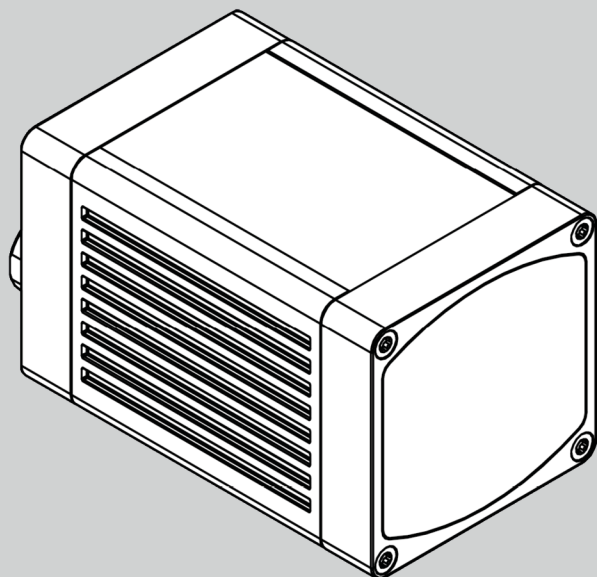


**SBSA/SBSX**

**Vision-Sensor**

**FESTO**

Kommunikationshandbuch  
Software-Version 2.4



## Copyright (Deutsch)

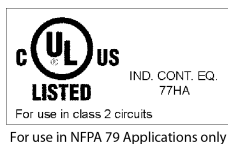
Die Wiedergabe bzw. der Nachdruck dieses Dokuments, sowie die entsprechende Speicherung in Datenbanken und Abrufsystemen bzw. die Veröffentlichung, in jeglicher Form, auch auszugsweise, oder die Nachahmung der Abbildungen, Zeichnungen und Gestaltung ist nur auf Grundlage einer vorherigen, in schriftlicher Form vorliegenden Genehmigung seitens Festo Corporation, zulässig.

Für Druckfehler und Irrtümer, die bei der Erstellung des Dokumentes unterlaufen sind, ist jede Haftung ausgeschlossen. Liefermöglichkeiten und technische Änderungen vorbehalten.

- Originalbetriebsanleitung -

Erstveröffentlichung 09/2021

Festo Corporation  
1377 Motor Parkway Suite 310  
Islandia, NY 11749  
United States



# Inhaltsverzeichnis

<b>1 Informationen zum Dokument</b> .....	<b>5</b>
1.1 Symbolerklärung .....	5
1.2 Weitere Dokumente .....	6
1.3 Dokumentversion .....	6
<b>2 Netzwerkanschluss</b> .....	<b>7</b>
2.1 Einbindung des SBS ins Netzwerk / Gateway .....	7
2.2 Netzwerkanschluss: Direkter Anschluss .....	8
2.3 Netzwerkanschluss: Anschluss über Netzwerk .....	9
2.4 Verwendete Ethernet-Ports .....	9
2.5 Zugriff auf SBS über Netzwerk .....	11
2.6 Zugriff auf SBS über das Internet / World Wide Web .....	12
<b>3 Konfiguration SBS Vision-Sensor</b> .....	<b>13</b>
<b>4 Ethernet TCP / IP, Port 2005 / 2006</b> .....	<b>17</b>
4.1 Beispiel: Datenausgabe vom SBS an PC / Steuerung .....	17
4.2 Beispiel: Kommandos (Requests) von PC / Steuerung an SBS .....	21
4.3 Beispiel: Jobumschaltung von PC / Steuerung an SBS .....	22
<b>5 Service / Visualisierung</b> .....	<b>27</b>
5.1 Backuperstellung .....	27
5.2 Visualisierung .....	27
<b>6 Telegramme SBS für PROFINET und EtherNet/IP</b> .....	<b>29</b>
6.1 Modul 1: „Control“ (Von Steuerung an SBS) .....	29
6.2 Modul 2: „Status“ (Von SBS an Steuerung) .....	31
6.3 Modul 3: „Data“ (Von SBS an Steuerung) .....	34
6.4 Modul 4: „Request“ (Von Steuerung an SBS) .....	35
6.5 Modul 5: „Response“ (Von Steuerung an SBS) .....	36
6.6 Start- / Ende-Kriterien je Telegramm .....	37
<b>7 Timing-Diagramme zur SBS Kommunikation</b> .....	<b>39</b>
<b>8 Request-Sequenzen</b> .....	<b>43</b>
8.1 Trigger Request Reihenfolge .....	44
8.2 Change Job Request Reihenfolge .....	45
8.3 Switch to Run Reihenfolge .....	46
8.4 Reihenfolge für Requests via Request / Response Modul .....	47
<b>9 PROFINET</b> .....	<b>49</b>
9.1 Konfigurationsbeispiel Siemens S7-1200 TIA 12 .....	49

9.1.1 Neues Projekt anlegen .....	49
9.1.2 GSD-Datei auswählen .....	49
9.1.3 SBS Vision-Sensor zum Projekt hinzufügen .....	50
9.1.4 Name in SBS schreiben .....	55
9.1.5 Projekt auf Steuerung laden .....	56
9.1.6 Zuordnung der Ausgangsdaten .....	56
<b>10 EtherNet/IP .....</b>	<b>61</b>
10.1 Konfigurationsbeispiel Rockwell CompactLogix™ .....	61
10.2 EDS-Datei installieren .....	63
10.3 Modul anlegen .....	69
10.3.1 Auswahl über Hardware-Katalog (mit EDS-Datei) .....	69
10.3.2 Verwendung eines Generic Device (ohne EDS-Datei) .....	73
10.4 Projekt in Steuerung laden .....	77
10.5 Zuordnung der Ausgangsdaten .....	79
<b>11 Telegramme und Datenausgabe .....</b>	<b>81</b>
11.1 Übersicht Telegramme .....	81
11.2 Telegramme: Verfügbarkeit und unterstützte Schnittstellen .....	86
11.3 Fehlercodes .....	89
11.4 Beschreibung Telegramme ASCII .....	92
11.4.1 Allgemein .....	92
11.4.2 Kontrolle .....	95
11.4.3 Jobeinstellungen .....	103
11.4.4 Kalibrierung .....	140
11.4.5 Visualisierung .....	163
11.4.6 Service (nur auf Port 1998 und in ASCII Format verfügbar) .....	165
11.4.7 Datenausgabe ASCII .....	175
11.5 Beschreibung Telegramme BINÄR .....	192
11.5.1 Allgemein .....	192
11.5.2 Kontrolle .....	195
11.5.3 Jobeinstellungen .....	204
11.5.4 Kalibrierung .....	239
11.5.5 Visualisierung .....	260
11.5.6 Datenausgabe BINÄR .....	262

# 1 Informationen zum Dokument

## 1.1 Symbolerklärung

### Warnhinweise



**VORSICHT / WARNUNG / GEFAHR**

Dieses Symbol weist auf möglicherweise gefährliche Situationen hin, die zum Tod oder zu schweren Verletzungen führen können, wenn sie nicht gemieden werden.



**WARNUNG**

Dieses Symbol weist auf möglicherweise gefährliche Situationen durch Laserstrahlen hin.



**ACHTUNG:**

Dieses Symbol kennzeichnet Textstellen, die unbedingt zu beachten sind. Die Nichtbeachtung kann zu Personen- oder Sachschäden führen.



**HINWEIS:**

Dieses Symbol hebt nützliche Tipps und Empfehlungen sowie Informationen für einen effizienten Betrieb hervor.

### Detektoren



Mustervergleich



Kontur



Kontrast



Helligkeit



Graustufe



Messschieber



BLOB



Kontur 3D



Barcode



Datacode



OCR



Farbwert



Farbliste



Farbfläche



Ergebnisverarbeitung



Zielmarke 3D

### Lagenachführung



Lagenachführung

Beinhaltet die Lage-Detektoren: Konturvergleich, Mustervergleich und Kantenantastung

## 1.2 Weitere Dokumente

Für den SBS Vision-Sensor stehen folgende Dokumente im Download-Bereich der Festo-Homepage zur Verfügung.

- SBS Kommunikationshandbuch
- SBS Benutzerhandbuch
- SBS Betriebsanleitung

Des Weiteren sind diese Dokumente Teil der Software-Installation und im Unterordner "...\\Documentation\\" sowie über das Windows-Startmenü zu finden.

## 1.3 Dokumentversion

Dieses Handbuch beschreibt die SBS Software -Version 2.4.

## 2 Netzwerkanschluss

### 2.1 Einbindung des SBS ins Netzwerk / Gateway

In Vision Sensor Device Manager/Aktive Sensoren werden alle SBS Vision-Sensoren als Liste angezeigt, die sich im gleichen Netzwerksegment wie der PC befinden, auf dem Vision Sensor Device Manager läuft. Zur Aktualisierung der Liste den Button „Finden“ drücken, für z.B. Sensoren, die erst nach Aufruf von Vision Sensor Device Manager eingeschaltet wurden.

Für Sensoren die zwar im Netzwerk angeschlossen sind, sich jedoch über ein Gateway in einem anderen Netzwerksegment befinden, bitte unter „Aktiven Sensor hinzufügen“ die entsprechende IP-Adresse des Sensors eingeben und den Button „Hinzufügen“ drücken. Der entsprechende Sensor erscheint nun ebenfalls in der Liste „Aktive Sensoren“ und kann angesprochen und bearbeitet werden.

## 2.2 Netzwerkanschluss: Direkter Anschluss

Herstellen einer direkten Ethernet-Verbindung zwischen SBS Vision-Sensor und PC

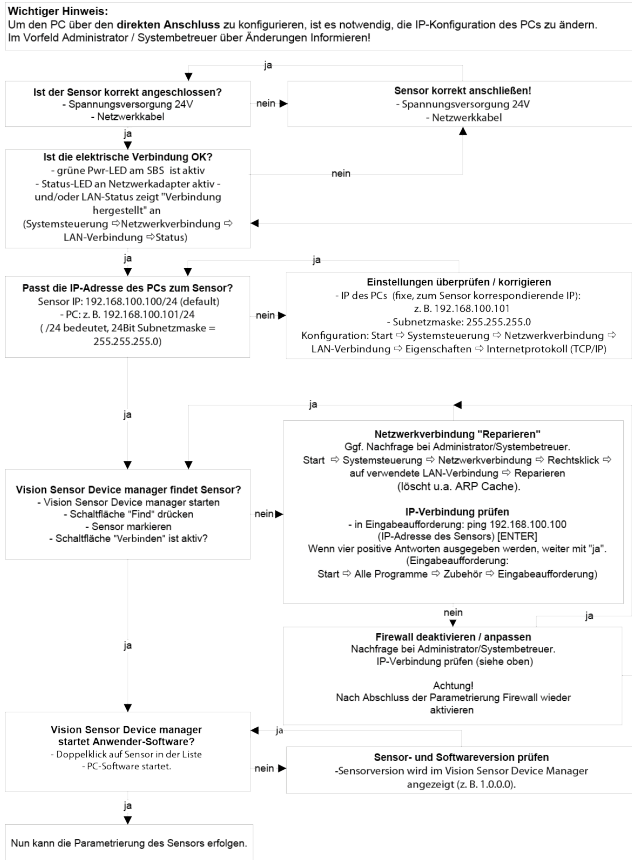


Abb. 1: Direkter Anschluss Sensor / PC, Ablauf und Problembehebung



### 2.3 Netzwerkanschluss: Anschluss über Netzwerk

Herstellen einer Ethernet-Verbindung zwischen SBS Vision-Sensor und PC über ein Netzwerk.

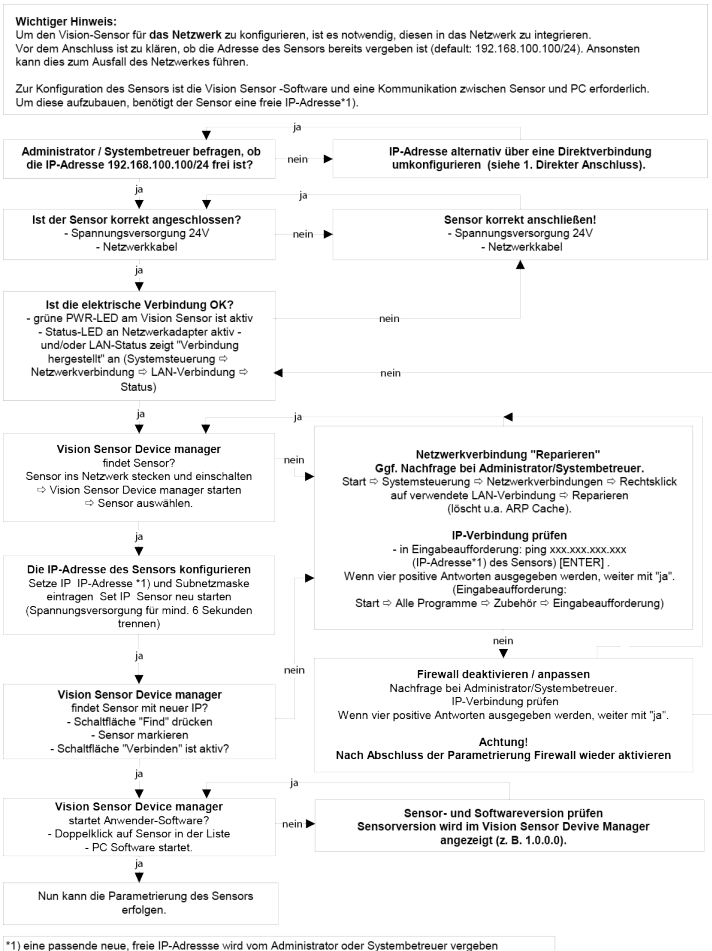


Abb. 2: Anschluss über ein Netzwerk Sensor / PC, Ablauf und Problembhebung

### 2.4 Verwendete Ethernet-Ports

Wenn der SBS Vision-Sensor in einem Netzwerk eingebunden werden soll, müssen die folgenden Ports ggf. durch einen Administrator entsprechend freigegeben werden. Dies ist nur dann der Fall,

wenn diese Ports im Firmennetzwerk bzw. durch eine auf dem PC installierte Firewall zuvor explizit gesperrt wurden.

Für die Kommunikation zwischen SBS Software (PC) und SBS Vision-Sensor werden folgende Ports verwendet:

- Port 2000, TCP
- Port 2001, UDP Broadcast (Zum Finden der Sensoren durch Vision Sensor Device Manager)
- Port 2002, TCP
- Port 2003, TCP
- Port 2004, TCP

Für die Kommunikation zwischen Steuerung (SPS oder Steuerungs-PC) und SBS Vision-Sensor werden folgende Ports verwendet.

