



Es gelten die Allgemeinen Lieferbedingungen für Erzeugnisse und Leistungen der Elektroindustrie, herausgegeben vom Zentralverband Elektroindustrie (ZVEI) e.V. in ihrer neusten Fassung sowie die Ergänzungsklausel: "Erweiterter Eigentumsvorbehalt".

<b>1</b>	<b>Einleitung .....</b>	<b>5</b>
<b>2</b>	<b>Konformitätserklärung .....</b>	<b>6</b>
<b>3</b>	<b>Sicherheit .....</b>	<b>7</b>
3.1	Sicherheitsrelevante Symbole .....	7
3.2	Bestimmungsgemäße Verwendung .....	7
3.3	Allgemeine Sicherheitshinweise.....	7
<b>4</b>	<b>Produktbeschreibung .....</b>	<b>8</b>
4.1	Bogenidentifikations-Sensor - Anwendung und Einsatzgebiete .....	8
4.2	Anzeigen und Bedienelemente .....	9
4.3	Schnittstellen und Anschlüsse .....	9
4.4	Lieferumfang .....	11
4.5	Zubehör .....	12
4.5.1	Spannungsversorgung .....	12
4.5.2	Netzwerkkabel.....	12
4.5.3	Eingänge und Ausgänge .....	12
<b>5</b>	<b>Installation.....</b>	<b>13</b>
5.1	Vorbereitung .....	13
5.2	Montage des Geräts .....	13
5.3	Anschluss des Geräts.....	14
5.4	Lagern und Transportieren .....	15
<b>6</b>	<b>Inbetriebnahme .....</b>	<b>16</b>
6.1	Zuweisung einer IP-Nummer zu einer Netzwerkverbindung unter WindowsXP .....	16
6.2	Zuweisung einer IP-Nummer zu einer Netzwerkverbindung unter Windows7 .....	18
6.3	Sensor Start .....	20
6.4	Funktionsweise .....	20
6.4.1	Bildvergleich.....	21
6.4.2	Codevergleich .....	21
6.5	Einlernvorgang.....	22
6.6	Zusatzfunktionen .....	23

<b>7</b>	<b>Bedienung.....</b>	<b>24</b>
7.1	Betriebszustand Inspektion .....	24
7.2	Betriebszustand Einlernen .....	25
7.3	Netzwerkschnittstelle .....	26
7.4	Software-Schnittstelle.....	27
7.4.1	Konfigurationsübersicht .....	34
7.4.2	Resultübersicht.....	37
<b>8</b>	<b>Software Vision-Configurator .....</b>	<b>40</b>
8.1	Aufbau des Anwendungsfensters.....	41
8.2	Menüleiste.....	42
8.2.1	Menü File.....	42
8.2.2	Menü View.....	43
8.2.3	Menü Sensor .....	43
8.2.4	Menü Image.....	44
8.2.5	Menü Administration .....	44
8.2.6	Menü Help .....	45
8.3	Parametrierbereich.....	45
8.3.1	System Karteikarte .....	45
8.3.2	Camera Karteikarte.....	47
8.3.3	Teach Karteikarte.....	47
8.3.4	Search Karteikarte .....	48
8.3.5	Barcode Karteikarte .....	49
8.3.6	Data Matrix Karteikarte .....	52
8.3.7	Rotary Encoder Karteikarte .....	53
8.4	Symbolleiste.....	54
8.5	Maske Sensor Data .....	55
8.6	Maske Sensor Output .....	55
8.7	Bildanzeige .....	56
<b>9</b>	<b>Wartung und Reparatur .....</b>	<b>58</b>
9.1	Wartung .....	58
9.2	Reparatur .....	58
<b>10</b>	<b>Störungsbeseitigung .....</b>	<b>59</b>
10.1	Was tun im Fehlerfall.....	59
<b>11</b>	<b>Anhang.....</b>	<b>60</b>
11.1	Technische Daten.....	60

# 1

## Einleitung

### Herzlichen Glückwunsch

Sie haben sich für ein Gerät von Pepperl+Fuchs entschieden. Pepperl+Fuchs entwickelt, produziert und vertreibt weltweit elektronische Sensoren und Interface-Bausteine für den Markt der Automatisierungstechnik.

### Verwendete Symbole

Dieses Handbuch enthält die folgenden Symbole:



#### ***Hinweis!***

Neben diesem Symbol finden Sie eine wichtige Information.



#### **Handlungsanweisung**

Neben diesem Symbol finden Sie eine Handlungsanweisung.

#### **Kontakt**

Wenn Sie Fragen zum Gerät, Zubehör oder weitergehenden Funktionen haben, wenden Sie sich bitte an:

Pepperl+Fuchs GmbH  
Lilienthalstraße 200  
68307 Mannheim  
Telefon: 0621 776-1111  
Telefax: 0621 776-271111  
E-Mail: [fa-info@de.pepperl-fuchs.com](mailto:fa-info@de.pepperl-fuchs.com)

## 2

## Konformitätserklärung

Dieses Produkt wurde unter Beachtung geltender europäischer Normen und Richtlinien entwickelt und gefertigt.

**Hinweis!**

Eine Konformitätserklärung kann beim Hersteller angefordert werden.

Der Hersteller des Produktes, die Pepperl+Fuchs GmbH in D-68307 Mannheim, besitzt ein zertifiziertes Qualitätssicherungssystem gemäß ISO 9001.



## 3 Sicherheit

### 3.1 Sicherheitsrelevante Symbole



#### **Gefahr!**

Dieses Symbol kennzeichnet eine unmittelbar drohende Gefahr.

Bei Nichtbeachten drohen Personenschäden bis hin zum Tod.



#### **Warnung!**

Dieses Zeichen warnt vor einer möglichen Störung oder Gefahr.

Bei Nichtbeachten drohen Personenschäden oder schwerste Sachschäden.



#### **Vorsicht!**

Dieses Zeichen warnt vor einer möglichen Störung.

Bei Nichtbeachten können Geräte oder daran angeschlossene Systeme und Anlagen bis hin zur völligen Fehlfunktion gestört werden.

### 3.2 Bestimmungsgemäße Verwendung

Das BIS-System ist ein Bogenidentifikations-Sensor und nur für den Zweck der Identifikation von Bogenfolgen konzipiert. Der Sensor beinhaltet Kamera, Beleuchtungseinheit und Auswerterechner mit digitalen Eingangs- und Ausgangssignalen sowie einer Netzwerk-Schnittstelle.

### 3.3 Allgemeine Sicherheitshinweise

Betreiben Sie das Gerät ausschließlich wie in dieser Anleitung beschrieben, damit die sichere Funktion des Geräts und der angeschlossenen Systeme gewährleistet ist. Der Schutz von Betriebspersonal und Anlage ist nur gegeben, wenn das Gerät entsprechend seiner bestimmungsgemäßen Verwendung eingesetzt wird.

Die Verantwortung für das Einhalten der örtlich geltenden Sicherheitsbestimmungen liegt beim Betreiber.

Die Installation und Inbetriebnahme aller Geräte darf nur durch eingewiesenes Fachpersonal durchgeführt werden.

Eigene Eingriffe und Veränderungen sind gefährlich und es erlischt jegliche Garantie und Herstellerverantwortung. Falls schwerwiegende Störungen an dem Gerät auftreten, setzen Sie das Gerät außer Betrieb. Schützen Sie das Gerät gegen versehentliche Inbetriebnahme. Schicken Sie das Gerät zur Reparatur an Pepperl+Fuchs.

## 4 Produktbeschreibung

### 4.1 Bogenidentifikations-Sensor - Anwendung und Einsatzgebiete

Der Bogenidentifikations-Sensor von Pepperl+Fuchs ermöglicht eine schnelle und einfache Überwachung der korrekten Bogenfolge z.B. in Zusammentrag-, Falz- und Bindemaschinen. Die Überwachung kann entweder durch einen Bildvergleich oder durch Einlesen von auf den Bögen aufgedruckten Barcodes erfolgen. Die Identifikation erfolgt schritt haltend im Maschinentakt bei Papiergeschwindigkeiten von bis zu 4 m/s und max. 10 Bögen/s. Der Sensor erhält ein Triggersignal und liefert digitale Signale zurück, ob der aktuelle Bogen dem eingelernten Muster entspricht. Das Triggersignal kann mittels eines Encodersignals wegsynchron verzögert werden. Das Gerät kann sowohl lokal ("stand alone") als auch im Verbund mit mehreren Sensoren über ein Netzwerk betrieben werden. Der Sensor beinhaltet Kamera, Beleuchtungseinheit und Auswerterechner mit digitalen Ein- und Ausgangssignalen sowie einer Netzwerk-Schnittstelle. Die Anbindung des Sensors an eine Maschinensteuerung erfolgt über diese Schnittstellen. Die zeitkritischen Signale wie Trigger und Ergebnis werden über die digitalen Ein-Ausgänge übertragen, über die Netzwerk-Schnittstelle können Kommandos zur Betriebsart-Umschaltung und zur Einstellung verschiedener Parameter an den Sensor geschickt sowie Bilder geladen werden



Der Bogenidentifikations-Sensor ist kompakt in einer Komponente verbaut: Kamera und Beleuchtungseinheit sowie eine DSP-Platine zur Digitalisierung und Weiterverarbeitung der aufgenommenen Bildinformationen finden in einem Gehäuse Platz.

## 4.2 Anzeigen und Bedienelemente

Auf der Beleuchtungseinheit sind 7 Anzeige-LEDs angebracht, die Sie über die verschiedenen Status des Gerätes informieren.

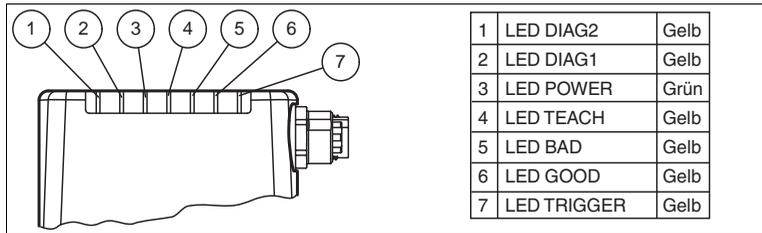


Abbildung 4.1 Anzeigen und Bedienelemente

- 1 **DIAG 2**  
gelbe LED. Signalisiert durch unterschiedliche Blinktakte Diagnosemeldungen.
- 2 **DIAG 1**  
gelbe LED. Signalisiert durch unterschiedliche Blinktakte Diagnosemeldungen.
- 3 **Power (PWR)**  
Leuchtet grün, wenn der Sensor betriebsbereit ist.
- 4 **Einlernen (Teach)**  
Leuchtet gelb, wenn der Teachvorgang gestartet ist.
- 5 **Ausführung der Lesung (BAD)**  
Leuchtet gelb, wenn die Lesung nicht erfolgreich war.
- 6 **Ausführung der Lesung (GOOD)**  
Leuchtet gelb, wenn die Lesung erfolgreich war.
- 7 **Triggersensor (TRG)**  
Leuchtet gelb, sobald ein angeschlossener Triggersensor auslöst.

## 4.3 Schnittstellen und Anschlüsse

Folgende Geräteanschlüsse befinden sich am Gerät:

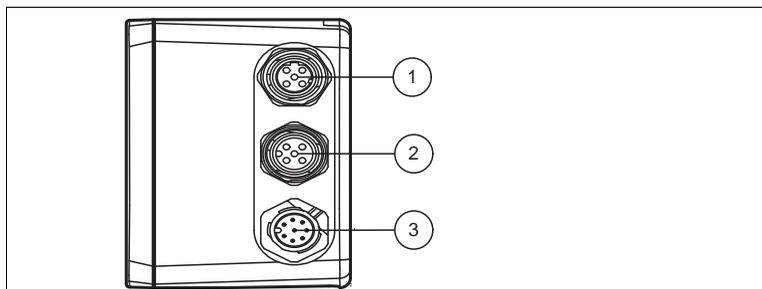


Abbildung 4.2 Geräteanschlüsse

- ① Netzwerk (4-poliger M12-Buchse)
- ② Eingang IO (5-polige M12-Buchse)
- ③ Spannungsversorgung, Eingänge und Ausgänge (8-poliger M12-Stecker)

### Spannungsversorgung

An der Gehäuseseite befindet sich eine 8-poliger M12-Stecker für den Anschluss der Spannungsversorgung und der Eingänge und Ausgänge. Die Pin-Belegung entnehmen Sie der folgenden Grafik:



Abbildung 4.3 Anschlussbelegung Betriebsspannung und Eingänge und Ausgänge

- 1 IN Trigger
- 2 +UB
- 3 OUT Good
- 4 OUT Bad
- 5 IN Start\_Teach (Durch Anlegen eines High-Pegels wird ein Einlernvorgang gestartet)
- 6 OUT Ready
- 7 GND
- 8 OUT Teach\_Active

### Eingang IO und Eingang Drehgeber

An der Gehäuseseite des Sensors befindet sich eine 5-polige M12-Buchse. Je nach Version kann hier ein Drehgeber oder auch ein Triggersensor angeschlossen werden. Die Pin-Belegung entnehmen Sie der folgenden Grafik:



Abbildung 4.4 Anschlussbelegung Eingang IO

- 1 24 V Versorgung
- 2 IN1
- 3 Masse (GND)
- 4 IN2
- 5 nicht belegt



Abbildung 4.5 Anschlussbelegung Eingang Drehgeber

- 1 24 V Versorgung
- 2 B
- 3 Masse (GND)
- 4 A
- 5 nicht belegt

### Netzwerk

An der Gehäuserückseite befindet sich eine 4-polige M12-Buchse für den Anschluss des Netzwerkes. Die Pin-Belegung entnehmen Sie der folgenden Grafik:



Abbildung 4.6 Anschlussbelegung Netzwerk

- 1 TX+ Ethernet
- 2 RX+ Ethernet
- 3 TX- Ethernet
- 4 RX- Ethernet

## 4.4

### Lieferumfang

- BIS510-\*

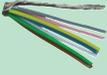
## 4.5 Zubehör

Es steht Ihnen verschiedenes Zubehör zur Verfügung.

### 4.5.1 Spannungsversorgung

Zum Anschluss der Spannungsversorgung, Eingänge und Ausgänge an den Sensor verwenden Sie folgende Verbindungskabel.

#### M12-Verbindungsleitungen

	Material	Länge	Kabelende, konfektionierbar 
 M12-Buchse, gerade, 8-polig	PUR	2 m	V19-G-2M-PUR-ABG
		5 m	V19-G-5M-PUR-ABG
		10 m	V19-G-10M-PUR-ABG

#### Konfektionierbare M12-Steckverbinder

Bestellbezeichnung	Beschreibung	mm <sup>2</sup>	Kabel-Ø
V19-G-ABG-PG9	gerade M12-Buchse, 8-polig	max. 0,75	5 ... 8 mm

Weitere und abweichende Längen auf Anfrage.

### 4.5.2 Netzwerkkabel

Der Sensor wird über einen M12-Stecker mit dem Netzwerk verbunden.

Bezeichnung	Beschreibung
V45-G	RJ45-Netzwerkstecker, konfektionierbar
V1S-G	M12-Stecker, 4-polig, konfektionierbar
V1SD-G-2M-PUR-ABG-V45X-G	Verbindungsleitung, RJ45-Netzwerkstecker mit M12-Stecker, gekreuzt, 4polig
V1SD-G-2M-PUR-ABG-V45-G	Verbindungsleitung, RJ45-Netzwerkstecker mit M12-Stecker, 4polig

### 4.5.3 Eingänge und Ausgänge

Die Eingänge und Ausgänge des Sensors werden über einen M12-Stecker verbunden.

