	Output 1 schaltet mit Timer 2 Pulsweite
	1	1	Output 1 schaltet mit Timer 3 Pulsweite
2 und 3	<b>Bit 3</b>	<b>Bit 2</b>	<b>Funktion</b>
	0	0	Output 2 schaltet mit Adr. 2Eh Bit 1
	0	1	Output 2 schaltet mit Timer 1 Pulsweite
	1	0	Output 2 schaltet mit Timer 2 Pulsweite
	1	1	Output 2 schaltet mit Timer 3 Pulsweite
4 und 5	<b>Bit 5</b>	<b>Bit 4</b>	<b>Funktion</b>
	0	0	Output 3 schaltet mit Adr. 2Eh Bit 2
	0	1	Output 3 schaltet mit Timer 1 Pulsweite
	1	0	Output 3 schaltet mit Timer 2 Pulsweite
	1	1	Output 3 schaltet mit Timer 3 Pulsweite
6 und 7	<b>Bit 7</b>	<b>Bit 6</b>	<b>Funktion</b>
	0	0	Output 4 schaltet mit Adr. 2Eh Bit 3
	0	1	Output 4 schaltet mit Timer 1 Pulsweite
	1	0	Output 4 schaltet mit Timer 2 Pulsweite
	1	1	Output 4 schaltet mit Timer 3 Pulsweite
8	0 = TxD 1 schaltet mit Transmitter 1 = TxD 1 schaltet mit Adr. 2Eh Bit 4 //Set_TxDSource bzw. Get_TxDSource		
9	0 = TxD 2 schaltet mit Transmitter 1 = TxD 2 schaltet mit Adr. 2Eh Bit 5		
10	0 = TxD 3 schaltet mit Transmitter 1 = TxD 3 schaltet mit Adr. 2Eh Bit 6		
11	0 = TxD 4 schaltet mit Transmitter 1 = TxD 4 schaltet mit Adr. 2Eh Bit 7		
12	0 = TxD 5 schaltet mit Transmitter 1 = TxD 5 schaltet mit Adr. 2Eh Bit 8		
13	0 = TxD 6 schaltet mit Transmitter 1 = TxD 6 schaltet mit Adr. 2Eh Bit 9		
14 - 15	reserviert		

Tabelle 33: Mode Opto- und TxD-Ausgänge

### Hinweis!

Output 1 bis 4 stehen nur bei der IF2008E zur Verfügung.

## 5.18 Mode Trigger-Ausgänge

Konfiguriert die sechs Triggerausgänge; Befehl in der MEDAQLib lautet Set\_TrgSource.  
Basisadr. + 32h (Schreib- und Lesezugriff)