Prozess Schnittstellen:

- Ethernet
  - Port 2005, TCP (Implizite Ergebnisse, d.h. vom Anwender konfigurierte Ergebnisdaten)
  - Port 2006, TCP (Explizite Anfragen, z.B. Trigger oder Jobwechsel)
- EtherNet/IP:
  - Port 2222, UDP
  - Port 44818, TCP
- PROFINET:
  - Port 161, UDP
  - Port 34962, UDP
  - Port 34963, UDP
  - Port 34964, UDP
- Service:
  - Port 22, TCP
  - Port 1998, TCP
- SBSxWebView:
  - Port 80



**HINWEIS:**

Wenn die Ports 2005 bzw. 2006 in der Konfigurationssoftware geändert werden, müssen diese auch bei der Firewall durch einen Administrator entsprechend geändert werden.

## 2.5 Zugriff auf SBS über Netzwerk

Beispielhafte Werte für IP etc.

Zugriff auf SBS 1 von PC 1 aus, wenn in gleichem Subnetz

- Über Vision Sensor Device Manager (/Finden)

Zugriff auf SBS 2 von PC1 aus, wenn in anderem Subnetz

nur wenn:

- Gateway in Sensor 2 korrekt gesetzt (hier auf 192.168.30.1) - und
- in Vision Sensor Device Manager über IP- hinzufügen die IP des Sensor 2 richtig eingegeben wurde  
> danach erscheint auch SBS 2 in Liste „Aktive Sensoren“ in Vision Sensor Device Manager!

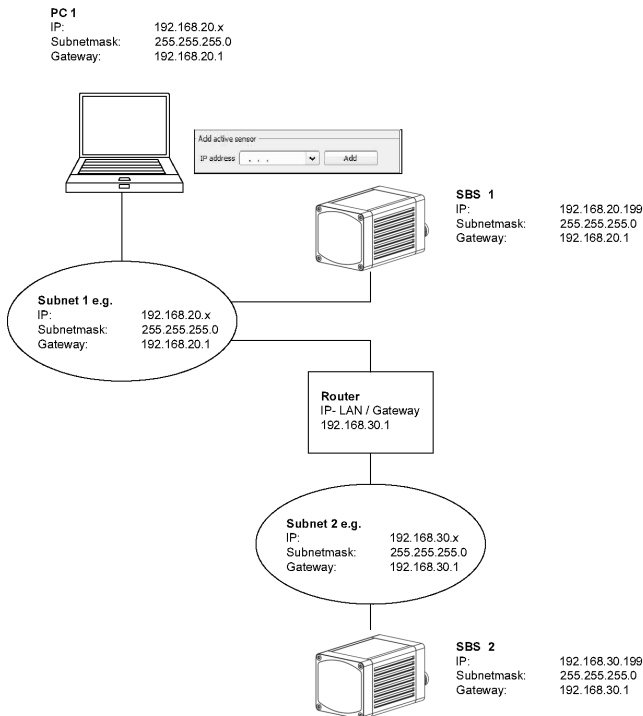


Abb. 3: Zugriff auf SBS über Netzwerk, gleiches oder anderes Subnetz

## 2.6 Zugriff auf SBS über das Internet / World Wide Web

Beispielhafte Werte für IP etc.

Zugang von PC 1 (Firmennetz 1), über das Word Wide Web, in Firmennetz 2 zu SBS 1.

1. Auf PC1 (Firmennetz 1) in Vision Sensor Device Manager die IP- WAN des Router 2 (Firmennetz 2) unter „Aktiven Sensor hinzufügen“ eintragen und hinzufügen (hier im Beispiel: 62.75.148.101)
2. In Router 2 die Ports die vom Sensor genutzt werden sollen im Router freigeben (S. Kapitel: [Verwendete Ethernet-Ports](#)).

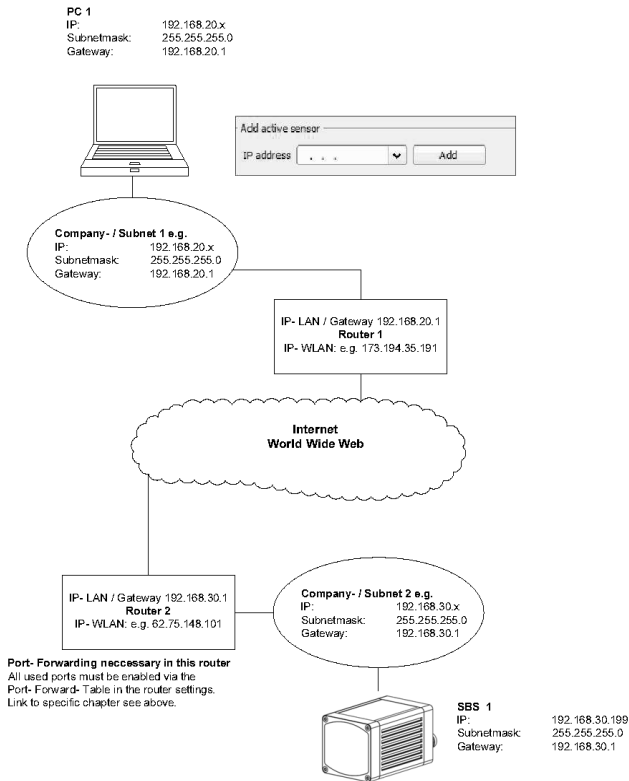
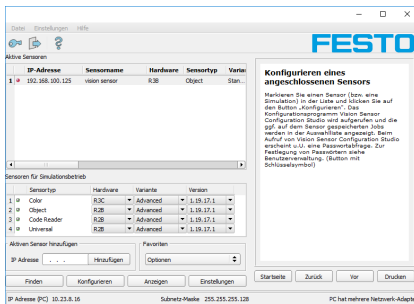


Abb. 4: Zugriff auf SBS über das Internet / World Wide Web

### 3 Konfiguration SBS Vision-Sensor

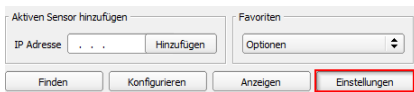
Zur Konfiguration des Vision-Sensors gehen Sie wie folgt vor.

#### Einstellungen in Vision Sensor Device Manager

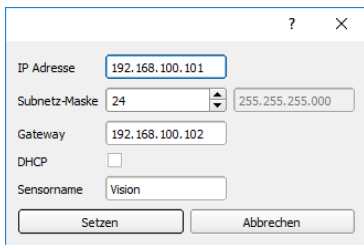


1. Starten Sie die SBS Software . Vision Sensor Device Manager wird geöffnet.

2. Drücken Sie den Button "Finden". Der Vision-Sensor wird im Fenster "Aktive Sensoren" aufgelistet.

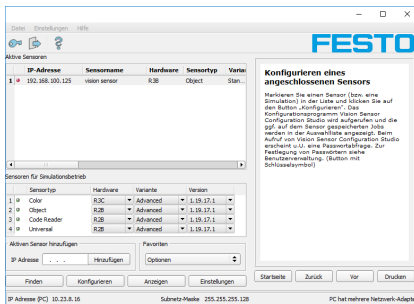


3. Drücken Sie den Button "Einstellungen". Es öffnet sich das Dialog-Fenster zur Konfiguration der IP-Adresse und des Sensornamens.

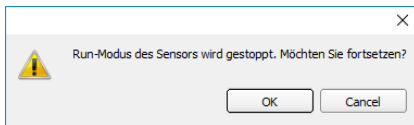


4. Vergeben Sie eine IP Adresse und einen Namen für den Sensor.

5. Drücken Sie den Button "Setzen". IP Adresse und Name sind aktualisiert.

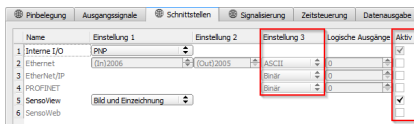


- Öffnen Sie Vision Sensor Configuration Studio indem Sie den gewünschten Sensor auswählen und anschließend den Button "Konfigurieren" drücken.



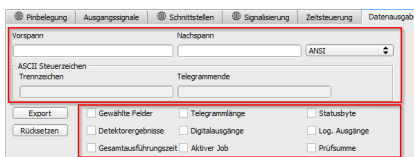
- Bestätigen Sie das nachfolgende Dialogfeld mit "OK" um Vision Sensor Device Manager zu stoppen und mit der Konfiguration in Vision Sensor Configuration Studio zu beginnen.

## Schnittstelle auswählen in Vision Sensor Configuration Studio

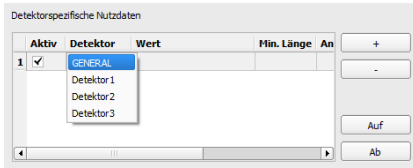


- Öffnen Sie über den Bedienschritt "Ausgabe" den Reiter "Schnittstellen".
- Aktivieren Sie die Schnittstelle indem Sie die entsprechende Checkbox in der Spalte "Aktiv" aktivieren.
- Wählen Sie in der Spalte "Einstellung 3" das Format, indem die Datenausgabe erfolgen soll.

## Telegramme / Datenausgabe definieren in Vision Sensor Configuration Studio



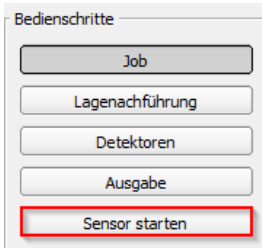
- Öffnen Sie über den Bedienschritt "Ausgabe" den Reiter "Datenausgabe".
- Stellen Sie die gewünschten Steuerzeichen für die Datenausgabe ein.
- Wählen Sie die gewünschten Auswahlfelder aus.



- Stellen Sie die gewünschten Daten ein, die ausgegeben werden sollen. Erzeugen Sie mit der Schaltfläche "+" einen neuen Eintrag.  
Funktion der Schaltflächen:
  - "+": Neuen Eintrag einfügen
  - "-": Markierten Eintrag löschen
  - "Auf", "Ab": Markierten Eintrag verschieben
- Wählen Sie in der Spalte "Detektor" den gewünschten Detektor aus.
- Wählen Sie in der Spalte "Wert" den gewünschten Detektorwert aus, der anschließend über die aktivierte Schnittstelle ausgegeben werden soll.

Weitere Informationen: Datenausgabe ([ASCII](#) / [Binär](#))

### Sensor starten



- Drücken Sie den Bedienschritt "Start Sensor". Die Daten werden auf den Vision-Sensor übertragen und der Vision-Sensor wird gestartet.



**HINWEIS:**  
Detektor muss angelegt sein.





## 4 Ethernet TCP / IP, Port 2005 / 2006

Numerische Daten, welche unter Ausgabe/Datenausgabe konfiguriert wurden, können in einem eigenen ASCII/BINAER Format ausgegeben werden.

Der Sensor ist hier der (Socket-)„Server“ und stellt die Daten über eine „Server-Socket“ Schnittstelle zur Verfügung. Hauptsächlich ist dies eine „Programmier-Schnittstelle“.

Um die Daten lesen/verarbeiten zu können muss ein „Socket-Client“ (PC, SPS, etc.) eine (Socket-t-)Verbindung (aktiv) zum Sensor aufbauen, und bekommt dann die Daten.

### Vorgehensweise, Einstellungen

#### 4.1 Beispiel: Datenausgabe vom SBS an PC / Steuerung

##### Schritt 1:

Nachdem der Job mit allen nötigen Detektoren, ggf. Lagenachführung etc. eingestellt ist, wird hier die Ethernet-Schnittstelle zur Datenausgabe aktiviert und ggf. parametrisiert.

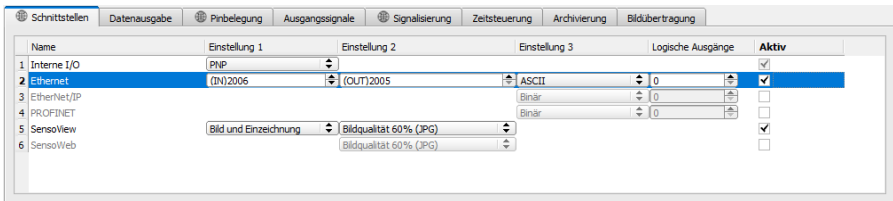


Abb. 5: Datenausgabe, Ethernet

Im Beispiel wird die Ethernet-Schnittstelle im unteren Parameter-Bereich im Reiter: „Schnittstellen“ mit einem Haken in der Checkbox „Aktiv“ aktiviert. Die Default- Einstellungen für Port Eingang (IN) = 2006 und Port Ausgang (OUT) = 2005 werden so übernommen. Hier können beliebige andere Einstellungen getroffen werden um die Datenausgabe an Ihre Netzwerkumgebung anzupassen. Dazu ggf. Ihren Netzwerkadministrator kontaktieren.

##### Schritt 2:

Im Reiter „Datenausgabe“ werden die via Ethernet Port 2005 auszugebenden Nutzdaten konfiguriert.

In diesem Beispiel sind das:

- der Vorspann „010“
- das Gesamtergebnis von Detektor 1
- der Nachspann „xxx“

Als Datenformat wurde „ASCII“ definiert, dies erleichtert die Nachvollziehbarkeit dieses Beispiels. Die Funktion mit anderen Daten, bzw. in Binär ist analog zu den hier beispielhaft gemachten Einstellungen.

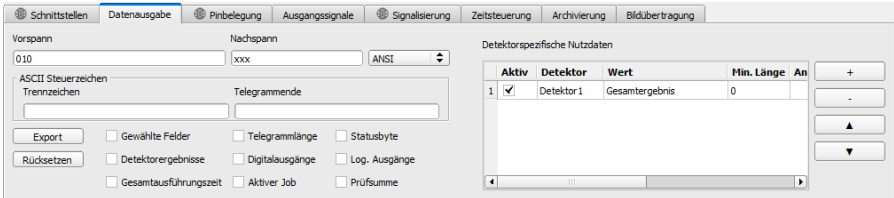


Abb. 6: Datenausgabe, Ausgabedaten konfigurieren

### Schritt 3:

Nach Start des Ethernet Tools Hercules muss der Reiter „TCP-Client“ für die Kommunikation via Ethernet mit dem Socket- Server SBS ausgewählt werden.

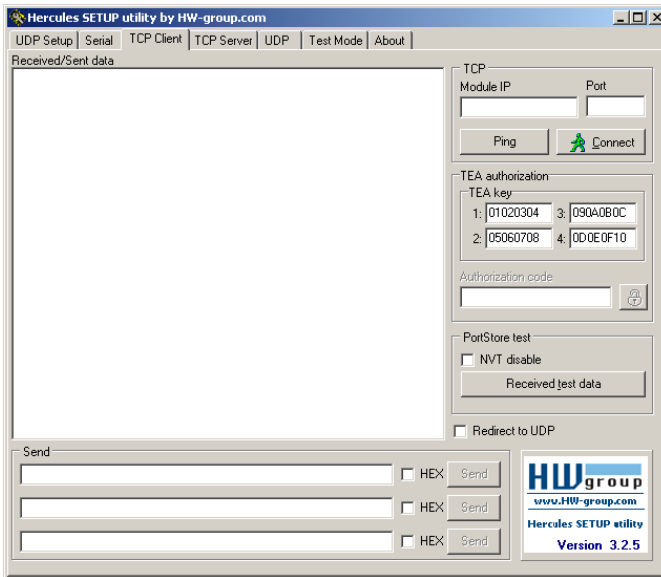


Abb. 7: Datenausgabe, Ethernet Tool / 1

Hier müssen nun noch die IP Adresse des SBS und der korrekte Port zum Datenempfang eingetragen werden.