Bezeichnung	Beschreibung
V15S-G-5M-PUR-ABG	Kabelstecker, M12, 5-polig, PUR-Kabel, Schirm auf Überwurfmutter

## 5 Installation

### 5.1 Vorbereitung



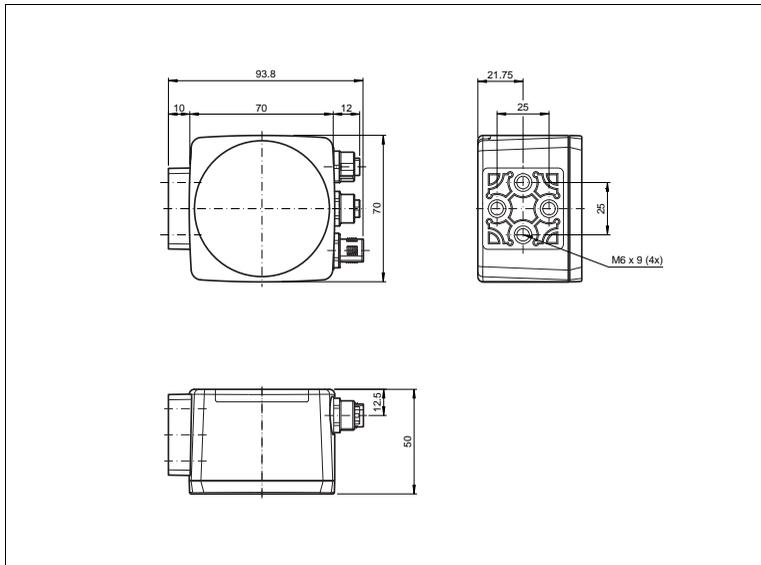
#### Gerät auspacken

1. Überprüfen Sie Verpackung und Inhalt auf Beschädigung.
  - ↳ Benachrichtigen Sie bei Beschädigung den Spediteur und verständigen Sie den Lieferanten.
2. Überprüfen Sie den Lieferumfang anhand Ihrer Bestellung und der Lieferpapiere auf Vollständigkeit und Richtigkeit.
  - ↳ Bei auftretenden Fragen wenden Sie sich bitte an Pepperl+Fuchs.
3. Bewahren Sie die Originalverpackung für den Fall auf, dass das Gerät zu einem späteren Zeitpunkt eingelagert oder verschickt werden muss.

### 5.2 Montage des Geräts

Zur einfachen Montage des Sensors in Ihrer Anlage verfügt das Gerät am Gehäuseboden über vier symmetrisch angeordnete M6-Gewinde.

Die folgende Abbildung zeigt alle relevanten Abmaße des Gehäuses in mm:



Der Vision-Sensor wird im spezifizierten Arbeitsabstand über der Bogen-Oberfläche montiert.

Die folgende Abbildung zeigt eine Beispiel-Montage:

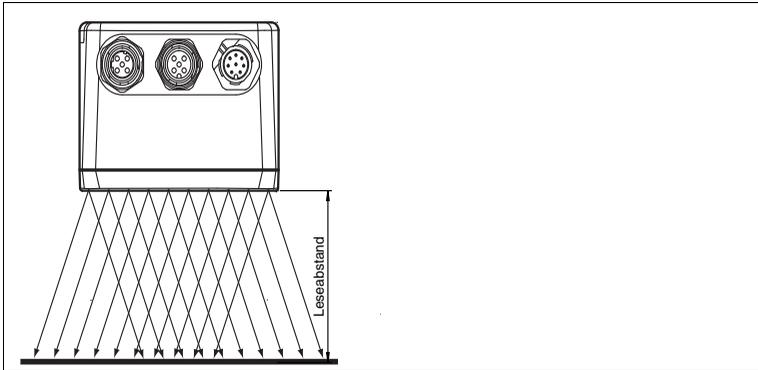


Abbildung 5.1 Montagebeispiel BIS

## 5.3

### Anschluss des Geräts



#### Versorgungsspannung anlegen

Um den Sensor mit Spannung zu versorgen, gehen Sie wie folgt vor:

1. Stecken Sie die Buchse M12, 8-polig in den dafür vorgesehenen Stecker an der Gehäuseseite.
2. Drehen Sie die Überwurfmutter über den Steckverbinder, bis zum Endanschlag.

↳ Damit ist das Versorgungskabel gegen versehentliches Herausziehen gesichert.



#### **Hinweis!**

##### **Netzwerkconfiguration dokumentieren**

Der Sensor kommuniziert mit der angeschlossenen Maschinensteuerung über das TCP/IP-Protokoll. Um eine korrekte Kommunikation zu gewährleisten, notieren Sie sich unbedingt alle Änderungen, die Sie an der Netzwerkconfiguration vornehmen.



#### **Hinweis!**

##### **Netzwerkverkabelung**

Benutzen Sie für eine Direktverkabelung des Sensors mit einem Rechner ein Crossover-Netzwerkabel. Falls Sie den Sensor im Netzwerk betreiben, benutzen Sie für den Anschluss im Netzwerk ein Twisted-Pair-Netzwerkabel.



### Netzwerkverbindung herstellen

Um eine Netzwerkverbindung herzustellen, gehen Sie wie folgt vor:

1. Falls Sie ein Netzkabel verwenden, das auf einer Seite einen RJ45 Netzwerkstecker und auf der anderen Seite eine M12 Buchse 4-polig hat, dann stecken Sie die Buchse M12, 4-polig in den Stecker auf der Seite des Sensors.
2. Im Auslieferungszustand besitzt der Sensor eine feste IP-Adresse (192.168.2.3). Um eine Kommunikation im Netzwerk zu ermöglichen, konfigurieren Sie Ihr Netzwerk. Entnehmen Sie die Konfigurationsdaten der Netzwerk-Konfigurationsübersicht.



### IP-Adresse zurücksetzen

Falls die IP-Adresse des Vision Sensors geändert wurde, können Sie die Default-IP-Adresse einstellen

1. Verbinden Sie den Vision Sensor mit der Spannungsversorgung.  
↳ Der Vision Sensor fährt hoch.
2. Warten Sie, bis die LEDs blinken.
3. Betätigen Sie die Bedientasten 1 & 2 gleichzeitig für ca. 2 Sekunden.  
↳ Die Blinksequenz der LEDs ändert sich.  
Der Vision Sensor fährt erneut mit der Default-IP-Adresse hoch.



### Triggersensor anschließen

Um einen Triggersensor anzuschließen, gehen Sie wie folgt vor:

1. Stecken Sie den 5-poligen M12-Stecker in die dafür vorgesehene Buchse am Gehäuse
2. Drehen Sie den Gewindenippel über den Steckverbinder, bis zum Endanschlag.  
↳ Damit ist das Triggerkabel gegen versehentliches Herausziehen gesichert.

## 5.4

### Lagern und Transportieren

Verpacken Sie das Gerät für Lagerung und Transport stoßsicher und schützen Sie es gegen Feuchtigkeit. Optimalen Schutz bietet die Originalverpackung. Beachten Sie darüber hinaus die zulässigen Umgebungsbedingungen, die Sie im Technischen Datenblatt ablesen können.

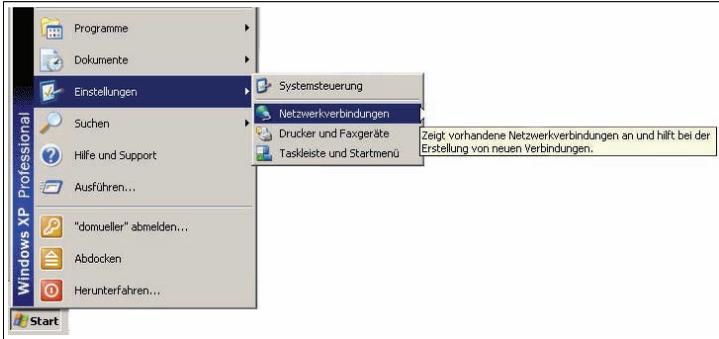
## 6 Inbetriebnahme

### 6.1 Zuweisung einer IP-Nummer zu einer Netzwerkverbindung unter WindowsXP



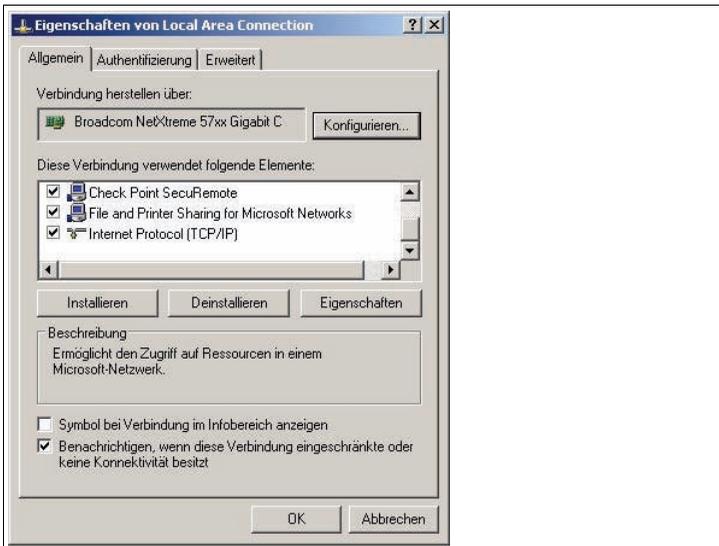
Um eine IP-Nummer unter WindowsXP einer Netzwerkverbindung zuzuweisen, gehen Sie wie folgt vor:

1. Wählen Sie zuerst **"Netzwerkverbindungen"** an:



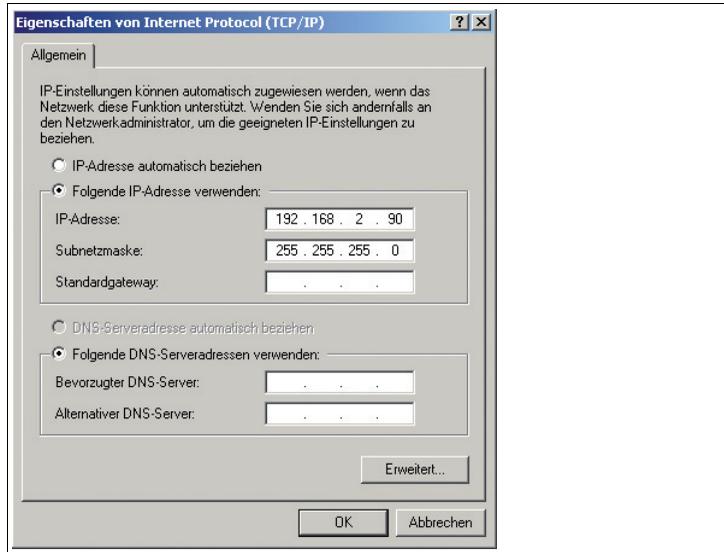
2. Öffnen Sie anschließend die gewünschte Verbindung per Doppelclick.

↳ Es erscheint der Eigenschaften-Dialog der betreffenden Verbindung:



3. Wählen Sie im Eigenschaften-Dialog das Element **"Internetprotokoll (TCP/IP)"** per Doppelclick aus.

↳ Der TCP/IP-Eigenschaften-Dialog erscheint.



4. Aktivieren Sie im TCP/IP-Eigenschaften-Dialog **"Folgende IP-Adresse verwenden"**.
5. Tragen Sie eine IP-Adresse ein, die sich nur im hintersten Segment von der Sensor-IP-Adresse unterscheidet.
6. Tragen Sie als Subnetzmaske 255.255.255.0 ein.
7. Bestätigen Sie anschließend die Eingaben sowohl auf den TCP/IP-Eigenschaften als auch auf den LAN-Verbindungseigenschaften mit **"OK"** und **"Schließen"**.

↳ Die Netzwerkkonfiguration ist damit abgeschlossen und der Sensor kann verwendet werden.

## 6.2 Zuweisung einer IP-Nummer zu einer Netzwerkverbindung unter Windows7



Um eine IP-Nummer unter Windows7 einer Netzwerkverbindung zuzuweisen, gehen Sie wie folgt vor:

1. Wählen Sie zuerst **"Systemsteuerung"** unter dem Menüpunkt **Start** an:

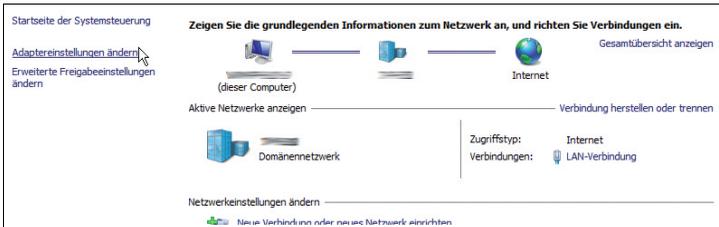


2. Wählen Sie den Menüpunkt **Netzwerk- und Freigabecenter**:



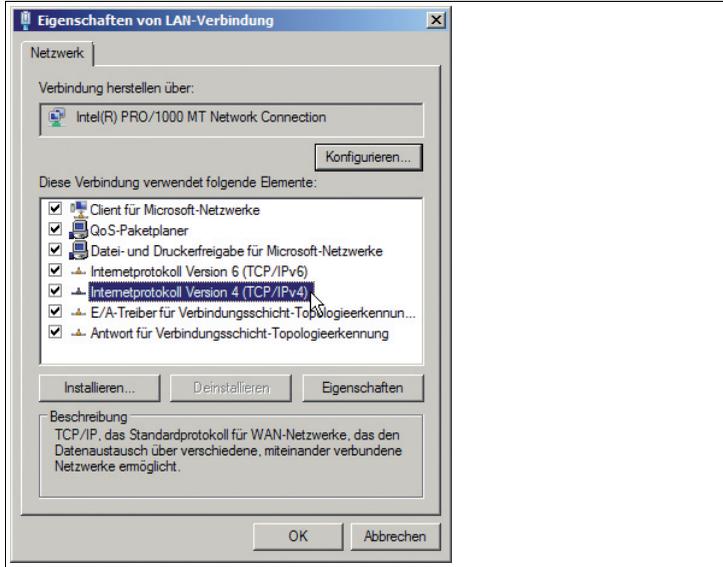
**Netzwerk- und Freigabecenter**

3. Wählen Sie in dem Fenster Netzwerk- und Freigabecenter den Menüpunkt **Adaptoreinstellungen ändern**



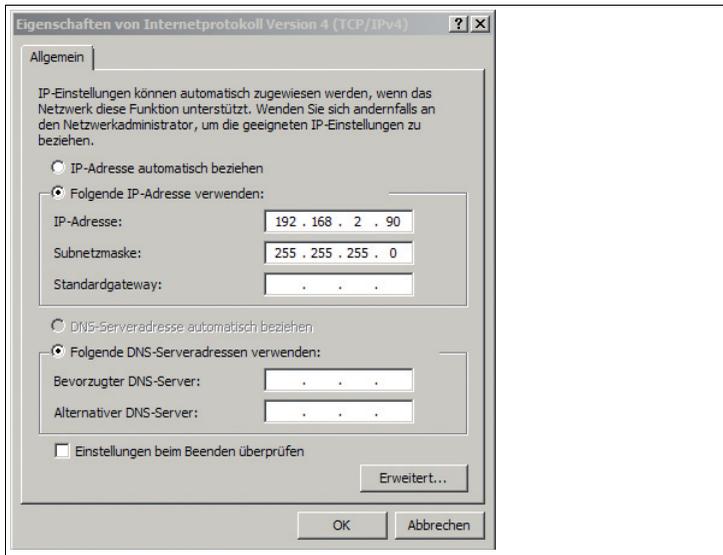
4. Öffnen Sie anschließend die gewünschte Verbindung per Doppelklick.