Bit	Funktion		
0 und 1	<b>Bit 1</b>	<b>Bit 0</b>	<b>Funktion</b>
	0	0	Trigger 1 schaltet mit Adr. 2Eh Bit 10
	0	1	Trigger 1 schaltet mit Timer 1 Pulsweite
	1	0	Trigger 1 schaltet mit Timer 2 Pulsweite
	1	1	Trigger 1 schaltet mit Timer 3 Pulsweite
2 und 3	<b>Bit 3</b>	<b>Bit 2</b>	<b>Funktion</b>
	0	0	Trigger 2 schaltet mit Adr. 2Eh Bit 11
	0	1	Trigger 2 schaltet mit Timer 1 Pulsweite
	1	0	Trigger 2 schaltet mit Timer 2 Pulsweite
	1	1	Trigger 2 schaltet mit Timer 3 Pulsweite
4 und 5	<b>Bit 5</b>	<b>Bit 4</b>	<b>Funktion</b>
	0	0	Trigger 3 schaltet mit Adr. 2Eh Bit 12
	0	1	Trigger 3 schaltet mit Timer 1 Pulsweite
	1	0	Trigger 3 schaltet mit Timer 2 Pulsweite
	1	1	Trigger 3 schaltet mit Timer 3 Pulsweite
6 und 7	<b>Bit 7</b>	<b>Bit 6</b>	<b>Funktion</b>
	0	0	Trigger 4 schaltet mit Adr. 2Eh Bit 13
	0	1	Trigger 4 schaltet mit Timer 1 Pulsweite
	1	0	Trigger 4 schaltet mit Timer 2 Pulsweite
	1	1	Trigger 4 schaltet mit Timer 3 Pulsweite
8 und 9	<b>Bit 9</b>	<b>Bit 8</b>	<b>Funktion</b>
	0	0	Trigger 5 schaltet mit Adr. 2Eh Bit 14
	0	1	Trigger 5 schaltet mit Timer 1 Pulsweite
	1	0	Trigger 5 schaltet mit Timer 2 Pulsweite
	1	1	Trigger 5 schaltet mit Timer 3 Pulsweite
10 und 11	<b>Bit 11</b>	<b>Bit 10</b>	<b>Funktion</b>
	0	0	Trigger 6 schaltet mit Adr. 2Eh Bit 15
	0	1	Trigger 6 schaltet mit Timer 1 Pulsweite
	1	0	Trigger 6 schaltet mit Timer 2 Pulsweite
	1	1	Trigger 6 schaltet mit Timer 3 Pulsweite



Bit	Funktion			
12 – 14	Bit 14	Bit 13	Bit 12	Latch-Source
	0	0	0	Hardware-Latch gesperrt
	0	0	1	Timer 1
	0	1	0	Timer 2
	0	1	1	Timer 3
	1	0	0	Sensor-Kanal 1
	1	0	1	Sensor-Kanal 2
	1	1	0	Sensor-Kanal 3
1	1	1	Sensor-Kanal 4	
15	reserviert			

Tabelle 34: Mode Trigger-Ausgänge

**Hinweis!**

Mit den Bits 12-14 kann ein Latch-Source selektiert werden, mit dessen Trigger-Ereignis die externen Eingänge (IN1-4) sowie die RxD-Eingänge (Sensor 1-6) synchron in den FIFO geschrieben werden.

Befehl in der MEDAQLib lautet Set\_DigitalInLatchSource.

## 5.19 ADC-Kontrollregister

Bit 0 bis 7 definieren, wann ein AD-Wert aufgenommen und in den FIFO geschrieben wird. Die Synchronisierung kann mit einem Timer, einem Sensorkanal oder durch einen Puls an den digitalen Eingängen IN1 ... IN4 erfolgen. Mit den digitalen Eingängen ist damit eine externe Triggerung möglich.

Basisadr. + 34h (Schreib- und Lesezugriff)

Bit 3	Bit 2	Bit 1	Bit 0	Konvertierungs-Source ADC1
Bit 7	Bit 6	Bit 5	Bit 4	Konvertierungs-Source ADC2
0	0	0	0	Hardware-Konvertierung gesperrt
0	0	0	1	Timer 1
0	0	1	0	Timer 2
0	0	1	1	Timer 3
0	1	0	0	Sensor-Kanal 1
0	1	0	1	Sensor-Kanal 2
0	1	1	0	Sensor-Kanal 3
0	1	1	1	Sensor-Kanal 4
1	0	0	0	Sensor-Kanal 5
1	0	0	1	Sensor-Kanal 6
1	0	1	0	IN 1 (nur mit IF2008E/ I/O)
1	0	1	1	IN 2 (nur mit IF2008E/ I/O)
1	1	0	0	IN 3 (nur mit IF2008E/ I/O)
1	1	0	1	IN 4 (nur mit IF2008E/ I/O)
1	1	1	0	reserviert
1	1	1	1	reserviert

Tabelle 35: ADC-Kontrollregister Bit 0-7

Bit	Funktion
8	0 = ADC1 Datenausgabe binär 2er-Komplement 1 = ADC1 Datenausgabe binär unkonvertiert
9	0 = ADC2 Datenausgabe binär 2er-Komplement 1 = ADC2 Datenausgabe binär unkonvertiert
10 – 15	reserviert

Tabelle 36: ADC-Kontrollregister Bit 8-15

Hinweis zu Bit 8 und 9: Interne Verwendung. Funktion wird in der MEDAQLib nicht genutzt. Die MEDAQLib kann die Schalterstellungen für die Analogbereiche nicht auslesen.