Die IP Adresse des SBS ist in Vision Sensor Device Manager sichtbar. Siehe erste Zeile im Fenster „Aktive Sensoren“ = 192.168.60.199

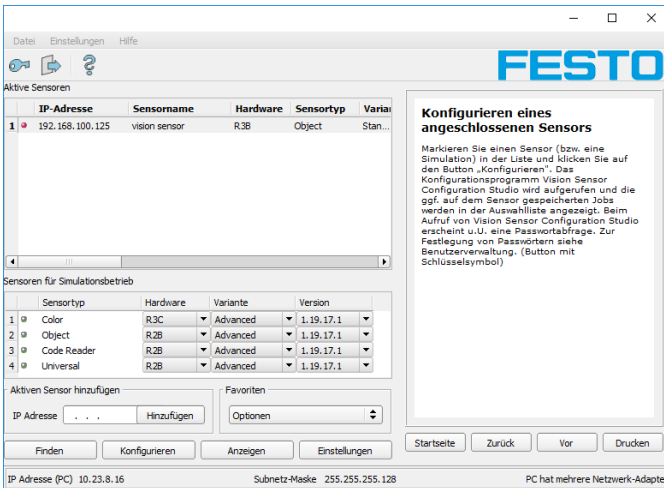


Abb. 8: Vision Sensor Device Manager, IP Adresse ...

Die Portnummer für den Ausgabeport wurde vorher unter Schritt 1 mit Port 2005 übernommen.

#### Schritt 4:

Deshalb werden folgende Einstellungen in Hercules gemacht, Module IP = 192.168.60.199, Port = 2005. Alle anderen Einstellungen bleiben auf den Defaultwerten. Mit einem Klick auf den Button „Connect“ wird auf den SBS verbunden und die Verbindung im Hauptfenster in grüner Schrift angezeigt.

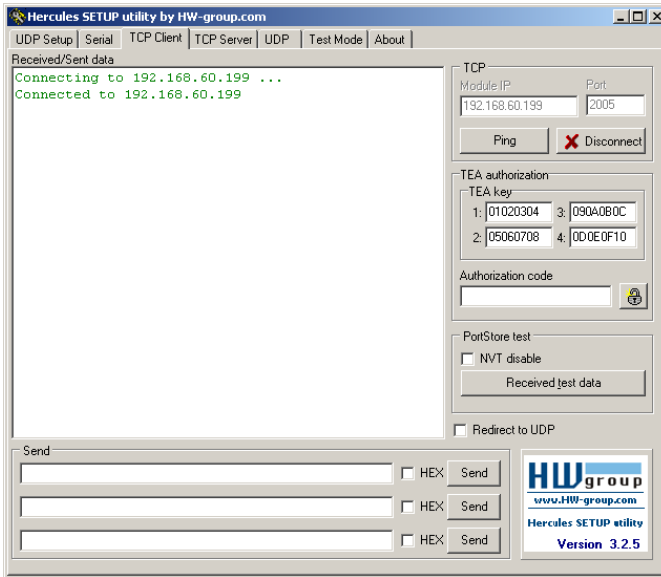


Abb. 9: Datenausgabe, Ethernet Tool / 2

**Schritt 5:**

Der SBS muss nun noch mit „Sensor starten“ von der PC- Anwendung aus gestartet werden (im späteren Betriebsfall ist der SBS direkt nach dem Einschalten im normalen Betrieb und schickt Daten wenn konfiguriert). Hier im Beispiel ist als Triggermodus = kontinuierlich eingestellt, d.h. es werden kontinuierlich Auswertungen gemacht und damit Daten gesendet. Diese sind nun im Hauptfenster von Hercules sichtbar.

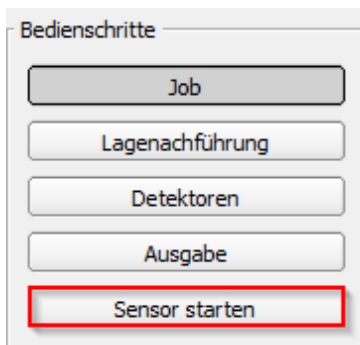


Abb. 10: Sensor starten

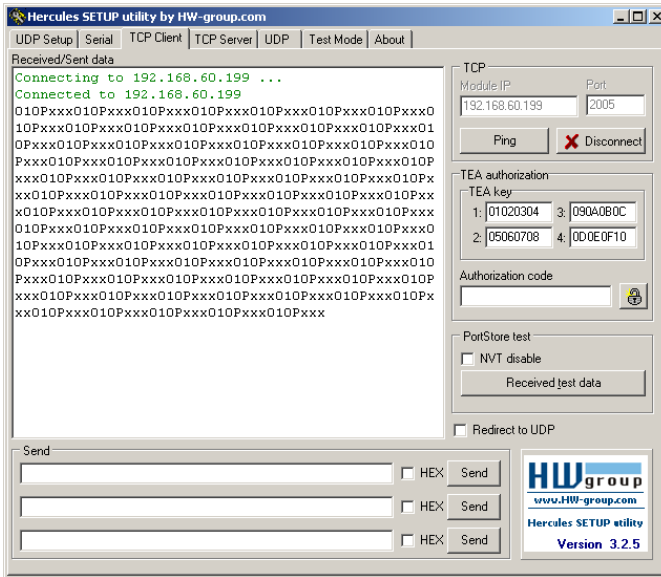


Abb. 11: Datenausgabe, Ethernet, Tool / 3

Die hier sichtbaren Daten sind wie unter „Datenausgabe“ eingestellt:

- der Vorspann „010“
- das Gesamtergebnis von Detektor 1 (hier ein „P“ für Positiv, da Prüfbedingung: Helligkeit erfüllt)
- der Nachspann „xxx“

## 4.2 Beispiel: Kommandos (Requests) von PC / Steuerung an SBS

### Mit Quittierung / Datenausgabe vom SBS

#### Schritt 1

Zur besseren Übersichtlichkeit wird hier für Beispiel 2 in den getriggerten Betrieb geschaltet. Das geschieht wie folgt: In Vision Sensor Configuration Studio unter Job/Bildaufnahme/ Triggermodus = „Trigger“ einstellen. Ansonsten bleiben die Einstellungen aus Ethernet Beispiel 1 im SBS unverändert.

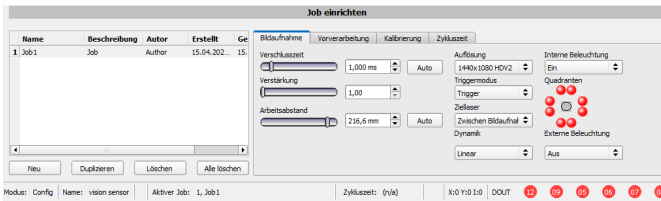


Abb. 12: Datenausgabe, Ethernet, Trigger

### Schritt 2

Zum Absetzen von Kommandos an den SBS wird die Anwendung Hercules ein zweites Mal geöffnet. Diesmal mit Port 2006 als Eingangsport des SBS auf dem dieser Kommandos empfangen kann. Alle Telegramme (Kommandos und Antwortstrings) zum und vom SBS sind in Kapitel [Übersicht Telegramme](#) beschrieben.

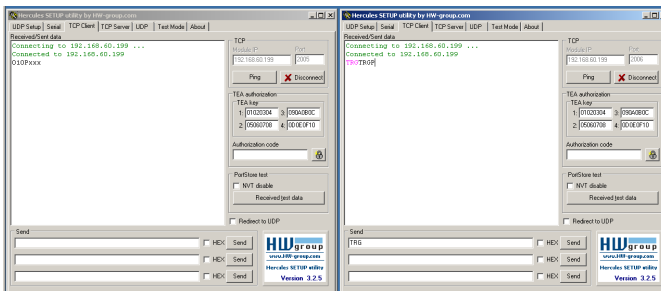


Abb. 13: Datenausgabe, Ethernet Tool / 4

Im Fenster rechts wurde von Port 2006 aus das Kommando „TRG“ (für Trigger, Kommando siehe unten erste Zeile) durch Klick auf den entsprechenden Button „Send“ an den SBS geschickt. Das Kommando wird beim Absenden im Hauptfenster in roter Schrift dargestellt. Der SBS antwortet auf Port 2006 mit einer Quittung aus Kommando „TRG“ und in diesem Falle „P“ für ein positives Ergebnis des Detektors 1 (schwarze Schrift im rechten Fenster).

Im linken Fenster schickt der SBS über Ausgabe-Port 2005 die unter Datenausgabe definierten Werte „010Pxxx“ wie auch in Beispiel Ethernet 1.

### 4.3 Beispiel: Jobschaltung von PC / Steuerung an SBS

Mit Quittierung / Datenausgabe vom SBS

#### Funktion der beiden Ethernet-Ports für Ein- und Ausgabe:

\*A: Port 2005, nur eine Richtung: Sensor » PC, alle Nutzdaten, definiert unter „Datenausgabe“

\*B: Port 2006, beider Richtungen: Sensor ↔ PC, Kommandos an mit Quittierung, + alle Respon-  
sedaten auf Kommandos (keine Nutzdaten)

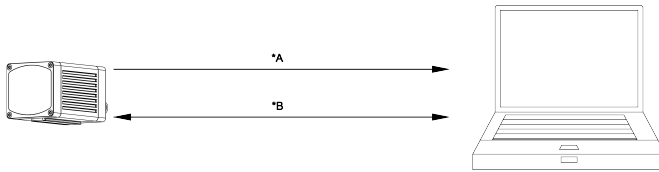


Abb. 14: Ethernet- Ports

**Schritt 1**

Zur besseren Übersichtlichkeit wird hier in den getriggerten Betrieb geschaltet. Das geschieht wie folgt: In Vision Sensor Configuration Studio unter Job / Bildaufnahme / Triggermodus = „Trigger“ einstellen. Ansonsten bleiben die Einstellungen aus Ethernet Beispiel 1 im SBS unverändert. Alle Datenausgabedefinitionen werden hier in „ASCII“ getroffen zur besseren Nachvollziehbarkeit der Beispiele.

Für dieses Beispiel müssen mindesten zwei Jobs auf dem SBS Vision-Sensor angelegt sein. Um einen neuen Job auf Basis eines bestehenden Jobs zu erstellen, können Sie die Funktion „Duplizieren“ verwenden. Passen Sie zur einfachen Überprüfung des Jobwechsels folgenden Parameter an. Später können Sie die Ausgabe frei definieren.

Für dieses Beispiel wurde Job1 definiert mit der Datenausgabe:

- Vorspann: „010“ und
- Nachspann: „xxx“

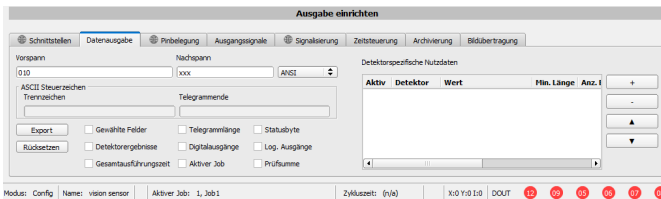


Abb. 15: Datenausgabe, Ethernet, Jobumschaltung Job 1

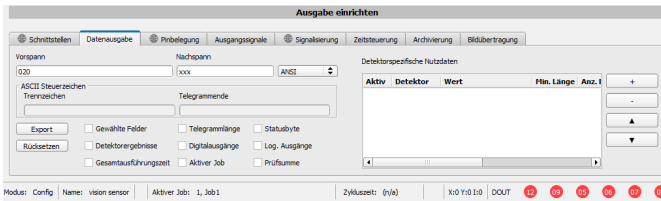


Abb. 16: Datenausgabe, Ethernet, Jobumschaltung, Job 2

## Schritt 2

Hier wurde die Anwendung Hercules zwei Mal geöffnet. Einmal mit Port 2005 (Empfangen von Ergebnissen wie in „Datenausgabe“ definiert) und Port 2006 (Kommandos + Quittung) als Eingangsport des SBS auf dem dieser Kommandos empfangen kann.

Alle Telegramme (Kommandos und Antwortstrings) zum und vom SBS sind in Kapitel [Übersicht Telegramme](#) beschrieben.

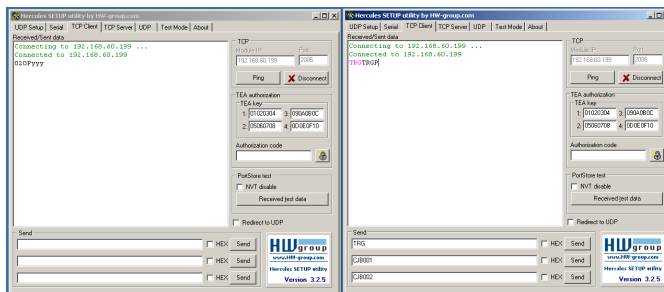


Abb. 17: Datenausgabe, Ethernet, Jobumschaltung, Tool / 1

Im rechten Fenster (Port2006) wurde das Kommando TRG (Trigger, siehe unten bei „Send“, erste Zeile) abgesetzt. Dieses wird im Hauptfenster in roter Farbe mit „TRG“ dargestellt. Der SBS antwortet sofort mit der Quittung „TRGP“ (Wiederholung des Kommandos „TRG“ und „P“ für Positiv, in schwarzer Schrift im rechten Fenster)

Im linken Fenster (Port2005) schickt der SBS, auf dem gerade der Job2 aktiv ist, den entsprechenden Ergebnisstring der unter Datenausgabe in Job 2 mit „020Pyyy“ definiert ist.



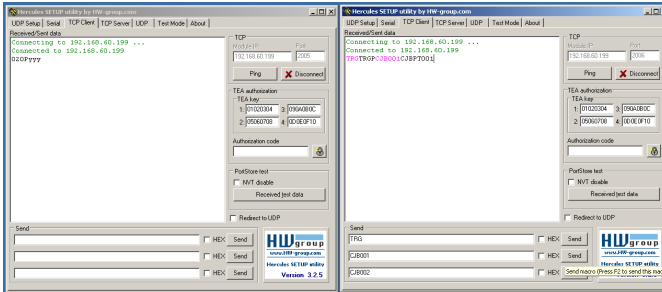


Abb. 18: Datenausgabe, Ethernet, Jobumschaltung, Tool / 2

Nun wurde im rechten Fenster (Port2006) das Kommando CJB001 (Change Job 001, 001 = Job Nr. 1, siehe unten bei „Send“, zweite Zeile) abgesetzt. Dieses wird im Hauptfenster in roter Farbe mit „CJB001“ dargestellt. Der SBS antwortet sofort mit der Quittung „CJBPT001“ (Wiederholung des Kommandos „CJB“, „P“ für Positiv, „T“ = Triggered, 001 Jobnummer auf die umgeschaltet wurde)

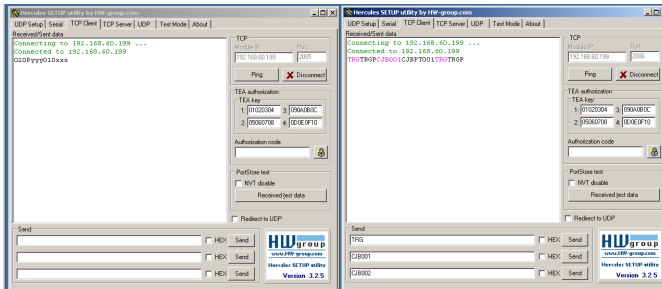


Abb. 19: Datenausgabe, Ethernet, Jobumschaltung, Tool / 3

Auf ein weiteres Trigger- Kommando TRG (siehe unten bei „Send“, dritte Zeile) wird dies im Hauptfenster wieder in roter Farbe mit „TRG“ dargestellt. Der SBS antwortet wieder sofort mit der Quittung „TRGP“ (Wiederholung des Kommandos „TRG“ und „P“ für Positiv)

Im linken Fenster (Port2005) schickt der SBS, nachdem zuvor auf Job 1 umgeschaltet wurde, nun den entsprechenden Ergebnisstring, der unter Datenausgabe in Job 1 mit „010xxx“ definiert wurde !