↳ Es erscheint der Eigenschaften-Dialog der betreffenden Verbindung:



5. Wählen Sie im Eigenschaften-Dialog das Element "**Internetprotokoll Version 4 (TCP/IPv4)**" per Doppelklick aus.

↳ Der TCP/IP-Eigenschaften-Dialog erscheint.



6. Aktivieren Sie im TCP/IP-Eigenschaften-Dialog "**Folgende IP-Adresse verwenden**".

7. Tragen Sie eine IP-Adresse ein, die sich nur im hintersten Segment von der Sensor-IP-Adresse unterscheidet.
8. Tragen Sie als Subnetmaske 255.255.255.0 ein.
9. Bestätigen Sie anschließend die Eingaben sowohl auf den TCP/IP-Eigenschaften als auch auf den LAN-Verbindungseigenschaften mit "OK" und "Schließen".

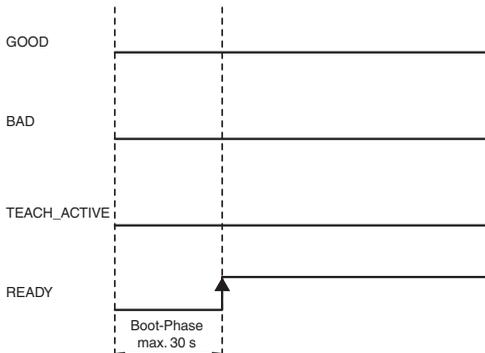
↳ Die Netzwerkkonfiguration ist damit abgeschlossen und der Sensor kann verwendet werden.

## 6.3

### Sensor Start

#### Startsequenz

Die Ausgänge GOOD, BAD, TEACH\_ACTIVE und READY sind bei dem Start des Sensors auf Low-Pegel. Sobald der Sensor betriebsbereit ist (max. 30 Sekunden), nimmt der Ausgang READY den High-Pegel an. Der Ausgang READY geht wieder auf Low-Pegel bei Auftreten eines Hardware-Fehlers.



## 6.4

### Funktionsweise

Der Sensor nimmt von jedem zu prüfenden Bogen mindestens ein Bild auf. Dieses Bild wird abhängig von der Betriebsart entweder mit einem vorher eingelernten Bild verglichen ("Bildvergleich") oder es wird ein im Bild befindlicher Code gelesen und dessen Inhalt mit einem eingelernten oder vorgegebenen Code verglichen (Codevergleich).

Folgende Betriebsarten sind möglich:

- Bildvergleich
- Codevergleich

#### 6.4.1 Bildvergleich

Bei dem Bildvergleich wird ein Musterbogen eingelernt und alle folgenden Bögen werden mit dem eingelernten Bild verglichen. Bei ausreichender Übereinstimmung wird "Gut" ausgegeben, ansonsten "Schlecht". Bei dem Bildvergleich ist eine Verschiebung des Bogens gegenüber dem eingelernten Bogen von bis zu 10 mm in horizontaler Richtung und vertikaler Richtung zulässig.

#### 6.4.2 Codevergleich

An dem zu überprüfenden Bogen müssen an der unteren oder oberen Bogenkante ein Code aufgedruckt sein. Dieser kann vom Sensor gelesen werden. Der Code muss sich dazu vollständig im Bildfeld des Sensors befinden. Für eine sichere Decodierung müssen die Barcode-Module ausreichende Kontrastwerte zum Hintergrund aufweisen. Hierzu wird von Pepperl+Fuchs eine Grauwert- Differenz festgelegt (z.B. 20 Graustufen). Da eine solche Angabe für den Endanwender nicht nachvollziehbar ist, müssen kritische Fälle im Einzelfall getestet werden. Der Code kann sowohl horizontal als auch vertikal im Bildfeld orientiert sein. Der Sensor bietet zwei Möglichkeiten den Code zu lesen. Einmal der **Codevergleich** oder die **Codelesung**. Wobei die Codelesung nur optional verfügbar ist.

- **Codevergleich:** Der Barcode wird mit einem vorgegeben Barcode verglichen. Der Sensor meldet als Ergebnis "Gut" wenn der gelesene Barcode mit dem gespeicherten Barcode übereinstimmt. Der Sensor meldet als Ergebnis "Schlecht" wenn beide Barcodes nicht übereinstimmen oder kein Barcode gelesen werden konnte. Der Vergleichsbarcode kann entweder als Wert vorgegeben werden oder durch Einlernen eines Bogens mit Barcode festgelegt werden.

Die folgenden Abmessungen des zu lesenden Barcodes müssen eingehalten werden:



- Minimale Strichbreite: 0,3 mm

Folgende Barcode-Typen können gelesen werden:

- 2/5 interleaved
- Code39
- Code128
- Pharmacode
- Data Matrix (optional)

## 6.5 Einlernvorgang

Der Einlern- oder Teachvorgang wird durch Setzen des Eingangs **START\_TEACH** oder durch ein XML-Kommando aktiviert. Der Sensor zeigt den Einlernvorgang durch Setzen des Signals **TEACH\_ACTIVE** an. Zum Einlernen werden ein oder mehrere Bilder aufgenommen. Der Abschluss des Lernvorgangs wird durch Rücksetzen des Ausgangs **TEACH\_ACTIVE** angezeigt.

Beim Einlernen laufen folgende Vorgänge ab:

- Bestimmung der Bogenlänge
- Bestimmung der optimalen Belichtung (Blitzzeit)
- Auswahl einer geeigneten Bildaufnahme-Position auf dem Bogen
- Berechnung des Code-Inhaltes bzw. des Musters aus der optimalen Bildaufnahme-Position

Folgende Einlern-Modi können ausgewählt werden:

- **Single image (external trigger):** Das nächste Bild wird mit der Standard-Blitzzeit eingelernt.
- **Single fix position (encoder trigger + external trigger):** Bildaufnahme an einer fest vorgegebenen Position. Die Einstellungen hierfür werden im Rotary Encoder Tab eingegeben. Die Blitzposition wird gegenüber der Blattkante von einem Encoder verzögert. Zur Detektion der Blattkante wird ein externer Trigger benötigt. Erst ab der Triggerung werden die Encodersignale ausgewertet.
- **Triple fix position (encoder trigger):** entspricht dem Single fix position, mit Ausnahme dass 1 Bild pro Bogen bei 3 Bögen aufgenommen werden. Dabei wird die Belichtungszeit variiert und das beste Bild als Referenzbild verwendet.
- **Automatic position (encoder trigger):** Auf 3 unterschiedlich belichteten Bögen werden jeweils über die Länge verteilt max. 5 Bilder aufgenommen. Aus allen aufgenommenen Bildern wird das sowohl von der Aussteuerung als auch vom Inhalt optimale Bild ausgesucht und für den Vergleich als Muster eingelernt.
- **Triple image (external trigger):** An der Triggerposition werden 3 Bilder aufgenommen und so die optimale Belichtungszeit ermittelt und mit dieser das Referenzbild eingelernt.
- **Direct Teach (no trigger):** Mit Setzen des Teachsignals wird je nach Modus gleichzeitig ein oder drei Bildaufnahmen ausgelöst. Dafür ist kein externer Trigger notwendig.

Es können während des Einlernvorgangs insgesamt maximal 15 Bilder aufgenommen werden, aus denen als optimales Bild jenes ausgesucht wird, welches die größte Anzahl eindeutiger Kanten aufweist. Wenn keines der zum Einlernen aufgenommenen Bilder eine ausreichende Anzahl Kanten enthält, arbeitet der Sensor im sog. Grauwert-Modus. Dabei werden im Musterbild Sollwerte und Toleranzen für folgende statistische Merkmale bestimmt:

- Grauwert-Minimum
- Grauwert-Maximum
- Grauwert-Mittelwert
- Kontrast (Differenz zwischen Maximal- und Minimal-Grauwert)
- Varianz

Bei der Inspektion wird dann überprüft, ob die Werte der Merkmale im aktuellen Bild innerhalb der Toleranzen liegen.

Die eingelernten Daten werden nach dem Einlernvorgang in den Flash-Speicher des Sensors übernommen und sind damit nichtflüchtig im Sensor gespeichert. Nach einer Unterbrechung der Stromversorgung kann dadurch ohne erneuten Einlernvorgang weiter geprüft werden.



### Einlernen eines Musterbogens

1. Starten Sie den Einlernvorgang durch das Senden einer steigende Impulsflanke am **TEACH**- Eingang oder senden Sie das entsprechende XML- Kommando an den Sensor. Der Sensor zeigt den Einlernvorgang durch Setzen des Signals **TEACH\_ACTIVE** an.
2. Durch eine oder mehrerer positive Flanken am Triggereingang, werden je nach Einlern-Betriebsart eine oder mehrere Bildaufnahmen ausgelöst.
3. Der Abschluss des Lernvorgangs wird durch Rücksetzen des Ausgangs **TEACH\_ACTIVE** angezeigt.

↳ Der Einlernvorgang ist somit abgeschlossen.

## 6.6

### Zusatzfunktionen

#### Statistik

Im Sensor werden mehrere Werte protokolliert, die bei jedem Einlernvorgang auf Null zurückgesetzt werden. Die Werte können mit einem Ergebnis-Telegramm ausgelesen werden. Folgendes wird protokolliert:

- Anzahl Bögen
- Anzahl Bögen mit Bildvergleich "Schlecht"
- Anzahl Bögen mit nicht lesbarem Barcode
- Anzahl Bögen mit falschem Barcode

Nach Abschalten der Betriebsspannung stehen die Zähler ebenfalls auf Null.

#### Aktiv-Modus

Im Normalfall ist der BIS510 an der Datenschnittstelle passiv, d.h. er sendet selbsttätig keinerlei Daten und reagiert lediglich auf Anforderungen. Nach Aktivieren des Aktiv-Modus sendet der Sensor nach Detektion von Fehlern selbsttätig das Fehlerbild.

#### Fehlerbildspeicher

Der BIS510 arbeitet mit einem Ringpuffer für Fehlerbilder. Der Ringpuffer kann bis zu 5 Bilder speichern. Bei Auftreten eines neuen Fehlers wird das älteste Bild mit dem neuen Fehlerbild überschrieben.

## 7 Bedienung

### 7.1 Betriebszustand Inspektion

Impulse am Eingang TRIGGER lösen eine Bildaufnahme mit anschließender Prüfung aus. Ein High-Pegel am Ausgang GOOD oder am Ausgang BAD zeigt das Inspektionsergebnis an.

Um eine Überprüfung zu starten gehen Sie wie folgt vor:

#### Bild überprüfen

1. Sobald eine steigende Flanke am Triggersignal anliegt, wird eine Bildaufnahme mit anschließender Prüfung ausgelöst.
2. Nach der Auswertung des Bildes gibt der Ausgang GOOD oder der Ausgang BAD einen High-Pegel aus.

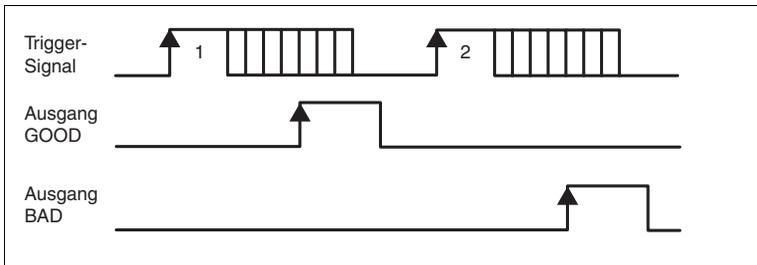


Abbildung 7.1 Signale im Betriebszustand "Inspektion"

Der Sensor kann bis zu 10 Ergebnisse (Gut/Schlecht) in einem Schieberegister zwischenspeichern. D.h. zwischen Sensor und Ausschleusung können sich maximal 10 Produkte befinden.

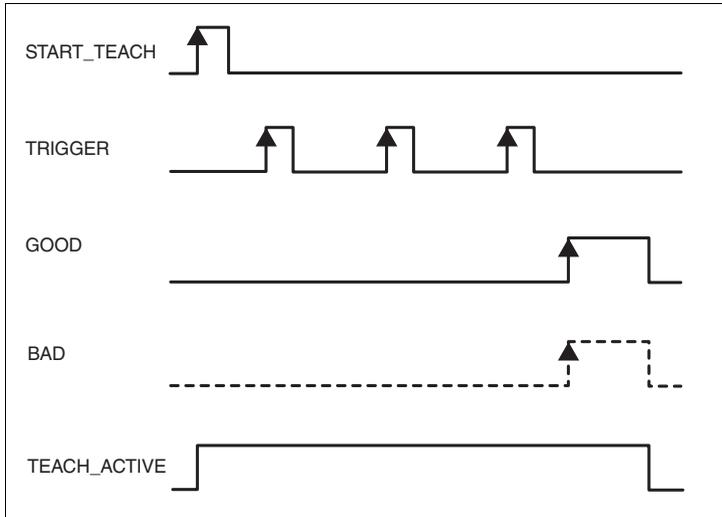
## 7.2 Betriebszustand Einlernen

Um ein Muster einzulernen gehen Sie wie folgt vor:



### Muster einlernen

1. Senden Sie eine positive Flanke am **START\_TEACH** Eingang. Der Sensor wird in den Einlernmodus versetzt und zeigt dies durch setzen des **TEACH\_ACTIVE** Ausgang an.



2. Durch positive Flanken am Triggereingang, werden je nach Einlern-Betriebsart eine oder mehrere Bildaufnahmen ausgelöst.
3. Der Sensor wählt von den aufgenommenen Bildern das Beste aus und lernt dieses als Muster ein.
4. Während des Einlernens ist der Ausgang **TEACH\_ACTIVE** gesetzt.

↳ Der Einlernvorgang ist dann abgeschlossen, wenn der Ausgang **GOOD** oder **BAD** auf High-Pegel gesetzt wird.

"**GOOD**" bedeutet, dass die Bilderkennung erfolgreich war.

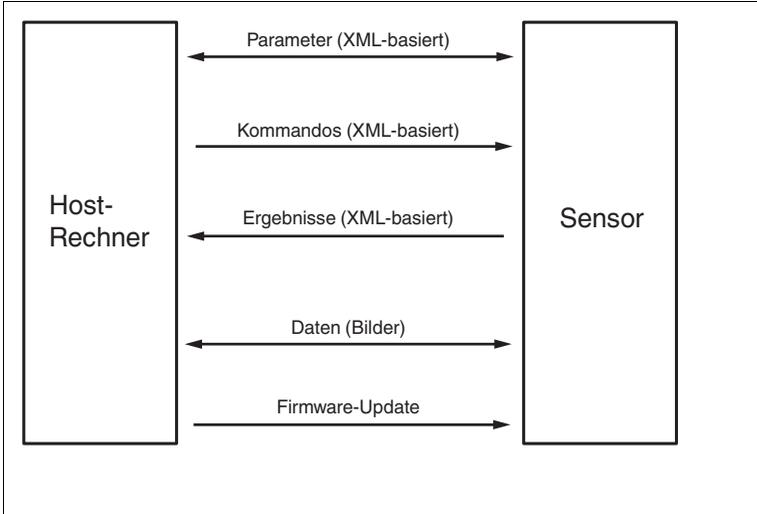
"**BAD**" bedeutet, dass die Bilderkennung nicht funktioniert hat und der Sensor im Grauwert-Vergleichsmodus arbeitet.