Analog Input				Digital Output	
0 – 5 V	0 – 10 V	+/-5 V	+/-10 V	Binär 2er-Kompliment	Binär unkonvertiert
+4,99 V	+9,99 V	+4,99 V	+9,99 V	7FFF	FFFF
2,5 V	5 V	0 V	0 V	0000	8000
+2,499 V	+4,999 V	-153 $\mu$ V	-305 $\mu$ V	FFFF	7FFF
0 V	0 V	-5 V	-10 V	8000	0000

Tabelle 37: ADC-Konvertierungsergebnis

## 5.20 Parity-Enable-Register

Basisadr. + 36h (Schreibzugriff)

Bit	Funktion
0	0 = Parity-Bit für Sensor-Kanal 1 gesperrt 1 = Parity-Bit für Sensor-Kanal 1 freigegeben (nur Even-Parity)
1	0 = Parity-Bit für Sensor-Kanal 2 gesperrt 1 = Parity-Bit für Sensor-Kanal 2 freigegeben (nur Even-Parity)
2	0 = Parity-Bit für Sensor-Kanal 3 gesperrt 1 = Parity-Bit für Sensor-Kanal 3 freigegeben (nur Even-Parity)
3	0 = Parity-Bit für Sensor-Kanal 4 gesperrt 1 = Parity-Bit für Sensor-Kanal 4 freigegeben (nur Even-Parity)
4	0 = Parity-Bit für Sensor-Kanal 5 gesperrt 1 = Parity-Bit für Sensor-Kanal 5 freigegeben (nur Even-Parity)
5	0 = Parity-Bit für Sensor-Kanal 6 gesperrt 1 = Parity-Bit für Sensor-Kanal 6 freigegeben (nur Even-Parity)
6-15	reserviert

Tabelle 38: Parity-Enable-Register

Wird ein Sensor, z. B. Laufzeitsensor, ausgewählt dessen Datenprotokoll das Paritybit verwendet, so aktiviert die MEDAQLib automatisch das zugehörige Parityregister.

## 5.21 Parity-Error-Register

Basisadr. + 36h (Lesezugriff)

Bit	Funktion
0	1 = Parity-Error Sensorkanal 1
1	1 = Parity-Error Sensorkanal 2
2	1 = Parity-Error Sensorkanal 3
3	1 = Parity-Error Sensorkanal 4
4	1 = Parity-Error Sensorkanal 5
5	1 = Parity-Error Sensorkanal 6
6 – 15	reserviert

Tabelle 39: Parity-Error-Register

## 6 Verdrahtungsempfehlung

### 6.1 Sensor ILD1302 und ILD1402

Pin X1/X2 IF2008A Pin X1 IF2008E	Signal	ILD 1302 und ILD1402		Signal Sensor
		Pin Sensor 1	Pin Sensor 2	
1	Sensor 1 TxD-	4		RxD-
2	Sensor 1 TxD+	3		RxD+
3	Sensor 1 RxD-	6		TxD-
4	Sensor 1 RxD+	5		TxD+
5	Spannungsversorgung 0 V	12	12	GND
6	Sensor 1 TRG+	9		TeachIn
7	Sensor 1 TRG-	NC	NC	
8	Sensor 2 TRG+		9	TeachIn
9	Sensor 2 TRG-	NC	NC	
10	Spannungsversorgung +24 V	7	7	+UB
11	Sensor 2 TxD-		4	RxD-
12	Sensor 2 TxD+		3	RxD+
13	Sensor 2 RxD-		6	TxD-
14	Sensor 2 RxD+		5	TxD+
15	GND (galvan. getrennt zu PC-GND)	12	12	GND