## 5 Service / Visualisierung

Für den SBS Vision-Sensor steht ein Service-Port (Ethernet TCP/IP Port 1998) zur Verfügung. Dieser ist unabhängig von der Konfiguration verfügbar.

### 5.1 Backuperstellung

Für das automatische Sichern und Wiederherstellen stehen folgende Telegramme zur Verfügung

- **Jobsatz einlesen** ([ASCII](#))

Mit dem Telegramm "Jobsatz einlesen" kann der Jobsatz des SBS Vision-Sensors getauscht werden. Die Jobsatz-Datei muss zuvor auf den SBS geladen werden.

- **Jobsatz sichern** ([ASCII](#))

Mit dem Telegramm "Jobsatz sichern" kann der Jobsatz des SBS Vision-Sensors gelesen werden.

### 5.2 Visualisierung

Für die Visualisierung der Applikationen stellt der SBS Vision-Sensor über den Service-Port alle Daten zur Verfügung.

Weitere Informationen: [Service \(nur auf Port 1998 und in ASCII Format verfügbar\)](#)



## 6 Telegramme SBS für PROFINET und EtherNet/IP

### 6.1 Modul 1: „Control“ (Von Steuerung an SBS)

Name in Steuerung "CTRL (3 Bytes)"


Byte Offset	Bit Adr..	Name	Datentyp	Bedeutung
0	0	Reset Error	1 Bit	Reset Error löscht den 4-Bit-Fehlercode im Status-Modul. Steigende Flanke (False → True) löscht Fehlercode.
	1	Trigger Disable	1 Bit	Dieses Bit dient zum Disable des Triggers. Gültig für Triggermodus Trigger und Freilauf. <ul style="list-style-type: none"> <li>False (0): Trigger aktiviert.</li> <li>True (1): Trigger deaktiviert.</li> </ul> Falls der Digital-Eingang "Trigger enable" genutzt wird, müssen zum Akzeptieren eines Triggers beide Bedingungen (Digitaler Eingang "Hardware Trigger" und "Trigger Disable-Bit") auf "Enable" gesetzt sein.
	2	Trigger	1 Bit	Steigende Flanke (False → True): Trigger, wird sofort ausgeführt. Falls der Trigger nicht ausgeführt werden konnte, bleibt das Trigger-Acknowledge-Bit False und das Modul Error Status weist den Fehlercode "1: Failure trigger request" auf. Siehe auch Timing-Diagramm, Kap. <a href="#">Fall: Trigger nicht möglich (not ready)</a>

Byte Offset	Bit Adr..	Name	Datentyp	Bedeutung
	3	Change job	1 Bit	<p>Steigende Flanke (False → True): Umschalten auf Job mit Nummer "Job number" aus Control-Modul.</p> <p>Beim Ausführen dieses Requests kann es zu Verzögerungen kommen.</p> <p>Nach einem erfolgreichen Jobwechsel zeigt das Byte "Job number" im Status-Modul den gleichen Wert wie im Control-Modul.</p> <p>Falls der Jobwechsel nicht durchgeführt werden konnte (aufgrund eines Fehlers, z.B. falsche Jobnummer), weist das Error Status Modul den Fehlercode "2: Failure Change Job" auf (und Ready bleibt False!). Siehe auch Timing-Diagramm, Kap. <a href="#">Fall: Jobchange nicht möglich (z.B. falsche Job Nummer)</a></p>
	4	Switch-to-Run	1 Bit	<p>Steigende Flanke (False → True): "Switch-to-Run" wird ausgeführt. Erfolg oder Fehler des Switch-to-Run-Requests wird angezeigt im Error Status Modul (Fehlercode "3: Failure Switch to run request") und Bit "Operation Mode". Siehe auch Timing-Diagramm, Kap. <a href="#">Fall: Switch to run nicht möglich</a></p>
	5-7	Reserve		
1		Reserve	1 Byte	
2		Job number	U8	<p>Jobnummer auf die umgeschaltet werden soll, bei der steigenden Flanke des Change-Job-Bit. Binärer Wert 1-255 für "Job number Change". 0 steht für: "Keine Umschaltung", auch wenn das Change-Job-Bit wechselt.</p>

[Timing-Diagramme zur SBS Kommunikation](#)

## 6.2 Modul 2: „Status“ (Von SBS an Steuerung)

Name in Steuerung „STAT (6 Bytes)“

Byte Offset	Bit Adr.	Name	Datentyp	Beschreibung
0	0	Ready	1 Bit	<p>SBS ist bereit für die nächste Auswertung. Ready=1.</p> <div style="display: flex; align-items: center;">  <p><b>Achtung:</b> Das Ready Bit ist ausschließlich zur Anzeige der Bereitschaft des SBS Vision-Sensors für die nächste Auswertung reserviert. Es ist nicht geeignet zur Anzeige, dass eine Auswertung abgeschlossen ist, bzw. die Ergebnisse einer Auswertung vorliegen!</p> </div>
	1	Reserve	1 Bit	
	2	Trigger acknowledge	1 Bit	Acknowledge (Bestätigung) für erfolgreichen Trigger-Request (via Trigger-Bit im Control-Modul). Acknowledge wird gelöscht als Response auf das Löschen des Trigger-Bit. Falls der Trigger nicht ausgeführt werden konnte, bleibt das Trigger-Acknowledge-Bit false.
	3	Change Job acknowledge	1 Bit	Acknowledge (Bestätigung) für den erfolgten Change-Job-Request (via Change-Job-Bit im Control-Modul) – unabhängig von dessen Erfolg. Acknowledge wird gelöscht, sobald das Change-Job-Request-Bit gelöscht wurde. Erfolg oder Fehler des Change-Job-Request wird angezeigt im Bitfeld "Error" (Fehlercode "2: Failure change Job") und im Byte "Job-number" im Modul Status. Bei Verzögerungen bei der Ausführung des Job-Change kann dieses Acknowledge-Bit ebenfalls verzögert gesetzt werden.

Byte Offset	Bit Adr.	Name	Datentyp	Beschreibung
	4	Switch to run acknowledge	1 Bit	Acknowledge (Bestätigung) für den erfolgten Switch-to-Run-Request (via Switch-to-Run-Request-Bit im Control-Modul). Acknowledge wird gelöscht, sobald das Request-Bit gelöscht wurde. Erfolg oder Fehler von Switch-to-Run-Request wird angezeigt im Bitfeld "Error" (Fehlercode "3: Failure switch to run Request") und Bit "Operation Mode". Acknowledge wird gesetzt, nachdem Vision Sensor Configuration Studio geschlossen und der Job aus dem Flash geladen wurde, oder wenn ein Fehler auftrat.
	5-7	Reserve		
1		Reserve	1 Byte	
2	0	Digital Results	1 Bit	12 RDBU
	1		1 Bit	09 RD
	2		1 Bit	05 PK
	3		1 Bit	06 YE
	4		1 Bit	07 BK
	5		1 Bit	08 GY
	6	Reserve	1 Bit	Dieses Byte wird gefüllt mit den Ergebnissen der digitalen Schaltausgänge. Die Bitposition ist fix. Der Wert des Ausgangs wird definiert im Reiter: Ausgabe/Ausgangssignale, Spalte: „Logischer Ausdruck“ in Vision Sensor Configuration Studio. Falls nicht ausgewählt als Ergebnis-Ausgabe-Pin, oder falls kein gültiger logischer Ausdruck zugewiesen ist, ist der Wert = 0.
	7	Reserve	1 Bit	
3		Job Number	U8	Nummer des aktuellen Jobs: Jobnummer 1-255
4		Image ID	U8	Image ID (0-255) wird bei jeder Job-Ausführung um 1 erhöht, unabhängig von der Trigger-Quelle.



Byte Offset	Bit Adr.	Name	Datentyp	Beschreibung
5	0-3	Error	4 Bit	<p>4 Bit Fehlercode (dezimal). Zeigt Fehler bei Requests via Control-Modul oder -Systemfehler an. Der Fehlercode kann durch "Reset error" zurückgesetzt werden oder wird durch nächsten Fehler überschrieben. Im Falle eines Archivierungsfehlers (8), können Sie auch ohne "Reset error" fortfahren.</p> <p>0: Kein Fehler            1: Fehler Trigger Request (Sensor nicht Ready)            2: Fehler Change Job            3: Fehler Switch-to-Run            4: Request abgelehnt            5: Fehler Schnittstelle in Job nicht aktiv            7: Fokus-Sperrzeit            8: Fehler Archivierung            15: System-Fehler</p>
	4	Trigger mode	1 Bit	<p>1 = Freilauf            0 = Trigger</p>
	5	Reserve	1 Bit	
	6	Operation mode	1 Bit	<p>1 = Run            0 = Config</p>
	7	Reserve	1 Bit	

## 6.3 Modul 3: „Data“ (Von SBS an Steuerung)

Name in Steuerung „DATA (2 + 8 / 16 / ... / 192 / 252 Bytes)“

Byte Offset	Bit Adr.	Name	Datentyp	Beschreibung
0		Image ID	U8	Image ID (0 - 255) wird bei jeder Job-Ausführung um 1 erhöht, unabhängig von der Trigger-Quelle.
1	0	Result data overrun	1 Bit	Ergebnis-Daten wurden abgeschnitten. 1: Data overrun = abgeschnitten 0: No overrun
	1 - 7	Reserve	7 Bit	
2		Result data	Byte- array	Daten wie definiert in Vision Sensor Configuration Studio unter "Ausgabe/Datenausgabe/Detektorspezifische Nutzdaten". Bei Nutzung von PROFINET muss im Reiter Schnittstellen "Binär" aktiviert sein.

## 6.4 Modul 4: „Request“ (Von Steuerung an SBS)

Name in Steuerung „REQU (4 + 8 / 16 / ... / 192 / 250 Bytes)“

Byte Offset	Bit Adr.	Name	Datentyp	Bedeutung
0	1	Key	1 Byte	Request key (Request counter)
1	1	Reserve	1 Byte	Reserve
2	1	Reserve	1 Byte	Reserve
3	1	Reserve	1 Byte	Reserve
4		Request Data	Byte Array	Weitere Informationen: <a href="#">Übersicht Telegramme</a>

## 6.5 Modul 5: „Response“ (Von Steuerung an SBS)

Name in Steuerung „RESP (4 + 8 / 16 / ... / 192 / 250 Bytes)“

Byte Offset	Bit Adr.	Name	Datentyp	Beschreibung
0		Key	U8	Response key = gespiegelt vom Request
1	0	Result Data overrun	1 Bit	Response-Daten wurden abgeschnitten
	1-7	Reserve	7 Bit	
2		Reserve	1 Byte	
3		Reserve	1 Byte	
4		Result Data	Byte Array	Weitere Informationen: <a href="#">Übersicht Telegramme</a>

## 6.6 Start- / Ende-Kriterien je Telegramm

<b>Telegramm (Modul „Control“)</b>	<b>Start-Bedingung (Modul „Status“)</b>	<b>Bestätigung Annahme (Modul „Status“)</b>	<b>Bestätigung Ausführung (Modul „Status“)</b>
Trigger	Ready = True	Trigger acknowledge = True	Image ID wechselt
Change Job	/	Change Job acknowledge = True	Job number wechselt
Switch-to-Run	Operation Mode = False	Switch-to-Run acknowledge = True	Operation Mode = True