Mit dem Rücksetzen von **GOOD** bzw. **BAD** wird auch der Ausgang **TEACH\_ACTIVE** zurück gesetzt.

### 7.3 Netzwerkschnittstelle

Die Netzwerk-Schnittstelle wird zur Übertragung von Daten (Parameter, Bilder, Software-Update) und Kommandos in beide Richtungen (Host zum Sensor und Sensor zum Host) verwendet.

Die Übertragung von Parametern, Kommandos und Ergebnissen erfolgt in Form von XML-Strings.



Die Kommunikation erfolgt mittels TCP/IP über Port 50005. Die Standard-IP-Adresse ist 192.168.2.3.

Jedes Datenpaket besteht aus eine Pre-Header sowie den eigentlichen Daten:

Pre-Header (Länge 12 Byte)	Daten (Länge unterschiedlich)
----------------------------	-------------------------------

## 7.4 Software-Schnittstelle

Zur einfachen Einbindung in eine PC- Software gibt es eine **.NET 2.0 basierende** Softwareschnittstelle. Diese wird im Form von .NET- DLLs zur Verfügung gestellt und übernimmt die Kommunikation zum Sensor. Binden Sie hierfür die Interface-DLLs in die Programmierumgebung ein (→ siehe Abbildung 7.2 auf Seite 27) und führen die darunter erwähnten Programmierzeilen aus.

Alle Beispiele beziehen sich auf die Programmierumgebung Visual- Studio 2008 und auf die Programmiersprache C# (CSharp).

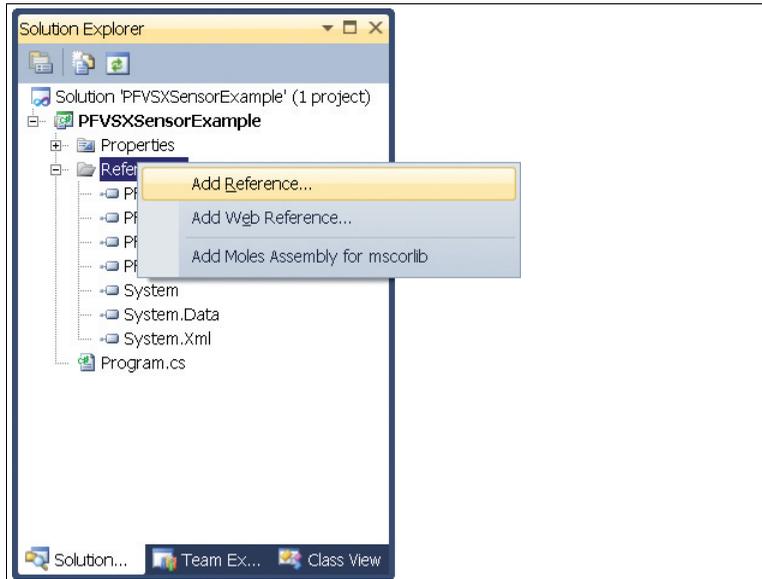


Abbildung 7.2 Einbindung Interface-DLL

Nachfolgend ein Beispielprogramm zur Einbindung und Triggern des Sensors:

```
class Program
{
    static void Main(string[] args)
    {
        PF.Foundation.VsxFactory.PFVsxFactoryVCCustom sensor;
        sensor = new PF.Foundation.VsxFactory.PFVsxFactoryVCCustom();
        sensor.Connect("192.168.2.3", 50005);
        System.Threading.Thread.Sleep(2000); //Wait for Connection
        sensor.SetSingleParameter("Command", "TriggerStart", "1");
        System.Threading.Thread.Sleep(1000);
        sensor.Disconnect();
    }
}
```

## Allgemein

Die Bibliothek dient zur Unterstützung beim Erstellen einer grafischen Bedienoberfläche für Sensoren, welche mit dem VSX-Protokoll arbeiten. Hierzu stellt die Bibliothek eine Verbindung mit dem Sensor her und übernimmt die Kommunikation entsprechend dem Protokoll. Dem Benutzer werden Funktionen zur Verfügung gestellt, mittels welchen Parameter auf dem Sensor eingestellt, Parameterwerte vom Sensor abgefragt und ganze Parametersätze sowohl lokal als auch auf dem Sensor gespeichert und geladen werden können. Weiterhin können Sensorbilder empfangen werden.

Die Bibliothek ist in C# implementiert und benötigt als Voraussetzung .NET 2.0 oder höher.

Stellen Sie sicher, dass sich die mitgelieferten Bibliotheken im Ausführungsordner des Projekts befinden.

## Anlegen eines Objektes

Erstellen Sie ein Objekt, um auf die Funktionen der Bibliothek zugreifen zu können.

```
PFVsxFactoryVCCustom _vsxFactory =new  
PFVsxFactoryVCCustom();
```

## Abfrage empfangener Parameterdaten

Empfangene Parameterdaten werden im Sensor in einer Liste gespeichert. Einzelne Parameterdaten aus dieser Liste können mit folgender Funktion abgefragt werden:

```
string GetSingleParameter(string configId, string  
parameterId)
```

## Sensordaten synchronisieren

Zur Synchronisation der Liste mit den aktuellen Sensordaten, kann folgende Funktion aufgerufen werden:

```
void GetAllParametersFromSensor()
```

## Anzeige geänderter Daten

Werden Daten vom Sensor empfangen und eine Änderung in der internen Liste findet statt, so wird dies durch das Event angezeigt:

```
event ParameterDataReceived(DataModifier modifier)
```

## Bibliotheksfunktionen

```
bool Connect(string ip,int port)
```

ip: IP des angeschlossenen Sensors  
port: Port des angeschlossenen Sensors  
**Antwort:** false falls Verbindung nicht erstellt werden konnte, ansonsten true  
Öffnet eine Verbindung zu einem Sensor mit gegebener IP und Port.

```
void Disconnect()
```

Trennt die geöffnete Verbindung

```
bool Connected {get;·}
```

**Antwort:** Gibt den Verbindungsstatus zurück

```
void SaveSensorSettings()
```

Speichert den aktuellen Parametersatz im Speicher des Sensors.  
Nachdem eine Bestätigung vom Sensor empfangen wurde, wird ein SaveDataOnSensorReceived Event ausgelöst.

```
void LoadSensorSettings()
```

Fordert vom Sensor einen mit SaveSensorSettings gespeicherten Parametersatz an.  
Nachdem der Parametersatz vom Sensor empfangen wurde, wird ein ParameterDataReceived Event mit Modifier = LOAD\_Data ausgelöst.  
Die Daten können dann über die Methode GetSingleParameter abgerufen werden.

```
void GetNetworkSettings()
```

Fordert vom Sensor dessen aktuelle Verbindungseinstellungen an.  
Nachdem die Daten vom Sensor empfangen wurden, wird ein GetNetworkReceived Event ausgelöst, welchem aktuelle IP Adresse, Netzwerkmaske und Gateway des Sensors entnommen werden können.

```
void GetLogMessages(bool on)
```

on: Schaltet die Log Meldungen ein  
off: Schaltet die Log Meldungen aus  
Legt fest, ob der Sensor Log Meldungen sendet oder nicht. Sind die Log-Meldungen eingeschaltet, wird bei jeder vom Sensor empfangenen Log-Meldung ein LogDataReceived Event ausgelöst.

```
void GetAllParametersFromSensor()
```

Fordert den aktuellen Parametersatz vom Sensor an. Nachdem der Parametersatz vom Sensor empfangen wurde, wird ein ParameterDataReceived Event mit Modifier = None ausgelöst. Dieses signalisiert, dass die interne Liste mit dem Parametersatz aktualisiert wurde.  
Anschließend können einzelne Parameter mit GetSingleParameter abgefragt werden.

```
public bool ExistsParameter(string configId, string parameterId)
```

Dient zum Abfragen, ob ein bestimmter Parameter auf dem Sensor existiert.  
configId: Configuration Id eines Parameters  
parameterId: Parameter Id eines Parameters  
**Antwort:** true oder false, je nachdem ob der Parameter existiert

```
string GetSingleParameter(string configId, string parameterId)
```

Gibt den Wert eines einzelnen Parameters, festgelegt durch die Configuration ID und die Parameter ID zurück. Der Wert wird einer internen Liste entnommen und entspricht dem Stand welcher als letztes vom Sensor empfangen wurde. Um die Liste zu aktualisieren, muss GetAllParametersFromSensor aufgerufen werden.

Ist der angegebene Parameter nicht in der Liste vorhanden, wird ein `InternalErrorEvent` mit `ErrorType = PARAMETER_NOT_FOUND` ausgelöst.

`configId`: Configuration ID eines Parameters

`parameterId`: Parameter ID eines Parameters

**Antwort**: Den aktuellen Wert des bezeichneten Parameters

```
void SetSingleParameter(string configId, string parameterId, string value)
```

`configId`: Configuration ID eines Parameters

`parameterId`: Parameter ID eines Parameters

`value`: Neuer Wert des Parameters

Setzt einen einzelnen Parameter, festgelegt durch die Configuration ID und die Parameter ID, auf den angegebenen Wert. Ändert der Sensor nur diesen einen Wert, so wird ein `SingleDataReceived` Event ausgelöst. Diesem kann der gesetzte Wert entnommen und geprüft werden, ob der Wert auf dem Sensor entsprechend eingestellt werden konnte. In speziellen Fällen kann es auch vorkommen, dass bedingt durch den gesetzten Wert weitere abhängige Parameter, ebenfalls auf dem Sensor geändert werden. In diesem Fall wird ein `ParameterDataReceived` Event mit `Modifier = NONE` ausgelöst und alle Parameter sollten mit `GetSingleParameter` auf neue Werte hin überprüft werden.

Ist der angegebene Parameter nicht in der Liste vorhanden, wird ein `InternalErrorEvent` mit `ErrorType = PARAMETER_NOT_FOUND` ausgelöst.

```
public bool ExistsResultParameter(ushort frameCounter, string configId, string parameterId)
```

`frameCounter`: Nummer des zum Ergebnisparameter gehörenden Bildes

`configId`: Configuration Id eines Parameters

`parameterId`: Parameter Id eines Parameters

**Antwort**: true oder false, je nachdem ob der Parameter existiert

Dient zum Abfragen, ob ein bestimmter Ergebnisparameter zu einem bestimmten Bild existiert.

```
string GetResultParameter(ushort frameCounter, string configId, string parameterId)
```

`configId`: Configuration Id eines Parameters

`parameterId`: Parameter Id eines Parameters

**Antwort**: Den aktuellen Wert des bezeichneten Parameters

Gibt den Wert eines einzelnen Parameters, festgelegt durch die Configuration ID und die Parameter ID zurück. Der Wert wird einer internen Liste entnommen und entspricht dem Stand welcher als letztes vom Sensor empfangen wurde.

Der Empfang von Ergebnisdaten zu einem bestimmten Bild wird durch das event `SensorResultDataReceived(ushort frameCounter)` angezeigt.

```
Bitmap GetImage(ushort frameCounter)
```

`frameCounter`: Nummer eines empfangenen Bildes

**Antwort**: Das zur Bildnummer gehörige Bild

Gibt ein zuvor vom Sensor empfangenes Bild zurück. Der Empfang eines Bildes mit einer bestimmten Nummer wird durch das event

`ImageReceived(ushort frameCounter, Bitmap image)` angezeigt.

```
IList<ElementResult> GetResultList(ushort frameCounter)
```

`frameCounter`: Nummer des zu den Grafiken gehörenden Bildes

**Antwort**: Liste der Overlay Grafiken für das bezeichnete Bild

```
IList<ElementShapeBase> GetShapes(ushort frameCounter)
```

frameCounter: Nummer des zu den Grafiken gehörenden Bildes  
**Antwort:** Liste der Overlay Grafiken für das bezeichnete Bild  
 Gibt eine Liste von Grafiken für ein Bildoverlay zurück. Der Empfang von solchen Grafiken wird durch das event ShapeDataReceived(ushort frameCounter, List<IElement> shapeList) angezeigt. Die einzelnen Elemente der Liste sind vom Typ ElementShapeBase.  
 Diese Klasse verfügt über folgende Attribute:  
 PointF ShapeLocation: Koordinaten der linken oberen Ecke der Grafik im Bild  
 Color ForeColor: Die Farbe der Grafik  
 string Type: Der Typ der Grafik, entweder Type="Rectangle" oder Type="Text". Je nach Typ kann die Grafik in ElementShapeRectangle oder ElementShapeText geparkt werden und verfügt dann jeweils über folgende Zusatzattribute:  
 ElementShapeRectangle:  
 Size Size: Die Größe der Rechteckgrafik  
 ElementShapeText:  
 stringText: Der Text der Textgrafik

```
void ResetSensor()
```

Setzt alle Parameter des Sensors auf deren Standardwerte zurück. Nachdem eine Bestätigung vom Sensor empfangen wurde, wird ein ParameterDataReceived Event mit Modifier = LOAD\_DEFAULT\_DATA ausgelöst.

```
void SaveSettingsToFile(string filename)
```

Speichert den aktuellen Parametersatz in die spezifizierte Datei. Vom Sensor wird der aktuelle Parametersatz abgerufen und nach Erhalt gespeichert. Nach erfolgreichem Speichern wird ein SaveDataOnHdd Event ausgelöst. Tritt beim Speichern ein Fehler auf, wird ein InternalError Event mit ErrorType = SAVE\_FILE\_ERROR ausgelöst.  
 filename: Gültiger Pfad und Dateiname

```
bool SetAllParameters(string filename)
```

Lädt einen Parametersatz aus der angegebenen Datei und sendet die Parameter an den Sensor. Nachdem der Empfang des Datensatzes vom Sensor quittiert wurde, wird ein ParameterDataReceived Event mit Modifier = NONE ausgelöst. Kann der Datensatz nicht ordnungsgemäß geladen werden, wird ein InternalError Event mit ErrorType = LOAD\_FILE\_ERROR ausgelöst.  
 filename: Gültiger Pfad und Dateiname

```
void SetNetworkSettings(string ipAddress, string networkMask, string gateway)
```

ipAddress: Gültige IP Adresse  
 networkMask: Gültige Netzwerkmaske  
 gateway: Gültiges Gateway  
 Stellt die Netzwerkparameter auf dem Sensor um. Die Verbindung zum Sensor wird anschließend getrennt und ein DisconnectReceived Event ausgelöst. Über Connect muss die Verbindung mit der neuen IP neu hergestellt werden.

```
void SendImage(Bitmap image)
```

Sendet ein Bild an den Sensor. Diese Funktion wird nicht von allen Gerätetypen unterstützt.  
 image: zu sendendes Bild

```
void SendVsxFile(string filepath)
```

filepath: Pfad und Dateiname einer gültigen VSX-Datei  
 Sendet den Dateinhalt einer Datei an den Sensor. Dieser muss dem VSX-Standard entsprechen. Können die Daten nicht ordnungsgemäß geladen werden oder entsprechen sie nicht der VSX-Syntax, wird einInternalError Event mit ErrorType = LOAD\_FILE\_ERROR ausgelöst.

```
string SensorName { get; }
```

Gibt den Namen des Sensors zurück.

```
float SensorVsxVersion { get; }
```

Gibt die auf dem Sensor installierte VSX-Version zurück.