Tabelle 40: Sensorverdrahtung ILD1302 und ILD1402

### 6.2 Sensor ILD1700

Pin X1/X2 IF2008A Pin X1 IF2008E	Signal	ILD1700		Signal Sensor
		Pin Sensor 1	Pin Sensor 2	
1	Sensor 1 TxD-	11		RxD-
2	Sensor 1 TxD+	12		RxD+
3	Sensor 1 RxD-	2		TxD-
4	Sensor 1 RxD+	1		TxD+
5	Spannungsversorgung 0 V	6	6	GND
6	Sensor 1 TRG+	3		TRG+
7	Sensor 1 TRG-	4		TRG-
8	Sensor 2 TRG+		3	TRG+
9	Sensor 2 TRG-		4	TRG-
10	Spannungsversorgung +24 V	5	5	+UB
11	Sensor 2 TxD-		11	RxD-
12	Sensor 2 TxD+		12	RxD+
13	Sensor 2 RxD-		2	TxD-
14	Sensor 2 RxD+		1	TxD+
15	GND (galvan. getrennt zu PC-GND)	6	6	GND

Tabelle 41: Sensorverdrahtung ILD1700

### 6.3 Sensor ILD2200

Pin X1/X2 IF2008A Pin X1 IF2008E	Signal	ILD2200		Signal Sensor
		Pin Sensor 1	Pin Sensor 2	
1	Sensor 1 TxD-	24		RxD-
2	Sensor 1 TxD+	11		RxD+
3	Sensor 1 RxD-	10		TxD-
4	Sensor 1 RxD+	23		TxD+
5	Spannungsversorgung 0 V	14	14	Versorgung Masse
6	Sensor 1 TRG+	20		SyncIn+
7	Sensor 1 TRG-	NC		
8	Sensor 2 TRG+		20	SyncIn+
9	Sensor 2 TRG-		NC	
10	Spannungsversorgung +24 V	1	1	+UB
11	Sensor 2 TxD-		24	RxD-
12	Sensor 2 TxD+		11	RxD+
13	Sensor 2 RxD-		10	TxD-
14	Sensor 2 RxD+		23	TxD+
15	GND (galvan. getrennt zu PC-GND)	7	7	SyncIn-

Tabelle 42: Sensorverdrahtung ILD2200

### 6.4 Sensor ILD2300

Pin X1/X2 IF2008A Pin X1 IF2008E	Signal	ILD2300		Signal Sensor
		Pin Sensor 1	Pin Sensor 2	
1	Sensor 1 TxD-	8		RxD-
2	Sensor 1 TxD+	7		RxD+
3	Sensor 1 RxD-	10		TxD-
4	Sensor 1 RxD+	9		TxD+
5	Spannungsversorgung 0 V	2	2	Versorgung Masse
6	Sensor 1 TRG+	5		SyncIn+
7	Sensor 1 TRG-	6		SyncIn-
8	Sensor 2 TRG+		5	SyncIn+
9	Sensor 2 TRG-		6	SyncIn-
10	Spannungsversorgung +24 V	1	1	+UB
11	Sensor 2 TxD-		8	RxD-
12	Sensor 2 TxD+		7	RxD+
13	Sensor 2 RxD-		10	TxD-
14	Sensor 2 RxD+		9	TxD+
15	GND (galvanisch getrennt zu PC-GND)	2	2	