## 7 Timing-Diagramme zur SBS Kommunikation

### Fall: Trigger ok

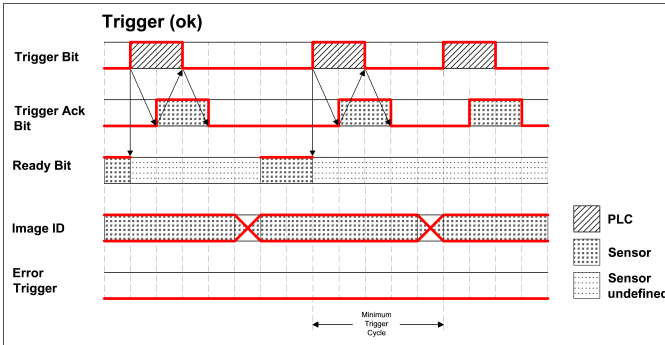


Abb. 20: Timing Trigger ok

### Fall: Trigger nicht möglich (not ready)

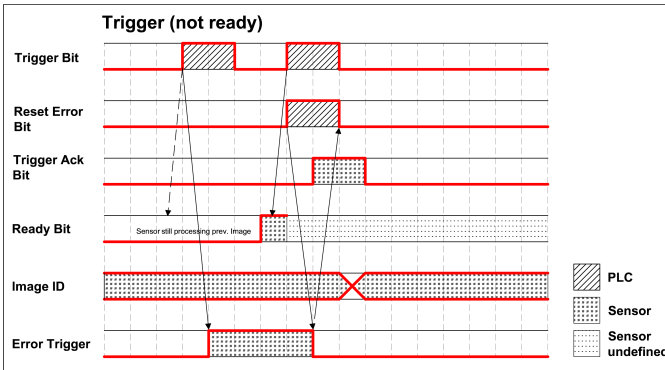


Abb. 21: Timing Trigger not ready

## Fall: Jobchange ok

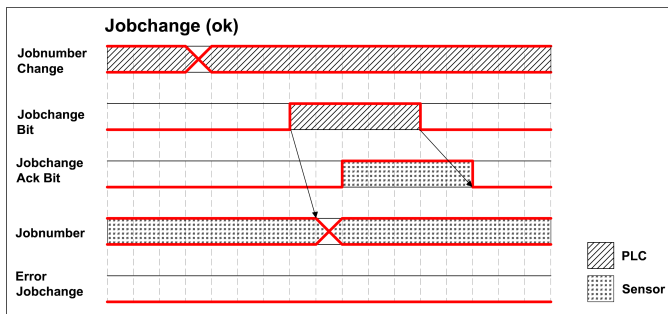


Abb. 22: Timing Jobwechsel ok

## Fall: Jobchange delayed

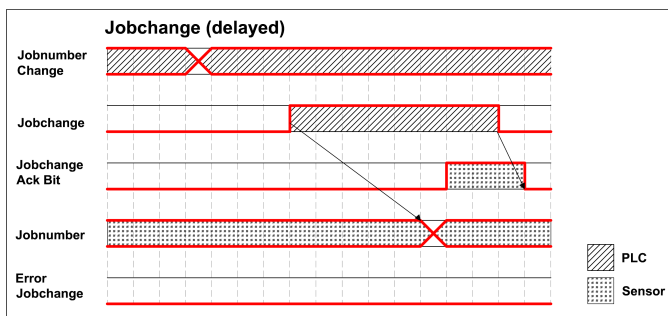


Abb. 23: Timing Jobwechsel verzögert



**Fall: Jobchange nicht möglich (z.B. falsche Job Nummer)**

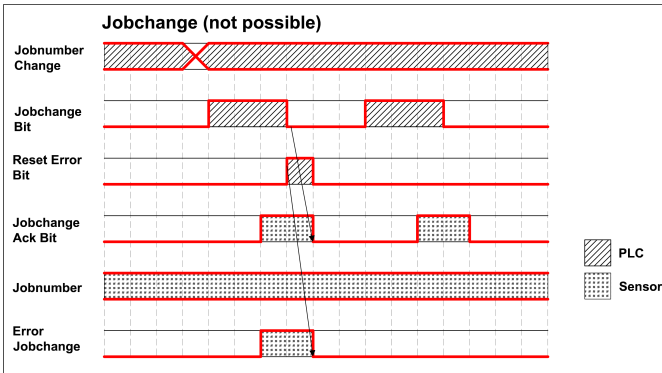


Abb. 24: Timing Jobwechsel nicht möglich

**Fall: Switch to run ok**

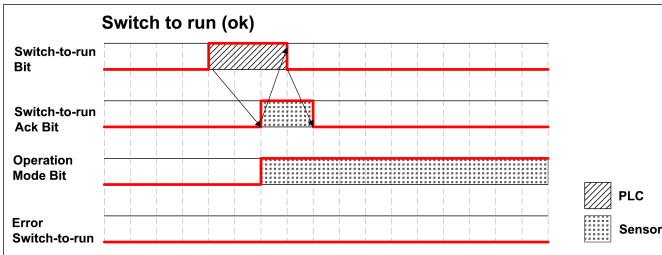


Abb. 25: Timing Switch to run ok

## Fall: Switch to run nicht möglich

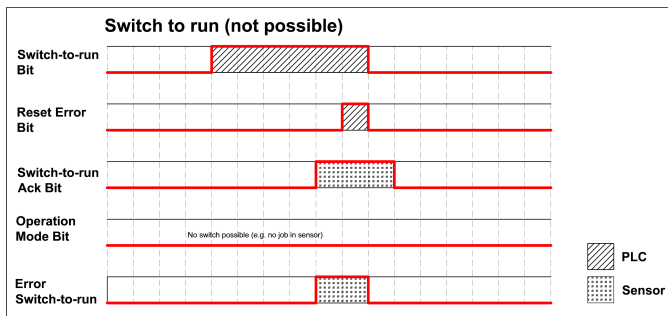


Abb. 26: Switch to run nicht möglich

## 8 Request-Sequenzen

### Wichtige Empfehlungen für SPS-Programmierer

1. Reihenfolge der Requests einhalten
2. Immer die komplette Ausführung einer Aktion abwarten bevor die nächste Aktion ausgelöst wird. Die komplette Ausführung ist dann erfolgt wenn beim Trigger Request die Image ID wechselt, bzw. bei den anderen Requests das entsprechende Acknowledge Bit gesetzt wurde.

**HINWEIS:**

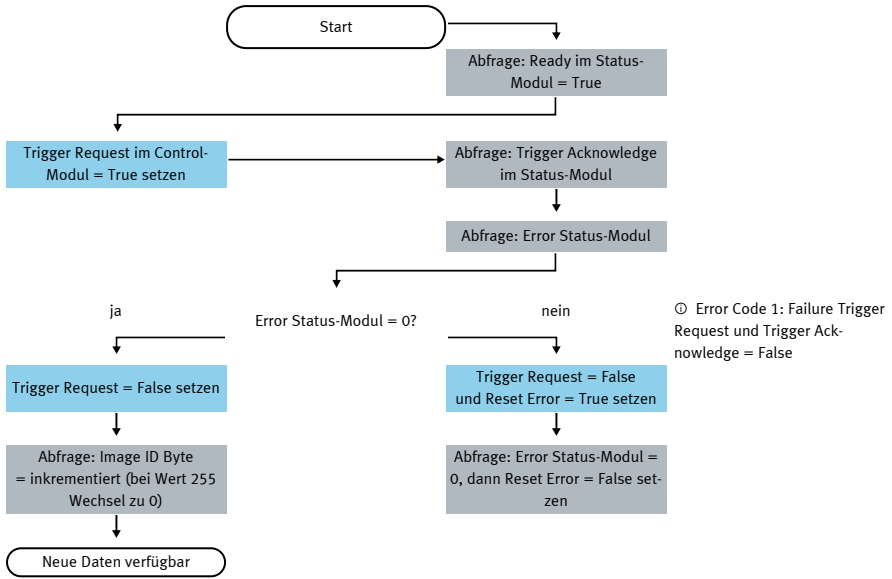
Die komplette Ausführung einer Aktion kann nicht auf Grund der Low-/ High-Wechsel von READY als sicher erkannt werden, da es, wegen evtl. langer Zykluszeiten zwischen SPS und (z.B. 32ms), dazu kommen kann, dass READY nie Low wird.

3. READY sollte immer High sein bevor ein Trigger Request gesendet wird

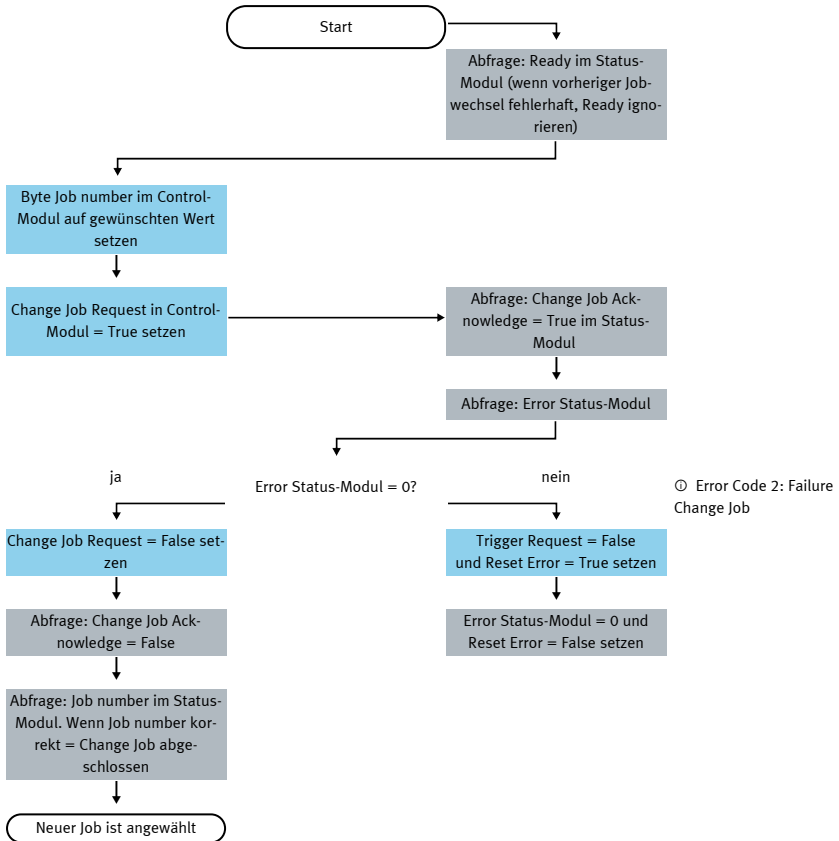
### Akzeptieren / Verwerfen von Requests des Control-Moduls

1. Request ist akzeptiert mit steigendem Acknowledge Bit
2. Request ist verworfen wenn Error Bit gesetzt.
3. Request ist verworfen ohne Error Bit und Acknowledge Bit, wenn der Sensor noch den vorherigen Request bearbeitet und für diesen noch kein Acknowledge gesetzt wurde. (d.h. der empfohlene „Handshake“ nicht befolgt wurde)

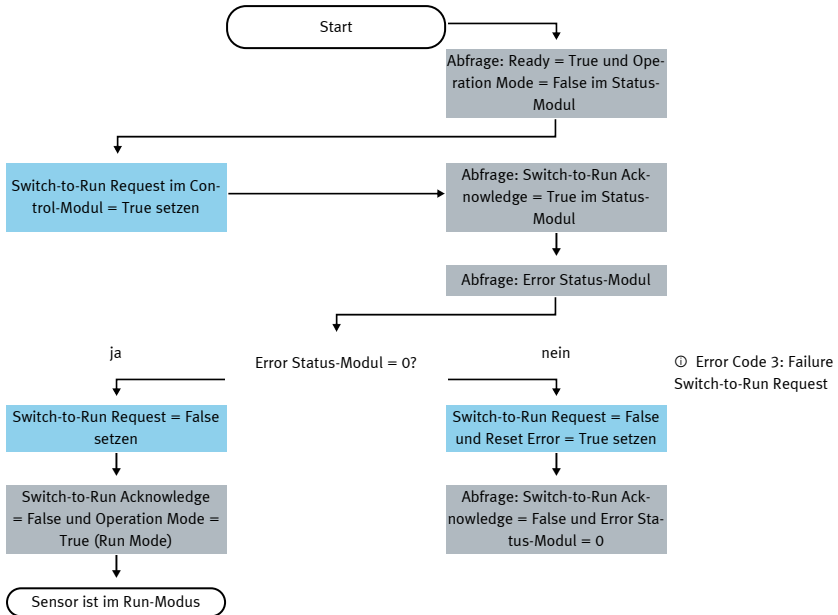
## 8.1 Trigger Request Reihenfolge



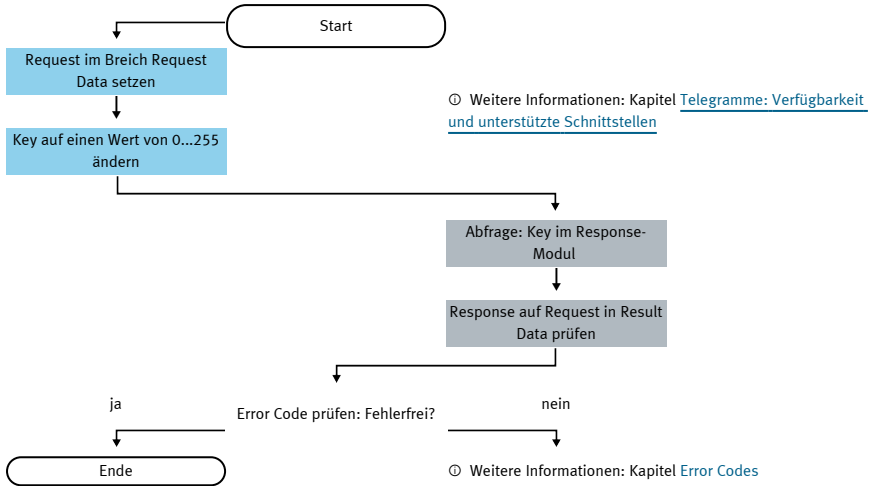
## 8.2 Change Job Request Reihenfolge



## 8.3 Switch to Run Reihenfolge



### 8.4 Reihenfolge für Requests via Request / Response Modul



**Weitere Informationen:**

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Fehlercodes](#)

**Error Reset (dargestellt im Usecase “Jobchange not possible”)**

1. Reset durch “Reset Error Bit”
2. Error Bits werden überschrieben durch neue Error Bits





## 9 PROFINET

Dieses Kapitel beschreibt den Betrieb des SBS Vision-Sensors mit PROFINET.

### 9.1 Konfigurationsbeispiel Siemens S7-1200 TIA 12

Diese Beschreibung zeigt alle SPS Screenshots in englischer Sprache, ggf. TIA Software auf Englisch umstellen.

#### 9.1.1 Neues Projekt anlegen

Projekt anlegen unter: Project/Create new project

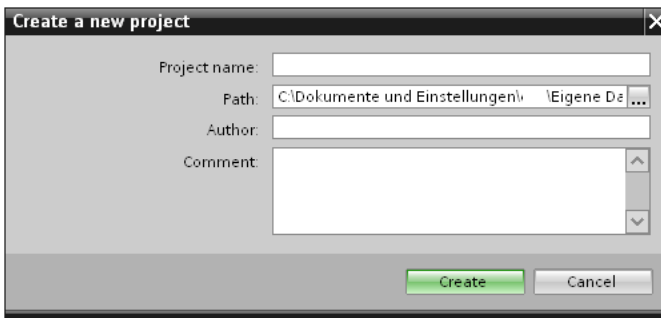


Abb. 27: PROFINET Neues Projekt anlegen

#### 9.1.2 GSD-Datei auswählen

Zunächst muss eine PROFINET-fähige Steuerung zum Projekt hinzugefügt werden.

Um die PROFINET-Funktionen des SBS Vision-Sensors nutzen zu können, muss die entsprechende SBS-GSD-Datei in der jeweils aktuellen Version installiert werden. Dies erfolgt unter: Options/Install general station description file. Die GSD-Datei ist im Installationspfad des SBS unter: ...\\Festo\\SBS Vision Sensor\\Tools\\PROFINET zu finden, und steht auch zum Download unter [www.festo.com](http://www.festo.com) bereit.

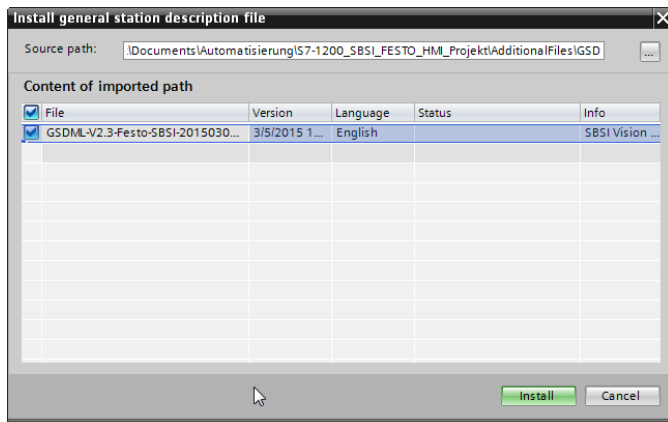


Abb. 28: GSD-Datei auswählen und installieren

### 9.1.3 SBS Vision-Sensor zum Projekt hinzufügen

Die SBS Module werden im Hardware-Katalog unter: Other field devices/PROFINET IO/Sensors/ Festo Corporation hinzugefügt.