## Bibliotheksevents

```
event ParameterDataReceived(DataModifier modifier)
```

Wird immer dann ausgelöst, wenn Parameterdaten vom Sensor empfangen wurden

Dies ist nach folgenden Aufruf der Fall:

```
LoadSensorSettings·(MODIFIER = LOAD_DATA)
GetAllParametersFromSensor·(MODIFIER = NONE)
SetSingleParameter·(MODIFIER = NONE)
ResetSensor·(MODIFIER = LOAD_DEFAULT_DATA)
SetAllParameters·(MODIFIER = NONE)
```

der Fall. Immer wenn dieses Event ausgelöst wurde, wurde die interne Parameterliste aktualisiert und die einzelnen Parameterwerte sollten mit `GetSingleParameter` abgefragt werden.

```
event SensorResultDataReceived(ushort framecounter)
```

Wird ausgelöst, wenn Ergebnisdaten vom Sensor empfangen werden. Parameter ist eine Bildnummer um die Daten ihrem entsprechenden Bild zuzuordnen zu können. Die einzelnen Ergebnisdaten können mit `GetResultParameter` abgefragt werden.

```
event SingleDataReceived(string configId, string parameterId, string value)
```

Wird nach `SetSingleParameter` ausgelöst, falls auf dem Sensor nur dieser eine Wert geändert wurde.

```
event AcceptReceived()
```

Wird nach `SetSingleParameter` ausgelöst, falls auf dem Sensor der zuvor gesetzte Wert erfolgreich geändert wurde.

```
event SensorInformationDataReceived(string type, string version, string macAddress)
```

Wird ausgelöst wenn beliebige Daten vom Sensor empfangen werden. Übergeben werden Angaben zum Sensortyp, dessen Firmware Version und dessen MAC Adresse.

```
event DisconnectReceived(string errorMessage)
```

Wird ausgelöst, wenn die Verbindung nicht durch `Disconnect` getrennt wurde. Übergeben wird eine Beschreibung des Grundes der verlorenen Verbindung.

```
event SaveDataOnSensorReceived()
```

Wird nach erfolgreichem Aufruf von `SaveSensorSettings` ausgelöst.

```
event GetNetworkReceived(string ipAddress, string mask, string gateway)
```

Wird ausgelöst, nachdem die mit `GetNetworkSettings` vom Sensor angeforderten Daten empfangen wurden. Übergeben werden die IP Adresse, die Netzwerkmaske und das Gateway des Sensors.

```
event LogDataReceived(string logData, LogMessageTypes logType)
```

Wird ausgelöst, wenn Logdaten vom Sensor empfangen wurden. Die Daten und der Typ der Logdaten werden als Parameter übergeben.

```
event ShapeDataReceived(ushort frameCounter)
```

Wird ausgelöst, wenn Grafiken für das Bildoverlay empfangen wurden. Die Bildnummer gibt an, zu welchem Bild die Grafiken gehören. Ein Liste aller Grafiken kann dann mit der Funktion `GetShapes(ushort frameCounter)` abgerufen werden.

**Folgendes Event wird nur von Lasertriangulationssensoren unterstützt:**

```
event LineDataReceived(ushort frameCounter, LineMulti lines,
ushort status)
```

Wird ausgelöst, wenn Liniendaten vom Sensor empfangen wurden. Neben den Liniendaten wird die Bildnummer ausgegeben, um die Daten ihrem entsprechenden Bild zuzuordnen zu können. Außerdem wird ein Status ausgegeben, der die Hardware Ein- und Ausgänge des Sensors wieder spiegelt.

```
event ImageReceived(ushort frameCounter)
```

Wird ausgelöst, wenn ein Bild vom Sensor empfangen wurde. Die Bildnummer wird als Parameter übergeben. Das Bild kann anschließend mit der Funktion `GetImage(ushort frameCounter)` abgerufen werden.

```
event SaveDataOnHddReceived()
```

Wird ausgelöst, wenn der aktuelle Parametersatz erfolgreich in eine Datei gespeichert wurde.

```
event ErrorReceived(string id, string name)
```

Wird ausgelöst, wenn ein Fehler in der Sensorsoftware aufgetreten ist. Die Parameter enthalten nähere Angaben zu diesem Fehler.

```
event InternalError(ErrorTypes errorType, string errorMessage)
```

Wird ausgelöst, wenn ein interner Fehler aufgetreten ist. Dies passiert in folgenden Fällen:

- Beim Aufruf von `GetSingleParameter (ErrorType = PARAMETER_NOT_FOUND)` wenn der angeforderte Parameter nicht in der internen Liste gefunden wurde
- Beim Aufruf von `SetSingleParameter (ErrorType = PARAMETER_NOT_FOUND)` wenn der zu setzende Parameter nicht vorhanden ist
- Beim Aufruf von `SaveSettingsToFile (ErrorType = SAVE_FILE_ERROR)` wenn die Parameterdaten nicht in einer Datei gespeichert werden konnten
- Beim Aufruf von `LoadSettingsFromFile (ErrorType = LOAD_FILE_ERROR)` wenn die Parameterdatei nicht geladen werden konnte
- Beim Aufruf von `SetAllParameters (ErrorType = LOAD_FILE_ERROR )` wenn die Parameterdaten nicht aus der Datei geladen werden konnten oder wenn die Datei ungültige Daten enthält
- In allen Fällen in denen `ParameterDataReceived` oder `SensorResultDataReceived` ausgelöst werden würde, die empfangenen Sensordaten jedoch fehlerhaft sind (`ErrorType = INVALID_DATA_RECEIVED`).
- Wenn Logdaten vom Sensor empfangen werden, diese jedoch fehlerhaft sind (`ErrorType = INVALID_DATA_RECEIVED`)
- Beim Aufruf von `SendVsxFile (ErrorType = LOAD_FILE_ERROR)` wenn die Datei nicht geladen werden kann oder nicht der VSX-Syntax entspricht

Der Parameter `errorMessage` enthält eine genauere Beschreibung des jeweiligen Fehlers.

```
enum_ErrorTypes{ INVALID_DATA_RECEIVED, SAVE_FILE_ERROR,
LOAD_FILE_ERROR, PARAMETER_NOT_FOUND }
```

Siehe Event InternalError.

```
enum_DataModifier{ NONE, LOAD_DATA, LOAD_DEFAULT_DATA }
```

Siehe Event ParameterDataReceived.

```
enum_LogMessageTypes { DEBUG, INFO, RESULT_OK, RESULT_NOT_OK,
WARNING, ERROR, CRITICAL, ASSERT }
```

Siehe Event LogDataReceived.

## 7.4.1 Konfigurationsübersicht

Die Parameter in der nachfolgenden Tabelle können mit Hilfe der `SetSingleParameter` und `GetSingleParameter`-Methode des Softwareinterfaces eingestellt werden. Jeder Parameter ist durch eine **configId** und eine **parameterId** gekennzeichnet.

### Konfigurationsparameter

Config ID	Parameter ID	Wert	Beschreibung
General	OptionSendErr	1/0	Automatisches Senden des Fehlerbildes aktivieren/deaktivieren
	OptionAutoResult	1/0	Automatisches Senden des Ergebnisses aktivieren/deaktivieren
	OptionShowHWTrigger	1/0	Log-Eintrag mit Zeitstempel beim Triggersignal aktivieren/deaktivieren
	PosMark	1/0	Bildfeldmarkierung aktivieren/deaktivieren
	IPAdress		Konfiguration der IP-Adresse
	SubNetMask		Konfiguration der SubNet Maske
	Gateway		Konfiguration des Gateways
	EnableBtn1	1/0	Taste 1 auf der Rückseite des Sensors aktivieren/deaktivieren
EnableBtn2	1/0	Taste 2 auf der Rückseite des Sensors aktivieren/deaktivieren	
Camera	FlashTime	0 ... 255	Einstellung der Belichtungszeit und Blitzzeit in µs
	Gain	0 ... 255	Einstellung der Verstärkung
	Snapshot	1/0	Aktiviert die Kameraaufnahme mit den aktuellen Einstellungen
	UseCamSettings	1/0	Benutzt die eingestellten Werte für den nächsten Teachvorgang
	GetCamTeachParams	1/0	Lesung der Werte aus dem Sensor
	StartLive	1/0	Startet die Livebildübertragung
	StopLive	1/0	Stoppt die Livebildübertragung
	CheckImage	1/0	Auswertung auf jedem im Livebild-Modus aufgenommenen Bild aktivieren/deaktivieren
DisplayPause	1 ... 10000	Einstellung der Pausenzeit zwischen zwei Bildaufnahmen	
Search Params	Sensitivity	0 ... 5	Empfindlichkeit des Mustervergleiches Level 0 -> schwach Level 5 -> stark

2012-10

Config ID	Parameter ID	Wert	Beschreibung
Command	TriggerStart	1	Löst eine Bildaufnahme aus
	ReloadData	1	Stellt die eingelernten Daten nach einem TeachStart wieder her
	GetLastImg	1	Aktuelles Bild wird angefordert
	GetErrImg	1	Letztes Fehlerbild wird angefordert
	GetNextErrImg	1	Das nächste Fehlerbild wird angezeigt
	GetErrorImgListSize	1	Anforderung der Anzahl gespeicherte Fehlerbilder
	GetLogMsg	1	Alle Log-Meldungen ausgeben
VOS510	TeachMode	TEACH_MODE_1_1_1	Einlernen eines Bildes auf einem Bogen mit einer festgelegten Belichtungszeit
		TEACH_MODE_IMAGE_AUTO	Einlernen eines Bildes pro Bogen auf insgesamt drei Bögen mit variierender Belichtungszeit. Das beste Bild wird übernommen
		TEACH_MODE_TRIPLE_IMAGE	Einlernen von insgesamt fünf Bilder pro Bogen auf insgesamt drei Bögen mit variierender Belichtungszeit. Das beste Bild wird übernommen
	InspectionMode	PATTERN_INSPECTION	Prüfmodus Bildvergleich
		BARCODE_INSPECTION	Prüfmodus Barcodevergleich
	DirectTeach	0/1	Direct Teach-Mode aktivieren/deaktivieren. Teach-Impuls bzw. Teach-Kommando löst dabei eine Bildaufnahme zum Einlernen aus
	CapturePositionOnPage	1 ... 5	Anzahl Positionen pro Bogen beim automatischen Einlernen
	TeachTimeout	100 ... 1000000	Timeout fürs Einlernen (Bildvergleich und Barcode)

Config ID	Parameter ID	Wert	Beschreibung	
Barcode	CompareCode		Eingabe des Vergleich-Strings des Referenz-Codes	
	Update	0/1	Setzt den zuletzt gelesenen Barcode als Referenz aktivieren/deaktivieren	
	Timeout	1 ... 1000000	Timeout für Barcode-Lesung und Barcode-Vergleich	
	StartAutomatic	0/1	Aktivierung oder Deaktivierung der automatischen Suche	
	Orientation	hor		Barcode- und Pharmacodesuche ausschließlich horizontal
		ver		Barcode- und Pharmacodesuche ausschließlich vertikal
	Code39	0/1	Aktivierung oder Deaktivierung Code39	
	Code128	0/1	Aktivierung oder Deaktivierung Code128	
	Code13	0/1	Aktivierung oder Deaktivierung Code13	
	Code25	0/1	Aktivierung oder Deaktivierung Code25	
	Checksum	0/1	Aktivierung oder Deaktivierung der Prüfsummen-Auswertung bei Code 2/5 interleaved	
	PharmaCodeOnly	0/1	Aktivierung oder Deaktivierung Pharmacode. Bei der Aktivierung ist der Barcode deaktiviert	
	MinCodeLen	0 ... 20	Minimale Anzahl Stellen des Pharmacode	
	MaxCodeLen	0 ... 20	Maximale Anzahl Stellen des Pharmacode	
	MinOkLine	0 ... 100	Minimale Anzahl Zeilen mit identischem Decodier-Ergebnis bei Pharmacode	
	PharmaDir	BOTTOM		Interpretation des Pharmacodes: Bottom/right = von rechts bzw. unten nach links bzw. oben
		TOP		Interpretation des Pharmacodes: Top/left = von links bzw. oben nach rechts bzw. unten
	SearchStart	BOTTOM		Suchrichtung für Pharmacode: Bottom/right = von rechts bzw. unten nach links bzw. oben
		TOP		Suchrichtung für Pharmacode: Top/left = von links bzw. oben nach rechts bzw. unten

Config ID	Parameter ID	Wert	Beschreibung
Rotation Encoder	Resolution	1 ... 9999999 pulses/m	Auflösung des Drehgebers Voreingestellter Wert: 10000 pulses/m
	TrigDistance	0 ... 9999999 mm	Abstand zwischen dem Triggersensor und dem mittigen Lesefeld des Sensors Voreingestellter Wert: 0
	FixPos	0 ... 9999999 mm	Festlegung einer fester Position, bei der die Lesung stattfinden soll
	MaxSpeed	10 ... 10000 mm/s	Maximale Geschwindigkeit während der Lesung Voreingestellter Wert: 3000 mm/s
	TrigPolNeg	1	Auslösen eines Triggersignals bei Low-Pegel aktivieren/deaktivieren
	DelayActive	0	Aktivierung/Deaktivierung eines Ausgabedelays
	ResultDelay	1 ... 2000 mm	Eingabe des Ausgabedelays
	PulseLength	1 ... 1000 mm	Eingabe der Pulslänge
	EncoderPos	0 inc	Aktuelle Drehgeberposition in [inc]
ReadEncoderPos	0/1	Lesen der Drehgeberposition aktivieren/deaktivieren	

Tabelle 7.1 Konfigurationsparameter

## 7.4.2

### Resultübersicht

#### Aufruf der Result-Daten

Result-Daten werden in folgenden Situationen vom Sensor gesendet:

- Nach jeder Übertragung eines Bildes
- Nach Abschluss eines Einlernvorganges
- Nach Anforderung des Sensorstatus
- Nach jeder Inspektion

Die Result-Daten sind im Knoten `CONFIGURATION Id="Result"` enthalten. Abhängig von den Situationen enthält der Knoten zwei verschiedene Result-Parameter.

- Result-Daten (`PF.Foundation.Protocol.XML.ElementResult`)
- Shape-Daten (`PF.Foundation.Protocol.XML.ElementShapeBase`)

Die Shape-Daten werden direkt ins Bild eingezeichnet. Sie bestehen entweder aus geometrischen Formen wie farbigen Vierecken (`ElementShapeRectangle`), oder als Label (`ElementShapeText`). Sie beinhalten Positions- und Größeneigenschaften. Bei den Result-Daten handelt es sich ausschließlich um Text-Daten, die im Barcodemodus und nach dem Einlernen übertragen werden.

### Result-Daten

Die Result-Daten werden mit dem Event `SensorResultDataReceived` übertragen und können mit der Methode `GetResultList` und `GetResultParameter` im Eventhandler abgerufen werden.

Folgende Event und Parameter IDs sind zulässig:

### Result-Daten

Event	Parameter ID	Wert	Beschreibung
Trigger	BCString		Daten des gelesenen Barcodes
	BCType	Code128 Code39 Code13 2/5 Interleaved Pharmacode	Ausgabe der gelesenen Barcode Symbologie
	BCTime	0 ... 200 ms	Decodierzeit des Barcodes in Millisekunden
	BCLQuality	0 ... 100%/0 ... 100%	Mittigkeit der Lage des Barcodes (horizontal / vertikal) 100%/100% -> Der gelesene Barcode liegt exakt in der Mitte.
Teach	Check	"SEARCH_GOOD" "SEARCH_NORMAL" "SEARCH_WEAK"	Vereinfachte Qualitätsausgabe aufgliedert in drei Bereiche
	TeachIndex	1 ... 5	Nummer des Referenzbildes
	TeachPosition		Position des Referenzbildes
	TeachQuality	1 ... 16	Detaillierte Ausgabe der Teach-Qualität. 1 -> schlechte Qualität, 16 -> beste Qualität

Tabelle 7.2 Result-Daten

### Shape-Daten

Die Shape-Daten werden mit den Events `ShapeDataReceived` und `ImageDataReceived` übertragen. Zum Abrufen der Daten wird die Methode `GetShapes` verwendet.