Tabelle 43: Sensorverdrahtung ILD2300

## 6.5 Encoder-Interface

Pin X3 IF2008A	Signal	1 V <sub>ss</sub> oder RS422		TTL (single-ended)	
		Signal Encoder 1	Signal Encoder 2	Signal Encoder 1	Signal Encoder 2
1	Encoder 1 Spur A+	A+		A	
2	Encoder 1 Spur A-	A-		open	
3	Encoder 2 Spur A+		A+		A
4	Encoder 2 Spur A-		A-		open
5	VCC (+5 V)	+UB	+UB	+UB	+UB
6	Encoder 1 Spur B+	B+		B	
7	Encoder 1 Spur B-	B-		open	
8	Encoder 2 Spur B+		B+		B
9	Encoder 2 Spur B-		B-		open
10	GND	GND	GND	GND	GND
11	Encoder 1 Spur R+	R+		R	
12	Encoder 1 Spur R-	R-		open	
13	Encoder 2 Spur R+		R+		R
14	Encoder 2 Spur R-		R-		open
15	GND	GND	GND	GND	GND

Tabelle 44: Encoder-Interface

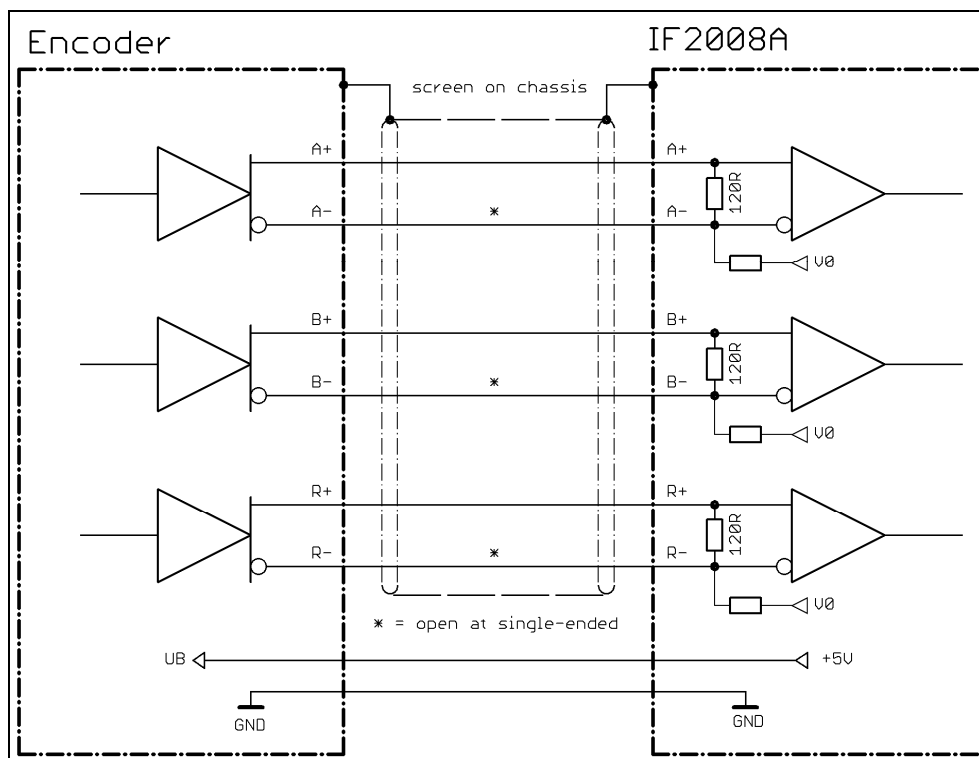


Bild 6: Blockschaltbild Encoder-Interface

### Hinweis!

Plus-Eingänge (A+, B+, R+) dürfen nicht offen bleiben. Wird zum Beispiel bei einem Ereigniszähler nur das Taktsignal verwendet, so müssen die nicht belegten Plus-Eingänge auf GND oder VCC gelegt werden.

Nicht belegte Minus-Eingänge (A-, B-, R-) dürfen nicht mit GND verbunden werden.