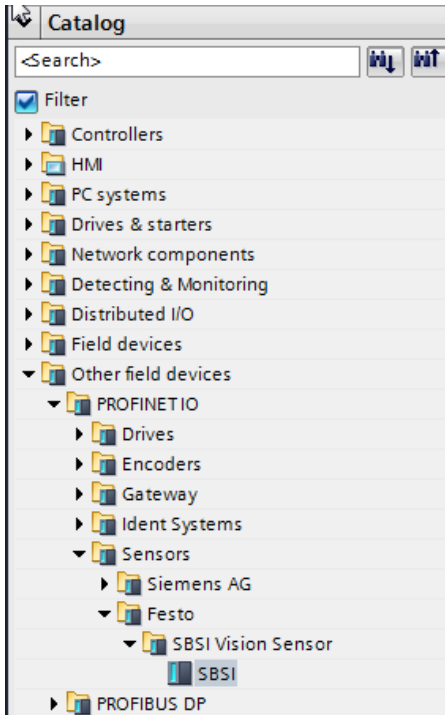


Abb. 29: SBS zum Projekt hinzufügen

### mit SPS verbinden

Per Drag and Drop kann nun ein Modul vom Katalog in die Network View gezogen werden. Der wird über PROFINET mit der Steuerung verbunden (Reiter Network View).

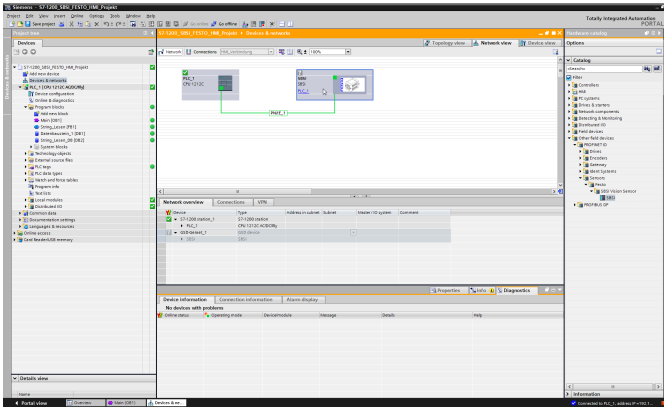


Abb. 30: SBS mit SPS verbinden

### Einfügen der I/O Daten

Im Reiter „Device View“ sind per Default die Module CTRL (Control) und STAT (Status) aktiviert. Optional kann ein DATA (Data Modul) mit einer bestimmten Nutzgröße hinzugefügt werden.

Im Beispiel: 2 Byte + 16 Byte Nutzdaten (1 Byte: Image ID, 1 Byte: Result data overrun (siehe [Modul 3: „Data“ \(Von SBS an Steuerung\)](#)), + 16 Byte Daten). Sind die Daten länger als der definierte Bereich werden diese abgeschnitten (Result data overrun = 1), sind sie kürzer wird der Rest bis 16 Byte mit 00h aufgefüllt.

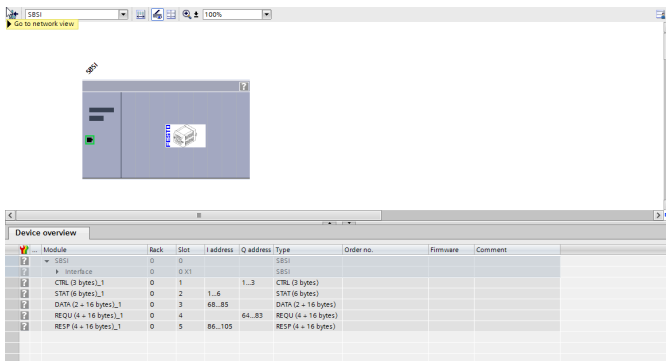


Abb. 31: I/O Daten einfügen

## IP Adresse setzen

### Möglichkeit 1: Im Projekt

Die IP Adresse des kann über das Projekt in der Steuerung vergeben werden. Dazu Option „Set IP address in the project“ auswählen und IP eingeben. Adresse aus Feld „IP address“ wird in den geschrieben. Die IP Adresse der Steuerung und des müssen ungleich sein, aber korrespondieren, d.h. im gleichen Adressraum liegen.

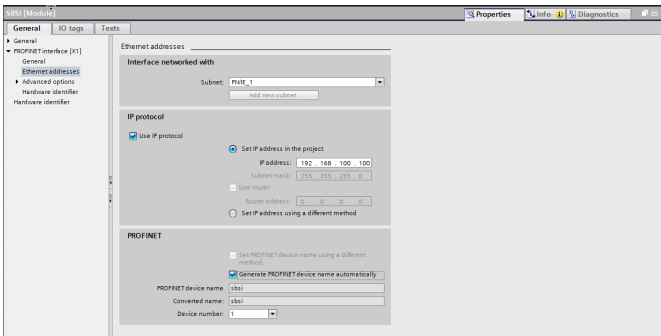


Abb. 32: IP Adresse im Projekt setzen

Der kann auch ohne laufendes PROFINET genutzt, und daher via Vision Sensor Device Manager konfiguriert werden. Stimmt die IP des mit der im TIA Projekt nicht überein, so wird die Steuerung das Setzen der IP veranlassen. Dabei wird die ursprüngliche Konfiguration im mit 0.0.0.0 übergeschrieben. Das heißt die IP wird zwar korrekt gesetzt, die IP- Konfiguration (wichtig für einen Neustart evtl. ohne angeschlossene Steuerung) aber gelöscht.

### Möglichkeit 2: In Vision Sensor Device Manager

Die IP- Adresse des kann auch über Vision Sensor Device Manager gesetzt werden. Dazu Option „Set IP Address using a different method“ im SPS / TIA Interface auswählen. IP Adresse über Vision Sensor Device Manager setzen (S. Kapitel: [Einstellungen in Vision Sensor Device Manager](#)).

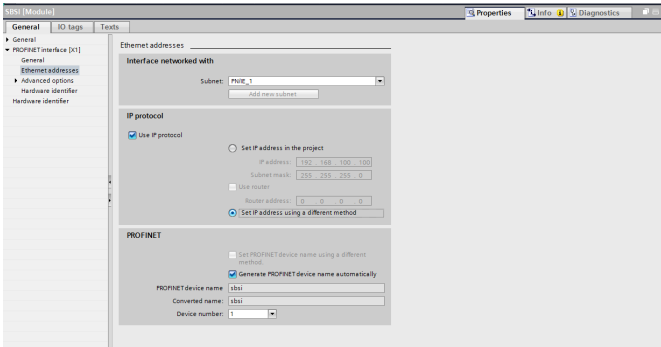


Abb. 33: IP Adresse des SBS in Vision Sensor Device Manager setzen, Einstellungen dazu im SPS/TIA Interface

## Name im TIA Portal setzen

Um den Namen des aus dem TIA Portal heraus zu setzen gibt es zwei Möglichkeiten.

### Name automatisch generieren

Der PROFINET Name des SBS kann in der Steuerung automatisch generiert werden. Option: „Generate PROFINET device name automatically“ übernimmt Name aus dem Projekt.

### Name manuell eingeben

Wenn die Option „Set PROFINET device name using a different method“ aktiviert ist kann ein beliebiger Name editiert werden.

Info: Im Feld „Converted name“ wird ein abweichender Name als eingegeben angezeigt, der dann auch so benutzt wird. Da im PROFINET nicht alle Zeichen genutzt werden können ist eventuell eine Konvertierung notwendig und wird hier automatisch gemacht (Namen müssen DNS kompatibel sein, s. dazu auch [Kap. Einstellungen in Vision Sensor Device Manager](#)).

Wird der Name des SBS Vision-Sensors über das TIA Portal gesetzt, muss dieser mit dem Tool "PROFINET device name" (wie in Kapitel [Name in SBS schreiben](#) beschrieben) auf den Sensor geschrieben werden.

Der PROFINET Name im Projekt und im müssen übereinstimmen.

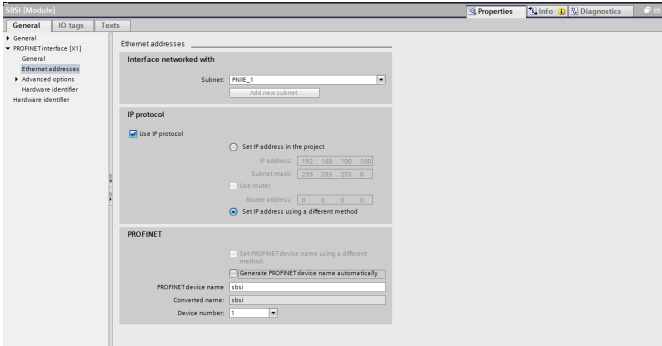


Abb. 34: Name im Projekt setzen

### 9.1.4 Name in SBS schreiben

Um eine Kommunikation aufbauen zu können, muss, wenn dieser aktualisiert werden soll, der PROFINET-Name noch in den SBS geschrieben werden.

Dies geschieht mit dem Tool: Online/Assign PROFINET device name. Dazu entsprechendes Gerät (SBS) auswählen und Name mit „Assign name“ übernehmen.

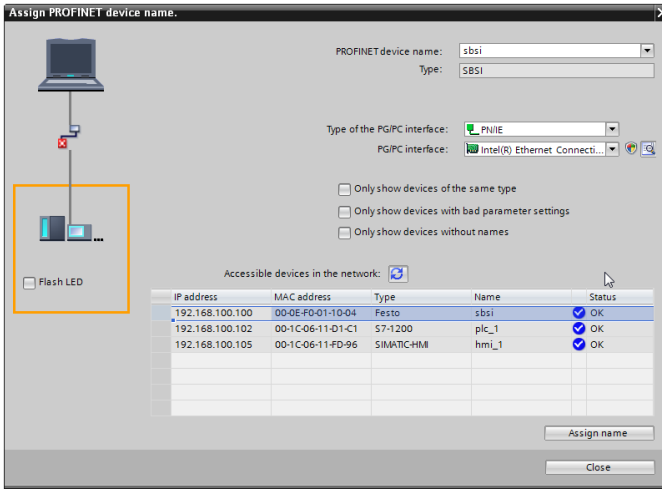


Abb. 35: Name in SBS schreiben

## 9.1.5 Projekt auf Steuerung laden

Zum Abschließen der Konfiguration und Übernehmen der Änderungen das Projekt 1. übersetzen, und 2. auf die Steuerung laden.



Abb. 36: Projekt übersetzen und auf Steuerung laden

## 9.1.6 Zuordnung der Ausgangsdaten

Die Zuordnung der Ausgangsdaten des SBS Vision-Sensors zu den Daten im PROFINET-Protokoll kann auf folgende Weise erfolgen:

**Schritt 1) Die Startadresse für eine Eingangsvariable kann aus „Device Overview“ entnommen werden.**

	Name	Address	Display format	Monitor value	Modify value
1		%I868	Hex	16#00	<input type="checkbox"/>
2		%I869	Hex	16#00	<input type="checkbox"/>
3	*Data1*	%I870	Hex		<input type="checkbox"/>
4	*Data2*	%I871	Hex		<input type="checkbox"/>
5	*Data3*	%I872	Hex		<input type="checkbox"/>
6	*Data4*	%I873	Hex		<input type="checkbox"/>
7	*Data5*	%I874	Hex		<input type="checkbox"/>
8	*Data6*	%I875	Hex		<input type="checkbox"/>
9	*Data7*	%I876	Hex		<input type="checkbox"/>
10	*Data8*	%I877	Hex		<input type="checkbox"/>
11	*Data9*	%I878	Hex		<input type="checkbox"/>
12	*Data10*	%I879	Hex		<input type="checkbox"/>
13	*Data11*	%I880	Hex		<input type="checkbox"/>
14	*Data12*	%I881	Hex		<input type="checkbox"/>
15	*Data13*	%I882	Hex		<input type="checkbox"/>
16	*Data14*	%I883	Hex		<input type="checkbox"/>
17	*Data15*	%I884	Hex		<input type="checkbox"/>
18	*Data16*	%I885	Hex		<input type="checkbox"/>
19		<Add new>			<input type="checkbox"/>

Abb. 37: Variablen-tabelle

**Schritt 2) Variablen-tabelle in der SPS erstellen**

Module	Rack	Slot	I address	Q address	Type	Order no.	Firmware	Comment
SBS1	0	0			SBS1			
Interface	0	0 X1			SBS1			
CTRL (3 bytes)_1	0	1		1...3	CTRL (3 bytes)			
STAT (6 bytes)_1	0	2	1..6		STAT (6 bytes)			
DATA (2 + 16 bytes)_1	0	3	68...85		DATA (2 + 16 bytes)			
REQU (4 + 16 bytes)_1	0	4	84...83		REQU (4 + 16 bytes)			
RESP (4 + 16 bytes)_1	0	5	86...105		RESP (4 + 16 bytes)			

Abb. 38: Geräte-Übersicht



**Schritt 3) Konfiguration in Vision Sensor Device Manager erstellen und konfiguriertes Protokoll als CSV Datei abspeichern.**

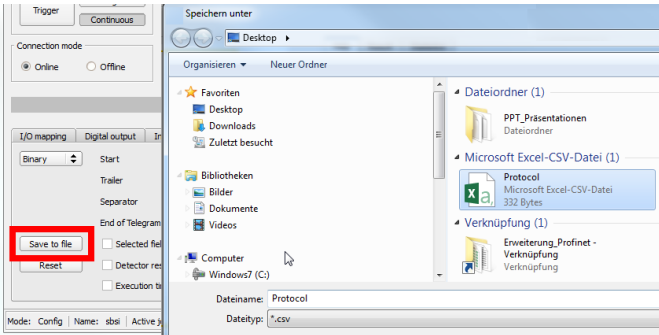


Abb. 39: Ausgabeformat als CSV- Datei speichern

**Schritt 4) Datei mit Textprogramm öffnen**

Byte position	Data type	Field	Detector name	Value	Length	Detector num	Detector type
1	Byte	Detector	Detector1	Overall result	1	1	Contour
3	Integer	Detector	Detector1	Pos X	4	1	Contour
4	Integer	Detector	Detector1	Pos Y	4	1	Contour
5	Integer	Detector	Detector1	Angle	4	1	Contour

Abb. 40: Ausgabeprotokoll in Excel- Darstellung

Für die Beschreibung des Formats der PROFINET Data- Modul, siehe [Modul 3: „Data“ \(Von SBS an Steuerung\)](#)

## Schritt 5) Es ergibt sich die folgende Zuordnung zwischen den Eingangsdaten der SPS

Name	Address	Display format	Monitor value	Modify value	Comment
	%B68	Hex	16#01		
	%B69	Hex	16#00		
"Data1"	%B70	Hex	16#01		
"Data2"	%B71	Hex	16#00		
"Data3"	%B72	Hex	16#03		
"Data4"	%B73	Hex	16#98		
"Data5"	%B74	Hex	16#C6		
"Data6"	%B75	Hex	16#00		
"Data7"	%B76	Hex	16#05		
"Data8"	%B77	Hex	16#98		
"Data9"	%B78	Hex	16#95		
"Data10"	%B79	Hex	16#FF		
"Data11"	%B80	Hex	16#FF		
"Data12"	%B81	Hex	16#FF		
"Data13"	%B82	Hex	16#78		
"Data14"	%B83	Hex	16#00		
"Data15"	%B84	Hex	16#00		
"Data16"	%B85	Hex	16#00		
<Add new>					

Abb. 41: Eingangsdaten SPS

... und dem konfigurierten Protokoll:

Byte position	Data type	Field	Detector name/value	Length	Detector num	Detector type
1	Byte	Detector	Overall result	1	1	Contour
2	Integer	Detector	Pos X	4	1	Contour
3	Integer	Detector	Pos Y	4	1	Contour
4	Integer	Detector	Pos Z	4	1	Contour
5	Integer	Detector	Angle	4	1	Contour

Abb. 42: Im Vision-Sensor konfiguriertes Protokoll

### Umrechnung von Binär- Werten

Alle detektorspezifischen Nutzdaten mit Nachkommastellen werden als ganze Zahlen, mit 1000 multipliziert, übertragen, und müssen nach Datenempfang deshalb wieder durch 1000 geteilt werden. Die Werte werden im Format „Big Endian“ übertragen. Die Länge richtet sich nach dem Wert, z.B. Score 32 Bit (DWord).