Folgende Parameter werden übertragen:

### Shape-Daten

Type	Parameter ID	Wert	Beschreibung
Element ShapeText	ImageType	ActImage ErrImage TeachImage	Aktuelles Bild Fehlerbild Referenzbild
	Operating Mode	Correlation Greylevel CodeCompare	Bildvergleichsmodus Grauwert-Modus. Vergleich des Bildes anhand von Grauwerten. Barcodevergleichsmodus
	Result	PatternGood PatternBad BarcodeGood BarcodeNoRead	Bildvergleich gut Bild ist nicht gleich dem Referenzbild Barcodevergleich ist gut Barcode konnte nicht gelesen werden
	BCCompare		Daten des Vergleichs-Barcode
	BCString		Daten des gelesenen Barcodes
	BCType	Code128 Code39 Code13 2/5 Interleaved Pharmacode	Ausgabe der gelesenen Barcode Symbologie
	BCTime	0 ... 100 ms	Decodierzeit des Barcodes in Millisekunden
BCLQuality	0 ... 100%/ 0 ... 100	Mittigkeit der Lage des Barcodes (horizontal / vertikal) 100%/100% -> Der gelesene Barcode liegt exakt in der Mitte.	
Element Shape Rectangle	Check		Angabe der Position des Teilbildes des gelesenen Bildes
	Ref		Angabe des Ankerpunktes in dem gelesenen Teilbild welches zum Vergleich mit dem Referenzbild genommen wird

Tabelle 7.3 Shape-Daten

## 8 Software Vision-Configurator

Die Inbetriebnahme und die Bedienung des Sensors erfolgt mit der Software Vision-Configurator.

Der Vision-Configurator ermöglicht Ihnen die komfortable Bedienung des Sensors durch eine übersichtliche Bedienoberfläche. Zu den Standardaufgaben des Vision-Configurators gehören z. B. die Verbindungsherstellung zu dem Sensor, die Parametrierung von Betriebsparametern, die Speicherung von Datensätzen, die Übermittlung und Visualisierung von Daten und die Fehlerdiagnose.



### Netzwerkverbindung herstellen

Um eine Netzwerkverbindung mit dem Sensor herzustellen, gehen Sie wie folgt vor:

1. Versorgen Sie den Sensor mit Spannung.
2. Starten Sie die Software Vision-Configurator.
3. Wählen Sie den angeschlossenen Sensor aus.
4. Überprüfen Sie, ob die richtige IP-Adresse eingegeben ist.
5. Geben Sie den Benutzername und das Passwort ein.

↳ Die Verbindung zum Sensor wird hergestellt.



### **Hinweis!**

#### **Netzwerkconfiguration dokumentieren**

Der Sensor kommuniziert mit der angeschlossenen Maschinensteuerung über das TCP/IP-Protokoll. Um eine korrekte Kommunikation zu gewährleisten, notieren Sie sich unbedingt alle Änderungen, die Sie an der Netzwerkconfiguration vornehmen.

Eine aktuelle Beschreibung der Software Vision-Configurator finden Sie auf <http://www.pepperl-fuchs.com>.

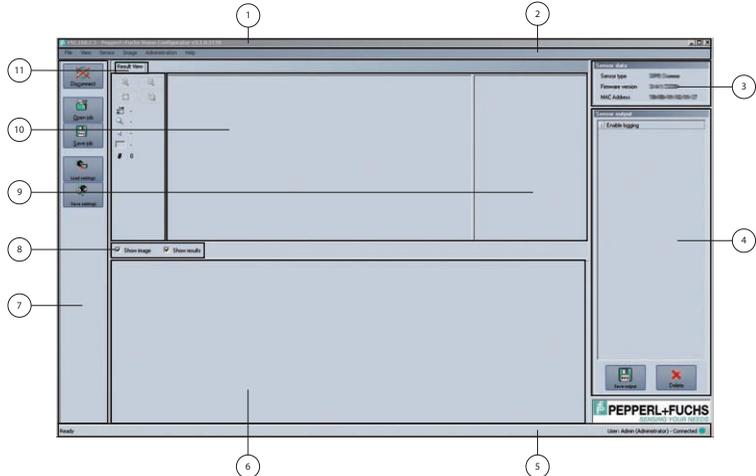
## 8.1 Aufbau des Anwendungsfensters

Nach erfolgreichem Login öffnet sich der Anwendungsbildschirm.



### **Hinweis!**

Die einzelnen Funktionen sind abhängig vom angeschlossenen Sensortyp und aktueller Berechtigungsstufe und sind somit nicht immer alle sichtbar.



Die Software ist analog zu den meisten Windows-Applikationen aufgebaut.

1	Titelleiste	<ul style="list-style-type: none"> <li>■ Zeigt die IP-Adresse, die Softwarebezeichnung und die Versionsnummer an</li> <li>■ enthält die Schaltflächen <b>Minimieren / Maximieren / Schließen</b></li> </ul>
2	Menüleiste	<ul style="list-style-type: none"> <li>■ zeigt alle Menüs des Programms an</li> <li>■ dient als Übersicht und Navigation</li> </ul>
3	Maske <b>Sensor data</b>	<ul style="list-style-type: none"> <li>■ zeigt die Sensordaten des angeschlossenen Sensors an</li> </ul>
4	Maske <b>Sensor output</b>	<ul style="list-style-type: none"> <li>■ zeigt die Loganzeige an</li> </ul>
5	Statusleiste	<ul style="list-style-type: none"> <li>■ zeigt die Statusinformationen zur Anwendung</li> </ul>
6	Parametrierbereich	<ul style="list-style-type: none"> <li>■ enthält die sensorspezifischen Parameter, die Sie einstellen können</li> </ul>
7	Symboleiste	<ul style="list-style-type: none"> <li>■ enthält symbolische Schaltflächen als Erweiterung zum Menü</li> </ul>
8	Kontrollkästchen	<ul style="list-style-type: none"> <li>■ Show images: Aktiviert oder deaktiviert die Bildanzeige</li> <li>■ Show results: Aktiviert oder deaktiviert den Ergebnisbereich</li> </ul>

9	Ergebnisbereich	<ul style="list-style-type: none"> <li>■ zeigt Ergebnisinformationen des Sensors</li> <li>■ Es können abhängig vom angeschlossenen Sensor unterschiedlich viele Registerkarten angezeigt werden.</li> <li>■ Dieses Feld kann mit dem Punkt <b>Show results</b> aktiviert oder deaktiviert werden</li> </ul>
10	Bildanzeige	<ul style="list-style-type: none"> <li>■ zeigt die aufgenommenen oder im Fehlerspeicher liegenden Bilder an</li> <li>■ Dieses Feld kann mit dem Punkt <b>Show images</b> aktiviert oder deaktiviert werden</li> </ul>
11	Registerkarte	<p>Zeigt Informationen über aktuelles Bild und des sich unter der Maus befindlichen Pixels an. So werden folgende Punkte angezeigt:</p> <ul style="list-style-type: none"> <li>■ Bildgröße</li> <li>■ Zoomstufe</li> <li>■ Mausposition in Bildkoordinaten</li> <li>■ aktueller Grauwert</li> <li>■ Bildnummer</li> </ul>

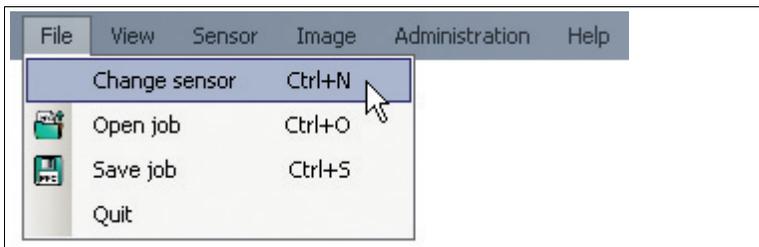
## 8.2 Menüleiste

In der Menüleiste werden verschiedene Menüfunktionen aufgeführt. Die einzelnen Funktionen sind abhängig vom angeschlossenen Sensortyp und aktueller Berechtigungsstufe und sind somit nicht immer alle sichtbar.



Abbildung 8.1 Menüleiste

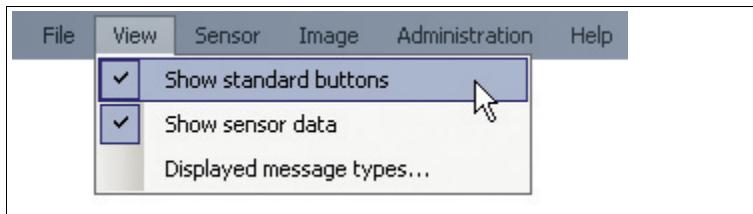
### 8.2.1 Menü File



<b>Change sensor</b>	Trennt die Verbindung zum Sensor und wechselt zurück zum Login-Dialog.
<b>Open job</b>	Lädt eine auf dem PC abgespeicherte Sensorkonfiguration.
<b>Save job</b>	Speichert die aktuelle Sensorkonfiguration auf dem PC.
<b>Quit</b>	Beendet das Programm.

Tabelle 8.1 Menü File

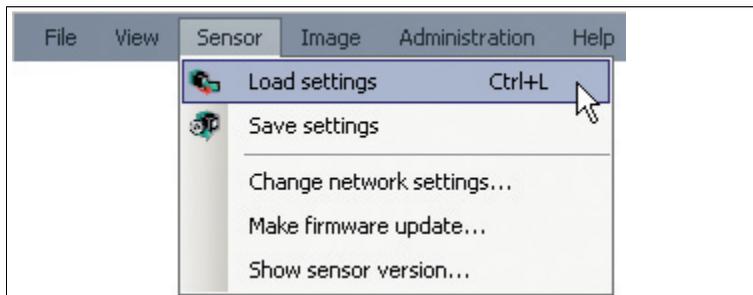
### 8.2.2 Menü View



<b>Show standard buttons</b>	Schaltet die Anzeige der Buttons in der linken Leiste ein und aus.
<b>Show sensor data</b>	Schaltet die Anzeige der Sensordaten rechts oben aus.
<b>Displayed message types...</b>	Hier kann eingestellt werden, welche Meldungstypen vom Sensor ausgegeben werden sollen. Die ausgegebenen Meldungen der gewählten Meldungstypen werden in der rechten Spalte "Sensor output" ausgegeben. Info: Informationen werden angezeigt Warnung: Warnungen werden angezeigt Error: Fehler werden ausgegeben Critical: schwerwiegende Fehler werden ausgegeben Assert: interne Fehler werden angezeigt

Tabelle 8.2 Menü View

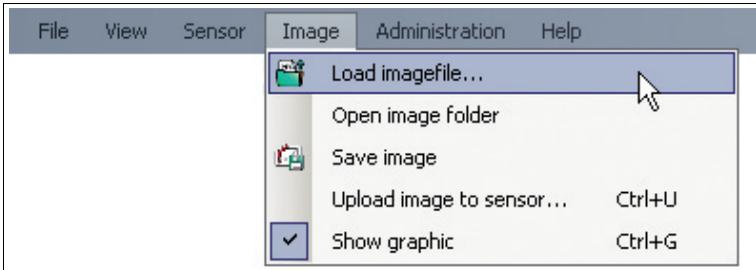
### 8.2.3 Menü Sensor



<b>Load settings...</b>	Lädt die gespeicherten Einstellungen aus dem Sensor
<b>Save settings...</b>	Speichert die Einstellungen in den Sensor
<b>Change network settings...</b>	Falls der Sensor per Ethernet mit dem PC verbunden ist, können hier verschiedene Verbindungseinstellungen auf dem Sensor geändert werden. Nach erfolgter Änderung wird die Verbindung zwischen Vision Configurator und Sensor automatisch getrennt. Ändern Sie gegebenenfalls die Netzwerkadresse auf die neu zugewiesene IP und verbinden Sie den Vision Configurator mit dem Sensor erneut. Nach drücken des Buttons <b>Connect</b> wird der Anmeldebildschirm erneut geöffnet.
<b>Make firmware update...</b>	Führt Firmware Updates durch. Dieser Befehl sollte nur durch erfahrene Anwender benutzt werden.
<b>Show sensor version...</b>	Zeigt die Versionsnummer des Sensor an.

Tabelle 8.3 Menü Sensor

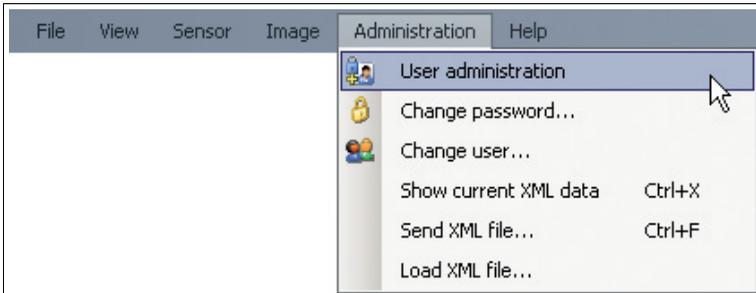
### 8.2.4 Menü *Image*



<b>Load imagefile</b>	Öffnet eine Bilddatei und zeigt das Bild in der Bildanzeige an.
<b>Open image folder</b>	Öffnet den Ordner, in dem aktuell Bilder abgespeichert werden.
<b>Save image</b>	Speichert das aktuell angezeigte Bild auf dem PC ab.
<b>Upload image to sensor</b>	Lädt eine Bilddatei vom PC auf den Sensor.
<b>Show graphic</b>	Schaltet vom Sensor gesendete Anzeigedaten im Bild ein und aus.

Tabelle 8.4 Menü **Image**

### 8.2.5 Menü *Administration*



<b>User administration</b>	Öffnet ein Fenster, in dem alle aktuell angelegten Benutzer gleicher oder niedrigerer Berechtigungsstufe angezeigt werden. Weiterhin können hier neue Benutzer gleicher oder niedrigerer Berechtigungsstufe angelegt oder gelöscht werden. Zusätzlich kann hier das Passwort eines Benutzers auf das Standardpasswort der jeweiligen Benutzerstufe zurückgesetzt werden.
<b>Change password</b>	Ändert das Passwort des aktuellen Benutzers.
<b>Change user</b>	Der Anmeldebildschirm wird geöffnet und es kann ein anderer Benutzer und / oder Sensor ausgewählt werden.
<b>Show current XML data</b>	Lädt die aktuellen XML-Daten aus dem Sensor und zeigt die Daten in einem eigenen Fenster an.
<b>Send XML file...</b>	Speichert die XML-Daten auf einem Rechner.
<b>Load XML file...</b>	Lädt XML-Daten von einem Rechner.

Tabelle 8.5 Menü **Administration**

### 8.2.6 Menü *Help*



<b>Info</b>	Zeigt Informationen über den Vision Configurator an.
-------------	--

Tabelle 8.6 Menü *Help*

### 8.3 Parametrierbereich

In dem Parametrierbereich werden verschiedene Parameterangaben aufgeführt. Die einzelnen Parameter sind abhängig von der aktuellen Berechtigungsstufe und sind somit nicht immer alle sichtbar. Einige Features sind nur in verschiedenen Varianten verfügbar. Abhängig von den eingestellten Parametern werden einige Felder ausgegraut dargestellt.

#### 8.3.1 System Karteikarte

Der gesamte Menüpunkt "Interface" ist nicht bei allen Sensorversionen sichtbar.