### 6.6 Optokoppler I/O

Pin X2 IF2008E	Signal
1	OUT 1
2	OUT 2
3	OUT 3
4	OUT 4
5	GND (galvanisch getrennt zu PC-GND)
6	IN 1
7	IN 2
8	IN 3
9	IN 4

Tabelle 45: Optokoppler I/O

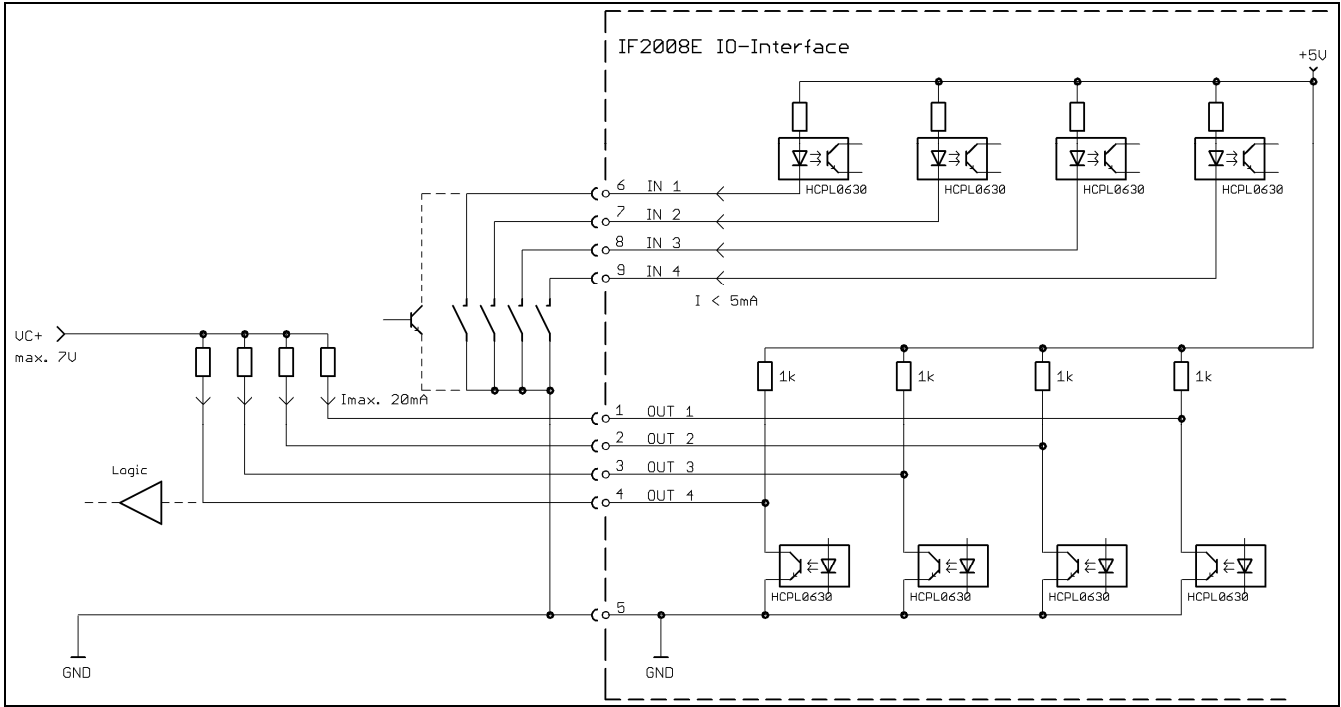


Bild 7: Blockschaltbild Optokoppler I/O

## 7 Beispiele für Synchronisieren, Triggerung, Torung

Synchronisieren = schnelle Anwendung (passiv, nur „horchen“)

Triggerung = langsame Anwendung (aktiv, Befehl und Antwort)

Die nachfolgenden Beispiele zeigen, exemplarisch am Sensor ILD1700, die Möglichkeiten zur Datenaufnahme.

### 7.1 Einen Messwert vom Sensor holen, Software-Trigger

- SET\_ERROROUTPUT
- GET\_MEASVALUE: Ein Messwert oder mehrere.
- DATAAVAIL: Ist ein Messwert oder mehrere verfügbar.
- TRANSFERDATA: Liest Messwert aus Ringpuffer aus.