Name	Address	Display format	Monitor value	Modify value	Comment
	%B68	Hex	16#01		
	%B69	Hex	16#00		
"Data1"	%B70	Hex	16#01		
"Data2"	%B71	Hex	16#00		
"Data3"	%B72	Hex	16#03		
"Data4"	%B73	Hex	16#98		
"Data5"	%B74	Hex	16#C6		
"Data6"	%B75	Hex	16#00		
"Data7"	%B76	Hex	16#05		
"Data8"	%B77	Hex	16#98		
"Data9"	%B78	Hex	16#95		
"Data10"	%B79	Hex	16#FF		
"Data11"	%B80	Hex	16#FF		
"Data12"	%B81	Hex	16#FF		
"Data13"	%B82	Hex	16#78		
"Data14"	%B83	Hex	16#00		
"Data15"	%B84	Hex	16#00		
"Data16"	%B85	Hex	16#00		
<Add new>					

Name	Address	Display format	Monitor value	Modify value	Comment
"Data DWord Pos..."	%D71	DEC+	409414		
"Data DWord Pos 1"	%D75	DEC+	423934		
"Data DWord Angla"	%D79	DEC+	-128		
<Add new>					

Results				Statistics													
Detector	Score	Time	Detector I	No. objects	No. of valid objects	Score	Position X [µm]	Position Y [µm]	Angle	Scale	Delta pos.X [µm]	Delta pos.Y [µm]	Delta angle	Position control	Count	Reset	
1 Detector 1	99.8	27ms	Contour			1	99.8	403.4	402.8	-0.1°	1	0.0	-0.2	-0.1°	Off	1	Reset
				Fail	1	100.00%											
				Fail	0	0.00%											
				Minimum resolution time	4ms												
				Minimum resolution time	4ms												
				Minimum resolution time	4ms												
				Average resolution time	4ms												



## 10 EtherNet/IP

Dieses Kapitel beschreibt den Betrieb des SBS Vision-Sensors mit EtherNet/IP.

### 10.1 Konfigurationsbeispiel Rockwell CompactLogix™

Zum Datenaustausch zwischen SBS Vision-Sensor und SPS via EtherNet/IP werden im Folgenden die erforderlichen Einstellungen in der SPS (beispielhaft für Rockwell CompactLogix™) beschrieben.

#### Rockwell Studio 5000

Diese Beschreibung zeigt alle SPS Screenshots (Studio 5000, Version 30 unter Windows 7) in englischer Sprache. Ggf. Rockwell Software auf Englisch umstellen.

1. Neues Projekt anlegen: "Create" / "New Project".

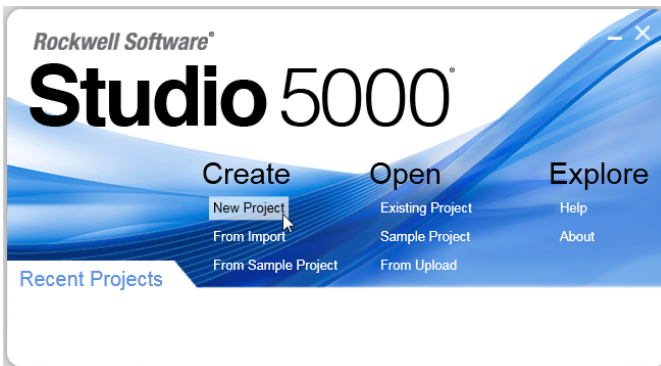


Abb. 43: EtherNet/IP Neues Projekt anlegen

2. Passenden Steuerungstyp auswählen und einen Namen vergeben.

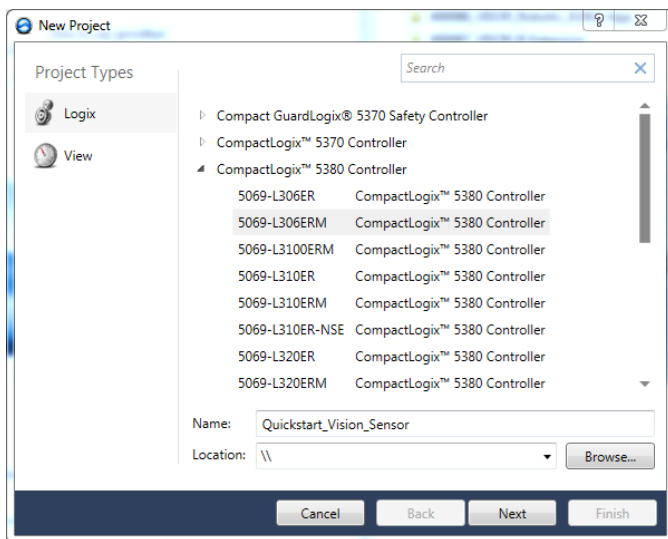


Abb. 44: EtherNet/IP Steuerungstyp auswählen.

3. Standardeinstellungen übernehmen. Mit "Finish" Projekt anlegen.

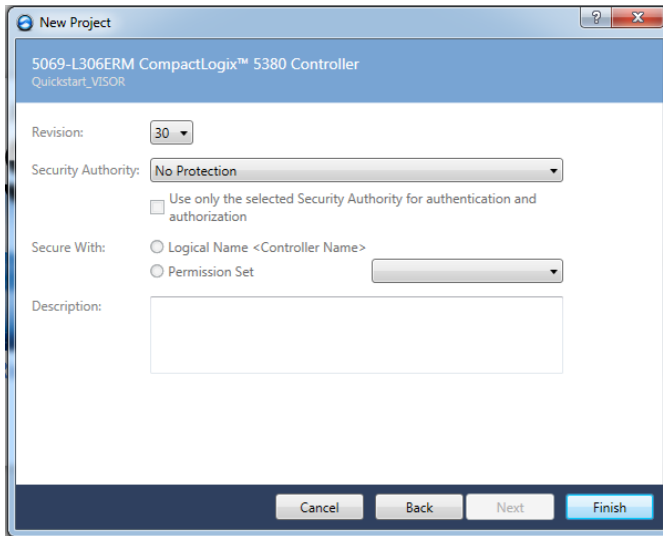


Abb. 45: EtherNet/IP Standardeinstellungen übernehmen.

## 10.2 EDS-Datei installieren

Die Projektansicht öffnet sich. Um die EtherNet/IP-Funktionen des SBS Vision-Sensors nutzen zu können, muss die entsprechende SBS-EDS-Datei in der jeweils aktuellen Version installiert werden.

Falls Steuerung EDS-Datei nicht unterstützt, Anweisungen in Kapitel [Modul anlegen](#) / [Verwendung eines Generic Device \(ohne EDS-Datei\)](#) befolgen.

1. EDS-Datei installieren unter "Tools" / "EDS Hardware Installation Tool".

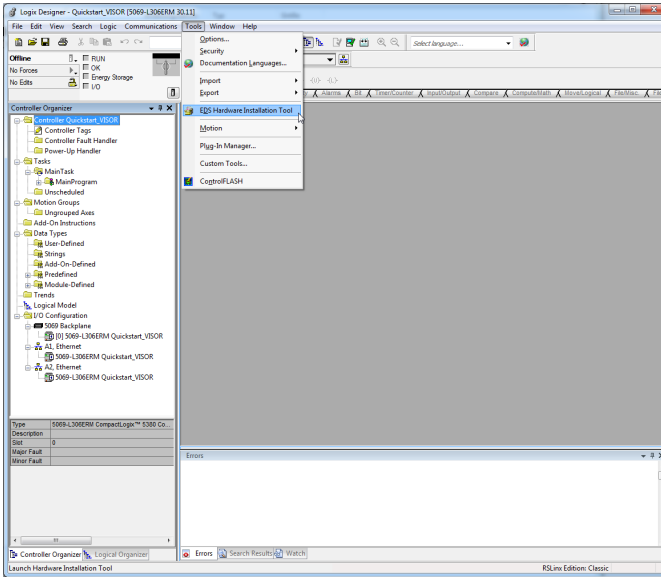


Abb. 46: Projektansicht, Tool, EDS Hardware Installation Tool



2. Informationen mit "Next" bestätigen.



Abb. 47: Informationen bestätigen

3. In den Optionen "Register an EDS File(s)" auswählen.

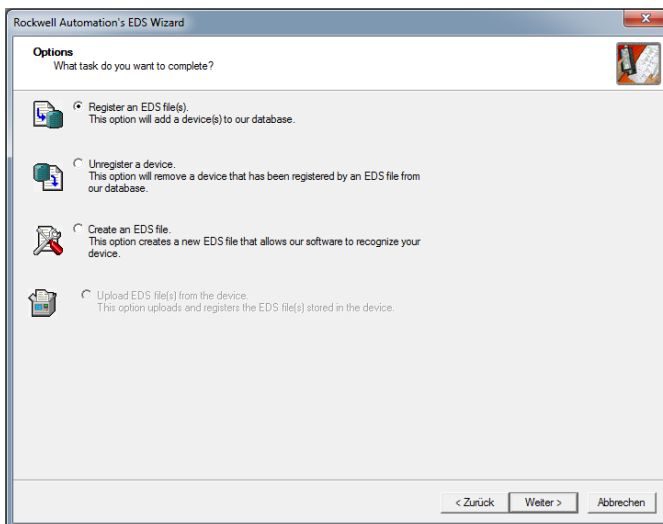


Abb. 48: Register an EDS File(s)

4. "Register a single file" auswählen.



**HINWEIS:**

Für alle SBS Vision-Sensoren kann die identische EDS-Datei verwendet werden.

5. Pfad zur EDS-Datei angeben.

Die EDS-Datei ist im Installationspfad des SBS unter: \Festo\SBS Vision-Sensor\Tools\EtherNet/IP zu finden, und steht auch zum Download unter [www.festo.com](http://www.festo.com) bereit.