#### Parametrierbereich System Menüpunkt Config



#### Network

<b>IP address</b>	Anzeige der IP-Adresse des Sensors
<b>Subnet mask</b>	Anzeige der Subnet-Maske des Sensors
<b>Gateway</b>	Anzeige des Gateways des Sensors

#### Parametrierbereich System Menüpunkt Interfaces

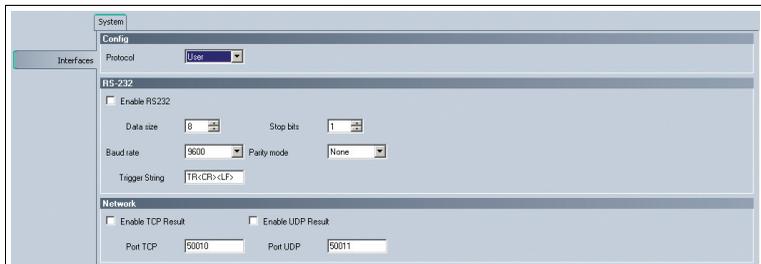


Abbildung 8.2 Parametrierbereich **System Tab - Interfaces**

### Config

<b>Protocol</b>	Datenübertragung User: Daten werden binär übertragen XML: Daten werden als XML-Daten übertragen
-----------------	---

### RS-232

<b>Enable RS232</b>	RS 232 Übertragung aktivieren oder deaktivieren
<b>Data size</b>	Anzahl der Datenbits. Einstellbereich 5 ... 8
<b>Stop bits</b>	Anzahl der Stopbits. Einstellbereich 1 ... 2
<b>Baud rate</b>	Auswahl der Baudrate. Einstellmöglichkeiten: 9600; 57600; 38400; 19200; 115200
<b>Parity mode</b>	Einstellung der Erkennung von Übertragungsfehlern. Einstellmöglichkeiten: Even; None; Odd
<b>Trigger string</b>	Legt eine Zeichenfolge fest. Wird diese Zeichenfolge über die RS 232-Schnittstelle gesendet, wird ein Trigger ausgelöst

### Network

<b>Enable Network Result</b>	Übermittlung der Ergebnisse (Codeinhalte) über die Netzwerkschnittstelle (TCP/IP)
<b>Port</b>	Eingabe des entsprechenden Ports

### Parametrierbereich System Menüpunkt Options

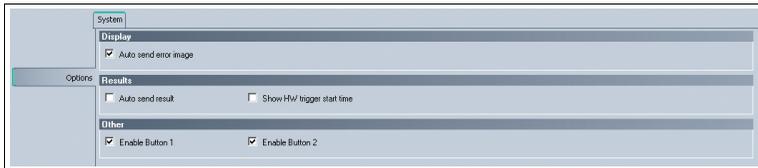


Abbildung 8.3 Parametrierbereich **System Tab - Options**

### Display

<b>Auto send error image</b>	Protokollierung von Fehlern/Fehlerbildern
------------------------------	---

### Results

<b>Auto send result</b>	Übertragung von Ergebnisdaten via Ethernet oder RS 232
<b>Show HW trigger start time</b>	Ausgabe der Trigger Start Zeit in der Ausgabedaten

### Other

<b>Enable Button 1</b>	Aktivierung oder Deaktivierung des Button 1
<b>Enable Button 2</b>	Aktivierung oder Deaktivierung des Button 2

### 8.3.2 Camera Karteikarte

#### Parametrierbereich Camera Menüpunkt Common

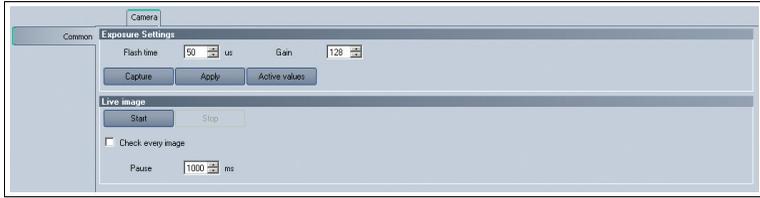


Abbildung 8.4 Parametrierbereich **Camera Tab - Common**

#### Exposure Settings

<b>Flash time</b>	Einstellung der Belichtungszeit in µs
<b>Gain</b>	Einstellung der Verstärkung
Button <b>Capture</b>	Nimmt ein Bild mit eingestellter Flash time auf
Button <b>Apply</b>	Die aktuell eingestellten Werte werden für den nächsten Teachvorgang verwendet
Button <b>Active values</b>	Rücksetzen auf die Werte des letzten Teachvorgangs

#### Live image

<b>Start</b>	Starten des Livebildes
<b>Stop</b>	Stoppen des Livebildes
<b>Check every image</b>	Livebild mit Ergebnisausgabe
<b>Pause</b>	Eingabe der Pause zwischen zwei aufgenommenen Bildern im Livebildmodus in ms

### 8.3.3 Teach Karteikarte

#### Parametrierbereich Teach Menüpunkt Common

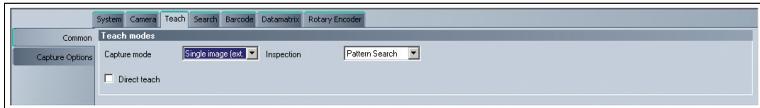


Abbildung 8.5 Parametrierbereich **Teach Tab - Common**

#### Teach modes

<b>Capture mode</b>	Siehe Kapitel 6.4.1
<b>Inspection</b>	Siehe Kapitel 6.4.2
<b>Direct teach</b>	Siehe Kapitel 6.4.2

### Parametrierbereich Teach Tab Menüpunkt Capture Options



Abbildung 8.6 Parametrierbereich Teach Tab - Capture Options

#### Exposure

Einstellungen der Belichtungsregelung	
<b>Flash start value</b>	Minimale Belichtungszeit in µs
<b>Flash step with</b>	Im Einlernprozess wird die Blitzlichtdauer automatisch eingestellt, indem sie Schrittweise erhöht wird Diese Schrittweite kann hier eingestellt werden

### 8.3.4

### Search Karteikarte

#### Parametrierbereich Search Menüpunkt Common



Abbildung 8.7 Parametrierbereich Search Tab - Common

#### Common

<b>Sensitivity</b>	Empfindlichkeitseinstellung Hohe Werte führen zu einer hohen Trefferrate, aber auch zu vermehrten falschen Zurückweisungen Bei Calendar werden selbst kleinste Unterschiede erkannt.
--------------------	--

#### Parametrierbereich Search Menüpunkt Positions

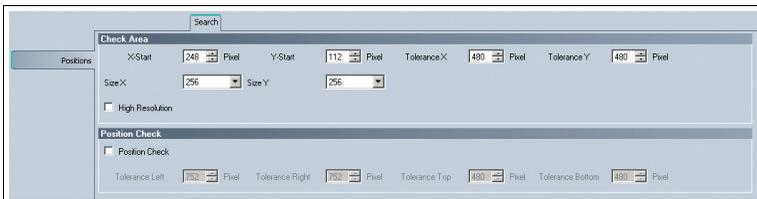


Abbildung 8.8 Parametrierbereich Search Tab - Positions

#### Check Area

<b>X-Start</b>	Eingrenzung des Lesebereiches. Eingabe der X-Startposition
<b>Y-Start</b>	Eingrenzung des Lesebereiches. Eingabe der Y-Startposition
<b>Tolerance X</b>	Eingabe des Toleranzbereiches in X-Richtung
<b>Tolerance Y</b>	Eingabe des Toleranzbereiches in Y-Richtung
<b>Size X</b>	Eingrenzung des Lesebereiches. Größe in X-Richtung
<b>Size Y</b>	Eingrenzung des Lesebereiches. Größe in Y-Richtung
<b>High Resolution</b>	Aktivierung der Suche nach hoher Auflösung

Sie können mit den Einstellungen der Prüf-Position (Check Area) im Bildvergleichs-Modus die Position des zu vergleichenden Bereiches im Musterbild verschieben.



**Hinweis!**

Nach jeder Änderung müssen Sie das Musterbild neu einlernen!

**Position Check**

<b>Position Check</b>	Aktivierung oder Deaktivierung der Überprüfung des Toleranzbereiches
<b>Tolerance Left</b>	Eingrenzung des Lesebereiches. Eingabe der X-Startposition
<b>Tolerance Right</b>	Eingabe des Toleranzbereiches nach rechts
<b>Tolerance Top</b>	Eingabe des Toleranzbereiches nach oben
<b>Tolerance Bottom</b>	Eingabe des Toleranzbereiches nach unten

Sie können zusätzlich zur Prüfung eines eingelernten Musters oder Barcodes eine Positionsprüfung durchführen. Dabei wird geprüft, ob die Position des Musters bzw. des Barcodes im aktuellen Bild innerhalb des angegebenen Toleranzbereiches um die eingelernte Position liegt. Bei aktiviertem Positionsscheck wird der Toleranzbereich im aktuellen Bild und im Musterbild als blauer Rahmen angezeigt. Alle Positionen werden in Pixel geprüft und angegeben.

8.3.5 Barcode Karteikarte

**Parametrierbereich Barcode Menüpunkt Common**



Abbildung 8.9 Parametrierbereich **Barcode Tab - Common**

**Settings**

<b>Compare string</b>	Zeichenkette mit dem der gelesene Barcode verglichen wird. Bei Codegleichheit wird "Gut" ausgegeben
<b>Update</b>	Anforderung des Referenzbarcodes
<b>Timeout</b>	Maximale Lesedauer, bei Überschreitung der Zeit wird ein "No-Read" ausgegeben

### Parametrierbereich Barcode Menüpunkt Codes

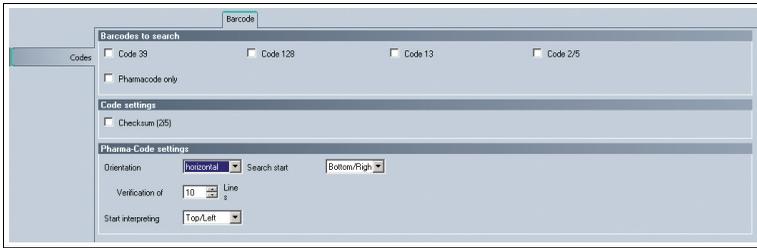


Abbildung 8.10 Parametrierbereich **Barcode Tab - Codes**

#### Barcodes to search

<b>Code 39</b>	Auswahl, ob der Code 39 gelesen werden soll
<b>Code 128</b>	Auswahl, ob der Code 128 gelesen werden soll
<b>Code 13</b>	Auswahl, ob der Code 13 gelesen werden soll
<b>Code 2/5</b>	Auswahl, ob der Code 2/5 gelesen werden soll
<b>Pharmacode only</b>	Auswahl, ob nur der Pharmacode eingelesen werden soll

#### Code settings

<b>Checksum (2i5)</b>	Checksumme bei Lesung des Code Interleave 2/5 aktivieren
-----------------------	--

#### Pharmacode settings

<b>Orientation</b>	Auswahl zwischen Leiter-Barcode(vertical) oder Gartenzaun-Barcode (horizontal)
<b>Search start</b>	Welche Bildecke ist dem zu lesendem Barcode am nächsten
<b>Verification of</b>	Die zu erwartende Anzahl der Pharmacode-Balken
<b>Start interpreting</b>	Leserichtung des Barcodes

### Parametrierbereich Barcode Menüpunkt Region of interest

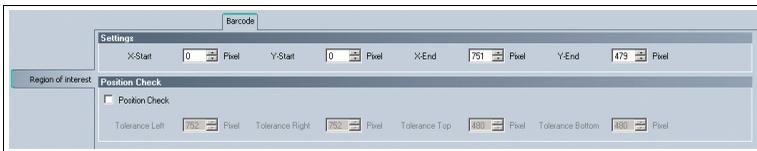


Abbildung 8.11 Parametrierbereich **Barcode Tab - Region of interest**

#### Settings

<b>X-Start</b>	Eingrenzung des Lesebereiches. Eingabe der X-Startposition
<b>Y-Start</b>	Eingrenzung des Lesebereiches. Eingabe der Y-Startposition
<b>X-End</b>	Eingrenzung des Lesebereiches. Eingabe der X-Endposition
<b>Y-End</b>	Eingrenzung des Lesebereiches. Eingabe der Y-Endposition

Sie können mit den Einstellungen der Lesebereiches (Settings) die Position des zu lesenden Bereiches, in dem sich der Barcode befinden soll, verschieben.



**Hinweis!**

Nach jeder Änderung müssen Sie den Barcode neu einlernen!

**Position Check**

<b>Position Check</b>	Aktivierung oder Deaktivierung der Überprüfung des Toleranzbereiches
<b>Tolerance Left</b>	Eingrenzung des Lesebereiches. Eingabe der X-Startposition
<b>Tolerance Right</b>	Eingabe des Toleranzbereiches nach rechts
<b>Tolerance Top</b>	Eingabe des Toleranzbereiches nach oben
<b>Tolerance Bottom</b>	Eingabe des Toleranzbereiches nach unten

Sie können zusätzlich zur Prüfung eines eingelernten Musters oder Barcodes eine Positionsprüfung (Position Check) durchführen. Dabei wird geprüft, ob die Position des Musters bzw. des Barcodes im aktuellen Bild innerhalb des angegebenen Toleranzbereiches um die eingelernte Position liegt. Bei aktiviertem Positionskontrolle wird der Toleranzbereich im aktuellen Bild und im Musterbild als blauer Rahmen angezeigt. Alle Positionen werden in Pixel geprüft und angegeben.

**Parametrierbereich Barcode Menüpunkt Output Strings**



Abbildung 8.12 Parametrierbereich **Barcode Tab - Output Strings**

**Output Strings**

<b>Format string good</b>	Ausgabestring bei Gut-Lesung PSTR: Ausgabe des gelesenen Wertes PSTR(0,5): Ausgabe des gelesenen Wertes ab Position 0, 5 Zeichen lang PSTR(3,3): Ausgabe des gelesenen Wertes ab Position 3, 3 Zeichen lang
<b>Format string bad</b>	Ausgabestring bei Schlecht-Lesung

### 8.3.6

## Data Matrix Karteikarte

Die Karteikarte Data Matrix ist optional und ist nicht bei allen Versionen verfügbar.

### Parametrierbereich Datamatrix Menüpunkt Output Strings



Abbildung 8.13 Parametrierbereich **Datamatrix Tab - Output Strings**

### Output Strings

<b>Format string good</b>	Ausgabestring bei Gut-Lesung PSTR: Ausgabe des gelesenen Wertes PSTR(0,5): Ausgabe des gelesenen Wertes ab Position 0, 5 Zeichen lang PSTR(3,3): Ausgabe des gelesenen Wertes ab Position 3, 3 Zeichen lang
<b>Format string bad</b>	Ausgabestring bei Schlecht-Lesung

### Parametrierbereich Datamatrix Menüpunkt Options



Abbildung 8.14 Parametrierbereich **Datamatrix Tab - Options**

### Options

<b>Timeout</b>	Maximale Lesedauer, bei Überschreitung der Zeit wird ein "No-Read" ausgegeben
----------------	---

### Parametrierbereich Datamatrix Menüpunkt Search



Abbildung 8.15 Parametrierbereich **Datamatrix Tab - Search**

### Search Window

<b>X start</b>	X-Startposition des Lesebereiches
<b>Y start</b>	Y-Startposition des Lesebereiches
<b>Width</b>	Breite des Lesebereiches
<b>Height</b>	Höhe des Lesebereiches

Sie können mit den Einstellungen der Lesebereiches (Search Window) die Position des zu lesenden Bereiches, in dem sich der Data-Matrix-Code befinden soll, verschieben.