### 7.2 Hardware-Triggerung

- SET\_ERROROUTPUT: Sensor in Trigger-Mode setzen, X = "2" oder "3".
- SET\_SYNCMODE/TRIGGERMODE: Reaktion des Sensoreingangs auf Flanke oder Pegel einstellen.
- Rechteckimpuls am Sensoreingang, ausgelöst durch z. B. SPS oder Triggerausgang der IF2008.
- DATAAVAIL: nicht unbedingt erforderlich.
- TRANSFERDATA: Liest Messwert aus Ringpuffer aus, liefert immer den ältesten Wert zuerst. ODER
- POLL: Liefert immer den neuesten Wert, Werte bleiben im Ringpuffer erhalten. Funktion für Steuer- und Regelanwendungen.

### 7.3 Softwaretorung (Gate) am Sensor

- DAT\_OUT\_ON: Schaltet die digitale Datenausgabe der Messwerte ein.
- TRANSFERDATA: Liest Messwert aus Ringpuffer aus, liefert immer den ältesten Wert zuerst.
- DAT\_OUT\_OFF: Schaltet die digitale Ausgabe der Messwerte aus.

### 7.4 Hardwaretorung (Gate) am Sensor

- SET\_SYNCMODE/TRIGGERMODE: Reaktion des Sensoreingangs auf Low- oder Highpegel einstellen.
- Pegel an Sensoreingang anlegen
- TRANSFERDATA: Liest Messwert aus Ringpuffer aus, liefert immer den ältesten Wert zuerst.
- Dieses Beispiel setzt die Funktionalität im Sensor voraus.



## 7.5 Hardwaretorung (Gate) mit IF2008

- Sensor liefert kontinuierlich Daten.
- USE\_GATE: Öffnet oder sperrt den FIFO.
- 5V TTL an IF2008: Daten werden aufgenommen oder gesperrt.
- CLEAR\_BUFFERS: Löscht den Ringpuffer und die Ein-Ausgangspuffer der IF2008. Damit wird vermieden, dass alte Messwerte verwendet werden.
- TRANSFERDATA: Liest Messwert aus Ringpuffer aus, liefert immer den ältesten Wert zuerst.

## 7.6 Synchrones Messwerteinlesen mit Encoder und IF2008

Aufbau: Sensor an Kanal 1, Encoder an Kanal 7 (erster Encoder).

- SensorID = CreateSensorInstance (SENSOR\_ILD1700)
- SetParameterString (SensorID, "IP\_Interface", "IF2008")
- OpenSensor (SensorID)
- SetParameterString (SensorID, "S\_Command", "Get\_Settings")
- SensorCommand (SensorID)
  
- EncoderID = CreateSensorInstance (PCI\_CARD\_IF2008)
- SetParameterString (EncoderID, "IP\_Interface", "IF2008")
- SetParameterInt (EncoderID, "IP\_ChannelNumber", 6)
- OpenSensor (EncoderID)
- SetParameterString (EncoderID, "S\_Command", "Set\_EncoderInterpolation")
- SetParameterInt (EncoderID, "SP\_EncoderInterpolation", 0) // 0 = Einfachauswertung
- SensorCommand (EncoderID)
  
- SetParameterString (EncoderID, "S\_Command", "Set\_EncoderLatchSource")
- SetParameterInt (EncoderID, "SP\_EncoderLatchSource", 4) //4 = Sensor an Kanal 1, synchron mit ILD1700 lesen
- SensorCommand (EncoderID)
  
- SetParameterString (SensorID, "S\_Command", "Clear\_Buffers")
- SetParameterInt (SensorID, "SP\_AllDevices", 1) // 1 = löscht alle angeschlossenen Gerätepuffer
- SensorCommand (SensorID)

```
while (running)
{
    TransferData (SensorID, rawData, scaledData, nbrValues, read)
    TransferData (EncoderID, rawData, scaledData, nbrValues, read)
    ...
}
```