### Hinweis!

Nach jeder Änderung müssen Sie den Data-Matrix-Code neu einlernen!

### 8.3.7 Rotary Encoder Karteikarte

In dieser Karteikarte haben Sie die Möglichkeit die Einstellungen des Drehbereingangs zu verändern.

#### Parametrierbereich Rotary Encoder Menüpunkt Common

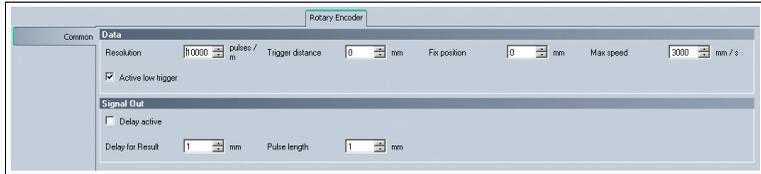


Abbildung 8.16 Parametrierbereich **Rotary Encoder Tab - Common**

#### Data

<b>Resolution</b>	Auflösung des Drehgebers in Impulse pro Meter
<b>Trigger distance</b>	Abstand zwischen Triggersensor/Bogenkante und Mitte des Sensor-Lesefeldes.
<b>Fix position</b>	manuelle Eingabe der Bildaufnahme position ab der Bogenkante
<b>Max speed</b>	Maximale Geschwindigkeit des Förderers
<b>Active low trigger</b>	Triggern bei fallender Flanke aktivieren oder deaktivieren

#### Signal Out

<b>Delay active</b>	Aktivierung der Einstellmöglichkeit der Wartezeit
<b>Delay for result</b>	Wartezeit in mm zwischen Trigger und Ergebnisausgabe
<b>Pulse length</b>	Länge des Ausgangsimpulses der Ergebnisausgabe in mm

#### Parametrierbereich Rotary Encoder Menüpunkt Test

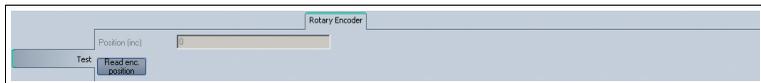


Abbildung 8.17 Parametrierbereich **Rotary Encoder Tab - Test**

#### Test

<b>Position (inc)</b>	Anzeige der aktuelle Encoder Position
<b>Read enc. position</b>	Encoderposition auslesen

## 8.4

### Symboleiste

In der Symboleiste werden verschiedene Symbolfunktionen aufgeführt.

 Disconnect	Die Verbindung zwischen dem PC und dem Sensor wird getrennt
 Open job	Öffnet eine auf der Festplatte abgespeicherte Einstellung
 Save job	Speichert die vorgenommenen Einstellungen auf der Festplatte
 Load settings	Verwirft alle Änderungen und lädt die gespeicherten Parameter vom Sensor
 Save settings	Speichert alle Daten permanent auf dem Sensor ab
Trigger start	Löst einen Triggerung des Sensors aus. Je nach Sensor-Betriebsart liefert er einen einzelnen Messwert oder kontinuierlich Messwerte zurück
Teach start	Startet den Einlernmodus, um ein Referenzbild zu lernen. Wartet danach auf Trigger Signale
Get last Image	Anzeige des zuletzt aufgenommenes Bildes. Bei Multitriggeraufnahmen können durch mehrmaliges Drücken sämtliche Bilder durchlaufen werden
Get pattern Image	Anzeige des eingelernten Referenzbildes
Get teach Image	Anzeige aller im Teachprozess aufgenommenen Bilder
Get error image	Anzeige des letzten Fehlerbildes
Get next error image	Anzeige der vorherigen Fehlerbilder

## 8.5 Maske Sensor Data

In dem Feld Sensor Data wird der angeschlossene Sensortyp, die Firmware Version des angeschlossenen Sensors und die MAC-Adresse angezeigt.

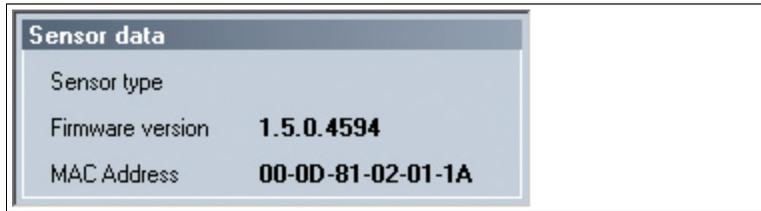


Abbildung 8.18 Maske Sensor Data

## 8.6 Maske Sensor Output

In dem Feld Sensor output wird die Kommunikation zwischen Host-Rechner und angeschlossenem Sensor angezeigt. Unter dem Menüpunkt **View / Displayed message types...** können die anzuzeigenden Nachrichten ausgewählt werden .



Abbildung 8.19 Maske Sensor output

Im unteren Bereiches des Fensters Sensor output befinden sich zwei Buttons.



<b>Save output</b>	Speichert den gesamten Fensterinhalt in einer txt-Datei
<b>Delete</b>	Löscht den gesamten Inhalt des Fensters

## 8.7

### Bildanzeige

In der Bildanzeige können verschiedene Bilder von Referenzbilder über Fehlerbilder zum aktuellen Bild angezeigt werden. Über die beiden Kontrollkästchen **Show image** und **Show results** können Sie die Bildanzeige und Ergebnisanzeige ein - und ausschalten.

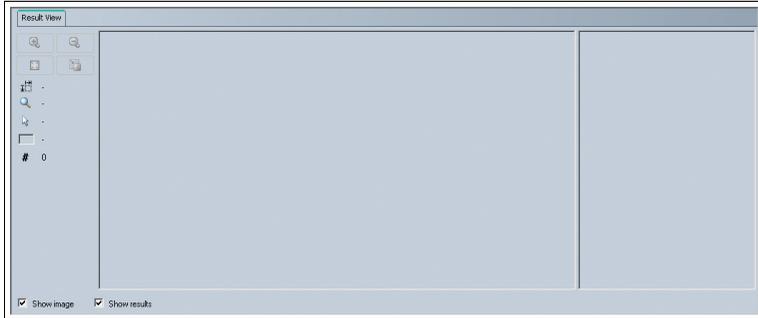


Abbildung 8.20 Bildanzeige Result View

Durch Betätigen der rechten Maustaste oder der Kontextmenü-Taste erscheint folgendes Kontextmenü:



Abbildung 8.21 Bildanzeige Image View Kontextmenü

#### Kontextmenü

<b>Load image file...</b>	Lädt ein Sensorbild. Das Sensorbild kann ausgewählt werden.
<b>Open image folder</b>	Öffnet den Speicherort
<b>Save image</b>	Speichert das angezeigte Sensorbild

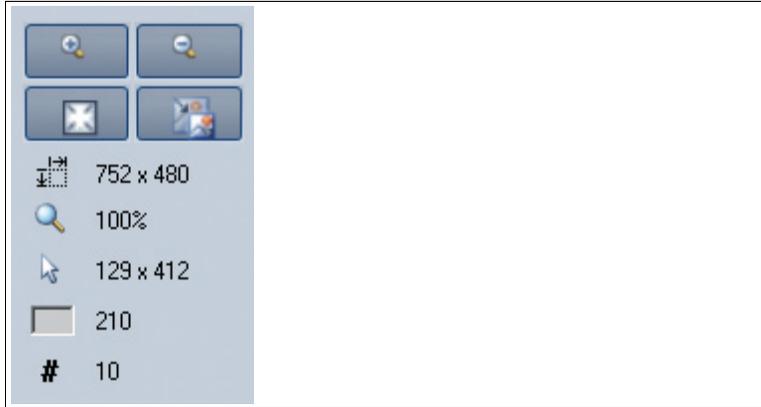


Abbildung 8.22 Bildanzeige Image View Werkzeugleiste

<b>Lupe +</b>	Sensorbild hinein zoomen
<b>Lupe -</b>	Sensorbild hinaus zoomen
<b>Fenster anpassen</b>	Sensorbild dem Fenster anpassen
<b>Originalgröße</b>	Originalbildgröße einstellen
<b>Größenangabe</b>	Angabe der Größe des Sensorbildes
<b>Zoomfaktor</b>	Anzeige des Zoomfaktors. Zoomfaktor 1 ist Originalgröße.
<b>Positionsangabe</b>	Angabe der Position des Mousezeigers
<b>Grauwertangabe</b>	Grauwertangabe des Pixels auf dem der Mousezeiger steht
<b>Bildnummer</b>	Angabe der Bildnummer

Tabelle 8.7 Menü **Image View Werkzeugleiste**

## 9 Wartung und Reparatur

### 9.1 Wartung

Das Kabel und das Netzteil sind wartungsfrei. Um die bestmögliche Geräteleistung zu erzielen, halten Sie die Optikeinheit des Gerätes sauber und reinigen Sie diese bei Bedarf.

Beachten sie bei der Reinigung folgende Hinweise:

- Berühren Sie die Optikeinheit nicht mit den Fingern.
- Tauchen Sie das Gerät nicht in Wasser ein. Besprühen Sie es nicht mit Wasser oder anderen Flüssigkeiten.
- Verwenden Sie zum Reinigen der Geräteoberfläche keine Scheuermittel.
- Benutzen Sie ein Baumwoll- oder Zellstofftuch, das mit Wasser oder Isopropylalkohol angefeuchtet (nicht getränkt) ist.
- Entfernen Sie Alkoholrückstände mit einem Baumwoll- oder Zellstofftuch, das mit destilliertem Wasser angefeuchtet (nicht getränkt) ist.
- Wischen Sie die Geräteoberflächen mit einem fusselfreien Tuch trocken.

### 9.2 Reparatur

Die Geräte dürfen nicht repariert, verändert oder manipuliert werden. Im Falle eines Defektes ist das Produkt immer durch ein Originalgerät zu ersetzen.

# 10 Störungsbeseitigung

## 10.1 Was tun im Fehlerfall

Bevor Sie einen Serviceeinsatz beauftragen, prüfen Sie bitte, ob folgende Maßnahmen erfolgt sind:

- Testen der Anlage durch den Kunden gemäß den folgenden Checklisten,
- Telefonische Beratung durch den Service-Center zur Eingrenzung des Problems.

### Checkliste

Fehler	Ursache	Behebung
LED "PWR" leuchtet nicht	Die Spannungsversorgung ist abgeschaltet.	Ermitteln Sie, ob es einen Grund für die Abschaltung gibt (Installationsarbeiten, Wartungsarbeiten ...). Schalten Sie ggf. die Spannungsversorgung ein.
LED "PWR" leuchtet nicht	Verdrahtungsfehler im Verteiler oder Schaltschrank.	Überprüfen Sie sorgfältig die Verdrahtung und beheben Sie ggf. vorhandene Verdrahtungsfehler.
keine Verbindung zum Gerät	Netzwerkkabel nicht verbunden.	Schließen Sie das Netzwerkkabel an.
keine Verbindung zum Gerät	Falsches Netzwerkkabel verwendet.	Direktverbindung zwischen PC und Gerät: Verwenden Sie ein Crossover-Netzwerkkabel. Verbindung über ein bestehendes Netzwerk: Verwenden Sie ein Twisted-Pair-Netzwerkkabel
Ausgang READY geht nicht auf High-Pegel	Hardware-Fehler	Überprüfen Sie den Drehgeber.

- Falls keiner der vorherigen Punkte zum Ziel geführt hat, nehmen Sie bitte Kontakt zum Service-Center auf. Halten Sie hier bitte die Fehlerbilder und die Versionsnummer des Sensors bereit. Die Versionsnummer finden Sie auf der Bedienoberfläche unten links.

# 11

## Anhang

### 11.1

#### Technische Daten

##### Allgemeine Daten

Lichtart	Integrierter LED-Blitz (weiß)
Symbolgien	2/5 interleaved, Code13, Code39, Code128, Pharmacode
Objektgröße	25 mm x 25 mm
Leseabstand	55 mm
Schärfentiefe	± 5 mm
Lesefeld	65 mm x 40 mm
Auswertefrequenz	10 Hz
Objektgeschwindigkeit	getriggert max. 4 m/s

##### Kenndaten

Bildaufnehmer	
Typ	CMOS , Global Shutter
Pixelanzahl	752 x 480 Pixel
Graustufen	256
Bildaufnahme	verzögerungsfrei , programmgesteuert oder extern getriggert

##### Anzeigen/Bedienelemente

Betriebsanzeige	LED grün: Betriebsbereit
Bedienelemente	2 x Taster
LED-Anzeige	Trigger, Good, Bad, Teach, Diag1, Diag2, PWR

##### Elektrische Daten

Betriebsspannung	24 V DC ± 15% , PELV
Leerlaufstrom	max. 250 mA
Leistungsaufnahme	6 W

##### Schnittstelle 1

Schnittstellentyp	Ethernet
Protokoll	TCP/IP
Übertragungsrate	100 MBit/s
Kabellänge	max. 30 m

## Eingang

Eingangsspannung	extern anzulegen 24 V $\pm$ 15% PELV
Anzahl/Typ	1 Triggereingang 1 Teacheingang 2 Eingänge (IN1, IN2)
Eingangsstrom	ca. 2 mA bei 24 V DC
Schaltswelle	low: < 10 V, high: > 15 V
Kabellänge	max. 30 m

## Ausgang

Anzahl/Typ	Teach aktiv, Gut, Schlecht, Ready
Schaltungsart	PNP , kurzschluss-/überlastfest
Schaltspannung	extern anzulegen 24 V $\pm$ 15 % PELV
Schaltstrom	max. 100 mA je Ausgang
Kabellänge	max. 30 m

## Umgebungsbedingungen

Umgebungstemperatur	0 ... 45 °C (32 ... 113 °F)
Lagertemperatur	-20 ... 60 °C (-4 ... 140 °F)
Relative Luftfeuchtigkeit	80 % , nicht kondensierend
Schockfestigkeit	< 50 g
Vibrationsfestigkeit	< 3 G , 11 ... 200 Hz

## Mechanische Daten

Schutzart	IP65
Anschluss	M12x1 Stecker, 8-polig, Standard (Versorgung+IO) , M12x1 Buchse, 5-polig, Standard (IO) , M12x1 Buchse, 4-polig, D-codiert (LAN)
Material	
Gehäuse	PC/ABS
Lichtaustritt	Kunststoffscheibe
Montage	4 x Gewinde M6
Masse	ca. 160 g

## Normen- und Richtlinienkonformität

Normenkonformität	
Störfestigkeit	EN 61326-1
Störaussendung	EN 61000-6-4
Schutzart	EN 60529
Laserklasse	IEC 60825-1:2007

# FABRIKAUTOMATION – SENSING YOUR NEEDS



## Zentrale weltweit

Pepperl+Fuchs GmbH  
68307 Mannheim · Deutschland  
Tel. +49 621 776-0  
E-Mail: [info@de.pepperl-fuchs.com](mailto:info@de.pepperl-fuchs.com)

## Zentrale USA

Pepperl+Fuchs Inc.  
Twinsburg, Ohio 44087 · USA  
Tel. +1 330 4253555  
E-Mail: [sales@us.pepperl-fuchs.com](mailto:sales@us.pepperl-fuchs.com)

## Zentrale Asien

Pepperl+Fuchs Pte Ltd.  
Singapur 139942  
Tel. +65 67799091  
E-Mail: [sales@sg.pepperl-fuchs.com](mailto:sales@sg.pepperl-fuchs.com)

[www.pepperl-fuchs.com](http://www.pepperl-fuchs.com)

 **PEPPERL+FUCHS**  
SENSING YOUR NEEDS

Änderungen vorbehalten  
Copyright PEPPERL+FUCHS • Printed in Germany

TDOCT2119D\_\_GER  
10/2012