## Abbildungsverzeichnis

Bild 1: Platinenansicht IF2008A .....	6
Bild 2: Platinenansicht IF2008E .....	7
Bild 3: Schalterstellungen Trigger-Level IF2008A (links) und IF2008E (rechts).....	10
Bild 4: Schalterstellungen ADC-Level .....	10
Bild 5: Timer-Frequenz und Pulsweite .....	19
Bild 6: Blockschaltbild Encoder-Interface .....	30
Bild 7: Blockschaltbild Optokoppler I/O .....	31

## Tabellenverzeichnis

Tabelle 1: Steckerbelegung Sensor-Interface .....	8
Tabelle 2: Steckerbelegung Encoder-Interface .....	8
Tabelle 3: Steckerbelegung Sensor-Power .....	9
Tabelle 4: Steckerbelegung I/O-Interface.....	9
Tabelle 5: Steckerbelegung Analog-Interface .....	9
Tabelle 6: Schalterstellungen Trigger-Level.....	10
Tabelle 7: Schalterstellungen ADC-Level.....	10
Tabelle 8: Header-Configuration .....	11
Tabelle 9: Lokale Adress-Belegung .....	11
Tabelle 10: Sende-Register.....	12
Tabelle 11: FIFO-Datenspeicher .....	12
Tabelle 12: FIFO-Datenspeicher – Datenquellen.....	12
Tabelle 13: Set- / Reset- / Latch-Register .....	13
Tabelle 14: FIFO-Volumen .....	13
Tabelle 15: FIFO-Enable-Register .....	14
Tabelle 16: Interrupt-Enable-Register .....	15
Tabelle 17: Interrupt-Status-Register .....	15
Tabelle 18: Basis-Adressen für Sensor Baud-Raten.....	16
Tabelle 19: Basis-Adressen für Zähler-Kontrollregister.....	16
Tabelle 20: Funktionsübersicht für Zähler-Kontrollregister.....	16
Tabelle 21: Encoder Interpolation .....	17
Tabelle 22: Encoder Zählrichtung .....	17
Tabelle 23: Zähler-Mode .....	17
Tabelle 24: Zähler Latch-Source .....	18
Tabelle 25: Basis-Adressen für Zähler Preload.....	18
Tabelle 26: Basis-Adressen für Zählerwert .....	18
Tabelle 27: Basis-Adressen für Timer .....	19
Tabelle 28: Timer Clock-Teiler .....	20
Tabelle 29: Basis-Adressen für ADC.....	20
Tabelle 30: Status .....	21
Tabelle 31: Input.....	21
Tabelle 32: Output-Register .....	22
Tabelle 33: Mode Opto- und TxD-Ausgänge.....	23
Tabelle 34: Mode Trigger-Ausgänge.....	25
Tabelle 35: ADC-Kontrollregister Bit 0-7 .....	26
Tabelle 36: ADC-Kontrollregister Bit 8-15 .....	26
Tabelle 37: ADC-Konvertierungsergebnis.....	26
Tabelle 38: Parity-Enable-Register .....	27
Tabelle 39: Parity-Error-Register.....	27
Tabelle 40: Sensorverdrahtung ILD1302 und ILD1402.....	28
Tabelle 41: Sensorverdrahtung ILD1700 .....	28
Tabelle 42: Sensorverdrahtung ILD2200 .....	29
Tabelle 43: Sensorverdrahtung ILD2300 .....	29
Tabelle 44: Encoder-Interface .....	30
Tabelle 45: Optokoppler I/O .....	31





MICRO-EPSILON MESSTECHNIK GmbH & Co. KG  
Königbacher Str. 15 · 94496 Ortenburg / Deutschland  
Tel. +49 (0) 8542 / 168-0 · Fax +49 (0) 8542 / 168-90  
info@micro-epsilon.de · www.micro-epsilon.de

X9750216-B041043MSC