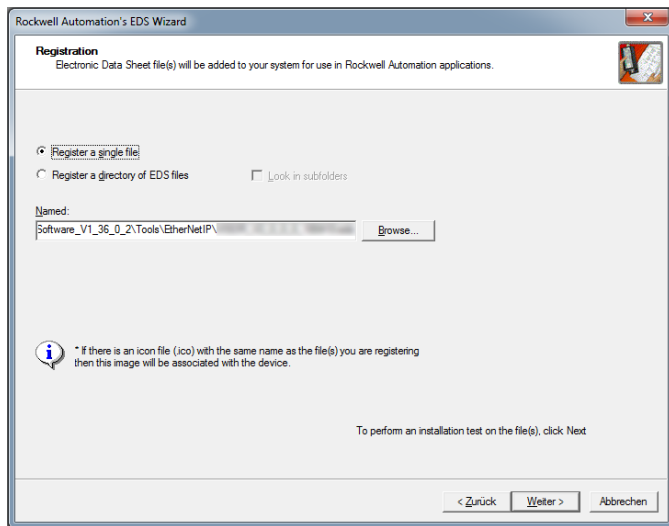


Abb. 49: EDS-Datei auswählen

6. EDS-Datei Test bestätigen.

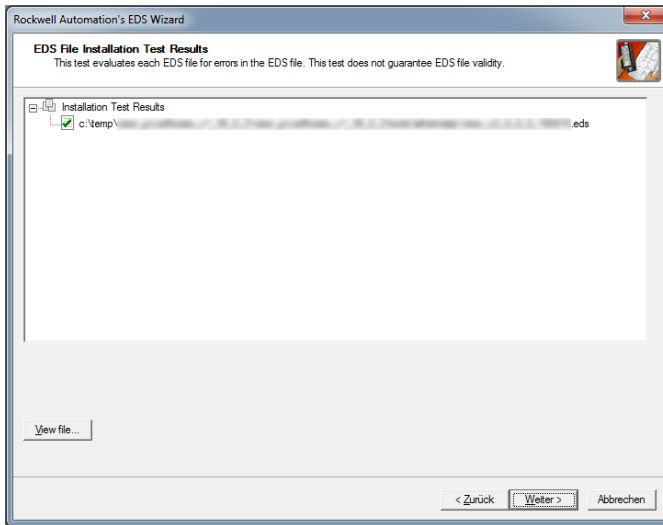


Abb. 50: EDS-Datei Test

7. Bei Bedarf Icon auswählen oder mit Standard-Icon fortfahren.

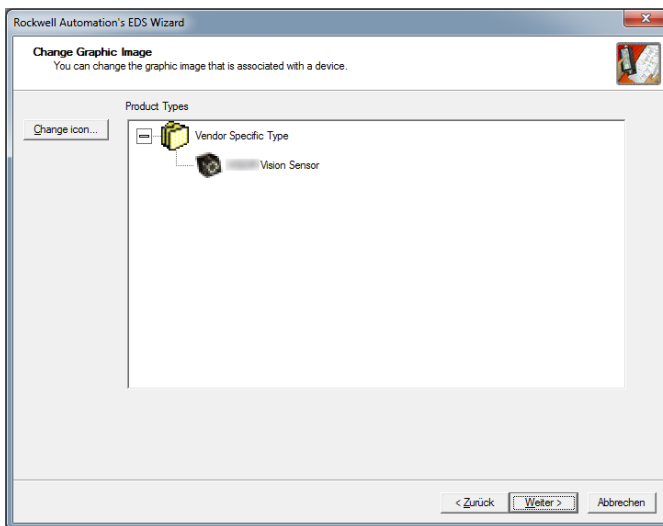


Abb. 51: Icon

## 8. Die Installation bestätigen.

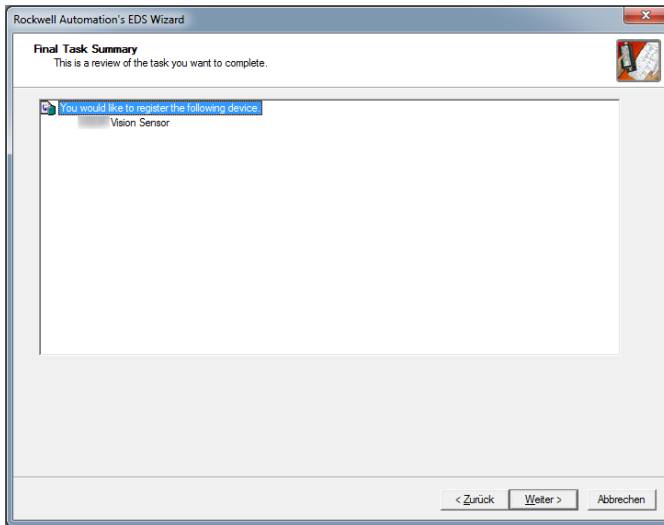


Abb. 52: Installation bestätigen

## 9. Installation mit "Fertig stellen" abschließen.

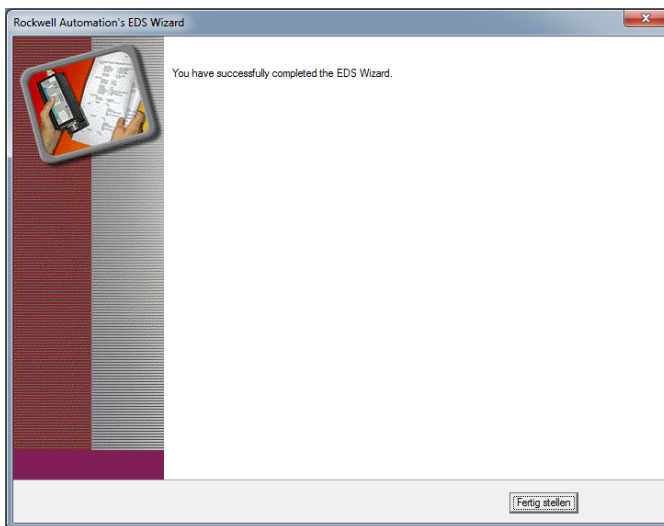


Abb. 53: Installation abschließen

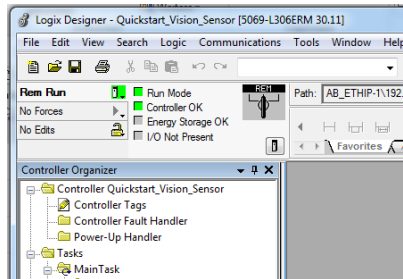
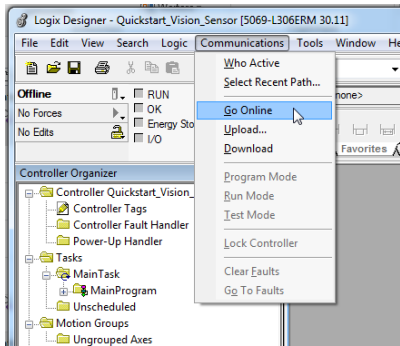
## 10.3 Modul anlegen

### 10.3.1 Auswahl über Hardware-Katalog (mit EDS-Datei)

1. Um mit dem Projekt Online zu gehen, Communications / "Go Online" wählen.

**HINWEIS:**

Der Projektpfad muss zuvor korrekt gesetzt worden sein.



2. Mit Rechtsklick auf die gewünschte Netzwerkverbindung ein neues Modul anlegen.

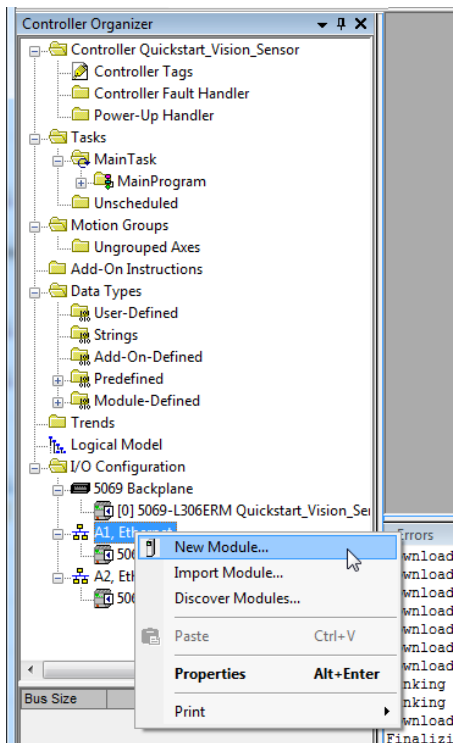


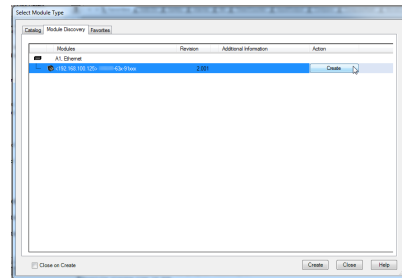
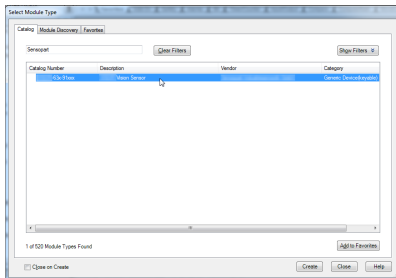
Abb. 54: Neues Modul anlegen

3. SBS aus dem Katalog auswählen oder Online nach verfügbaren Geräten suchen.



**HINWEIS:**

Für die Option "Online suchen" muss die Software bereits online sein (siehe Modul anlegen / Schritt 1).



Im Hardware-Katalog kann nach "Festo" gesucht werden. Die entsprechenden Geräte werden aufgelistet. Alternativ kann über den Reiter "Module Discovery" nach erreichbaren Teilnehmern gesucht werden.

4. Gerätenamen und IP Adresse des SBS vergeben.

- Der Gerätename wird später als Variablenname für die Daten verwendet.
- Die IP Adresse kann über Vision Sensor Device Manager ausgelesen werden.

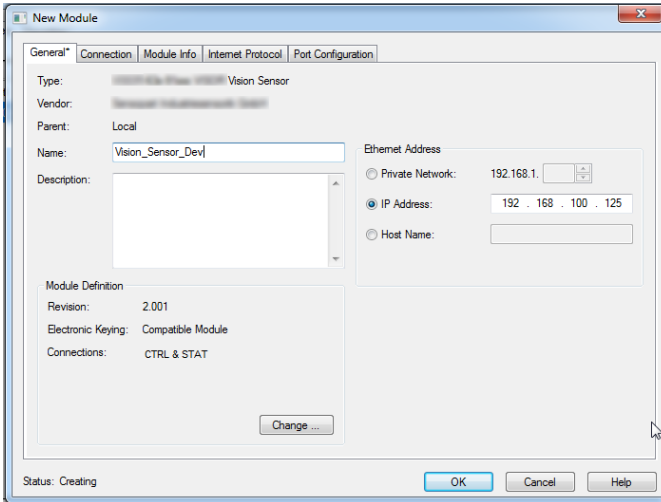
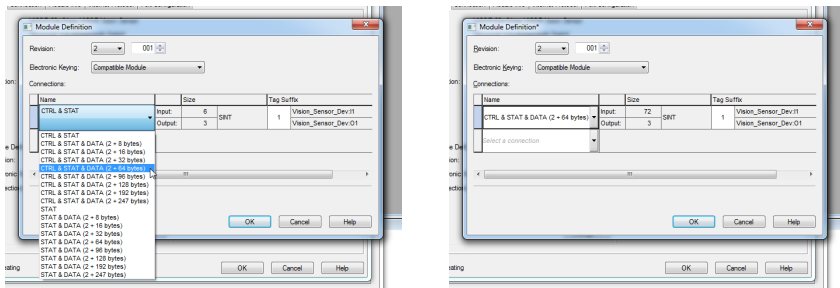


Abb. 55: Geräte- und IP-Adresse setzen

5. Über "Change ..." können die gewünschten Module und Modulgrößen ausgewählt werden.



6. Im Reiter "Connection" die gewünschte Aktualisierungsrate (RP) einstellen.



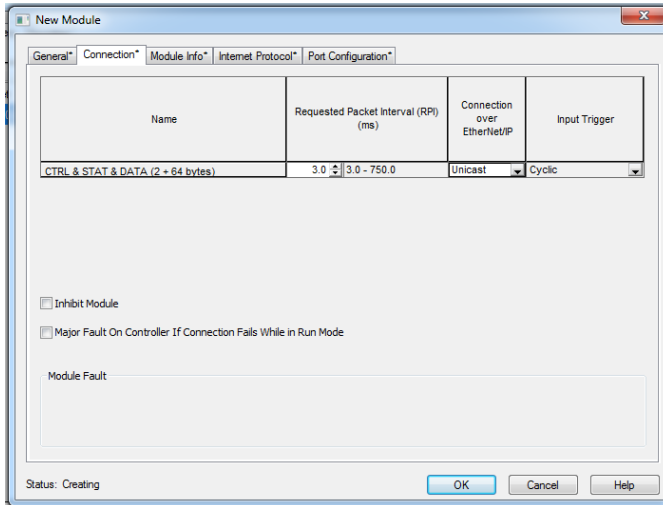


Abb. 56: Aktualisierungsrate einstellen.

7. Über "OK" die Einstellungen des Teilnehmers abschließen.

### 10.3.2 Verwendung eines Generic Device (ohne EDS-Datei)

Falls die Steuerung EDS-Dateien nicht unterstützt, mit folgenden Schritten fortfahren.

1. Mit Rechtsklick auf die gewünschte Netzwerkverbindung ein neues Modul anlegen.

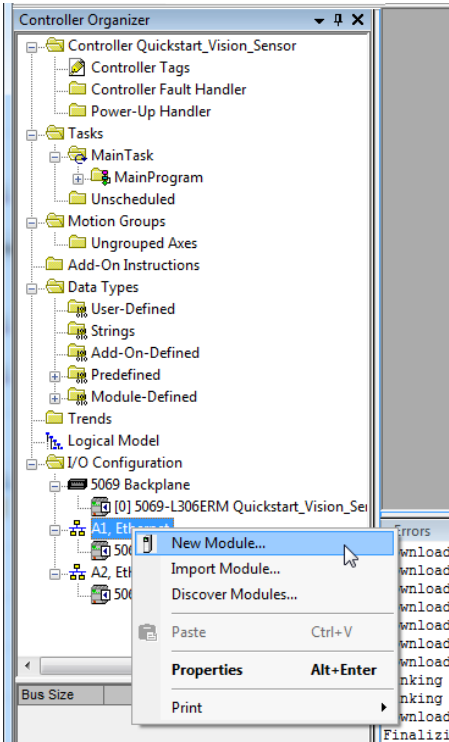


Abb. 57: Neues Modul anlegen

2. Aus dem Katalog ein Modul vom Typ Ethernet-Module - "Generic Ethernet Module" auswählen.

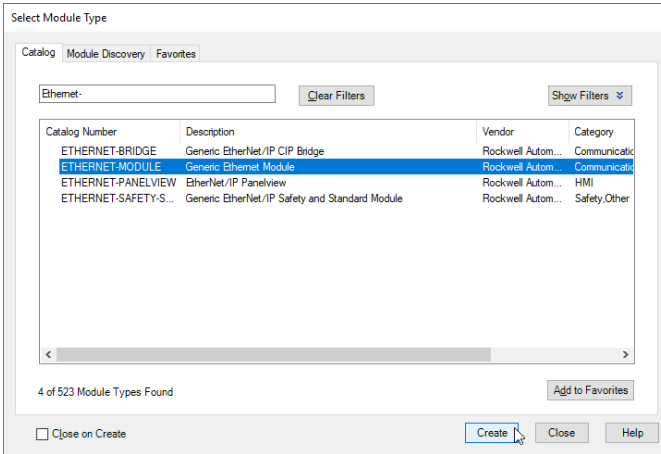


Abb. 58: Auswahl "Generic Ethernet Module"

3. Gerätenamen und IP Adresse des SBS vergeben (A).

- Der Gerätename wird später als Variablenname für die Daten verwendet.
- Die IP Adresse kann über Vision Sensor Device Manager ausgelesen werden.

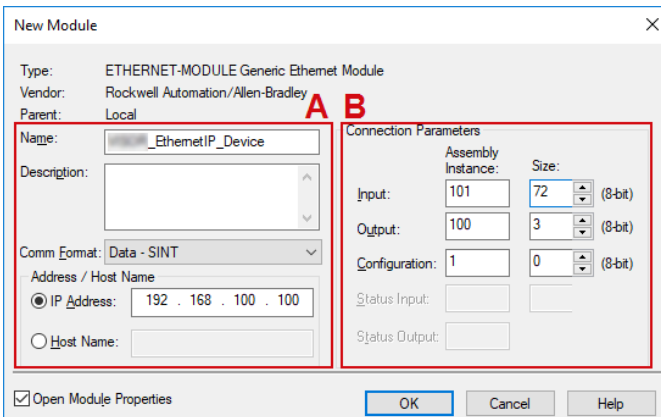


Abb. 59: Vergabe des Gerätenamens und der IP-Adresse

4. Datenformat mit Parameter "Comm Format" auf "Data - SINT" (8 Bit Format) umstellen (A).

5. Verbindungsparameter eintragen (B) (siehe folgende Tabelle).

	Assembly instance (dec)	Size (dec)	Assembly instance (hex)	Size (hex)
<b>Control + Status</b>				
Input	101	6	0x65	0x06
Output	100	3	0x64	0x03
Configuration	1	0	0x01	0x00
<b>Control + Status + Data (2+8)</b>				
Input	102	16	0x66	0x10
Output	100	3	0x64	0x03
Configuration	1	0	0x01	0x00
<b>Control + Status + Data (2+16)</b>				
Input	103	24	0x67	0x18
Output	100	3	0x64	0x03
Configuration	1	0	0x01	0x00
<b>Control + Status + Data (2+32)</b>				
Input	104	40	0x68	0x28
Output	100	3	0x64	0x03
Configuration	1	0	0x01	0x00
<b>Control + Status + Data (2+64)</b>				
Input	105	72	0x69	0x48
Output	100	3	0x64	0x03
Configuration	1	0	0x01	0x00
<b>Control + Status + Data (2+96)</b>				
Input	105	104	0x69	0x68
Output	100	3	0x64	0x03
Configuration	1	0	0x01	0x00
<b>Control + Status + Data (2+128)</b>				
Input	105	136	0x69	0x88
Output	100	3	0x64	0x03
Configuration	1	0	0x01	0x00
<b>Control + Status + Data (2+192)</b>				
Input	105	200	0x69	0xCB
Output	100	3	0x64	0x03

	Assembly instance (dec)	Size (dec)	Assembly instance (hex)	Size (hex)
Configuration	1	0	0x01	0x00
<b>Control + Status + Data (2+247)</b>				
Input	105	255	0x69	0xFF
Output	100	3	0x64	0x03
Configuration	1	0	0x01	0x00

## 10.4 Projekt in Steuerung laden

1. Das Projekt über "Communications" / "Download" auf die Steuerung herunterladen .



### HINWEIS:

Hierfür muss die Software bereits online sein (siehe Modul anlegen / Schritt 1).

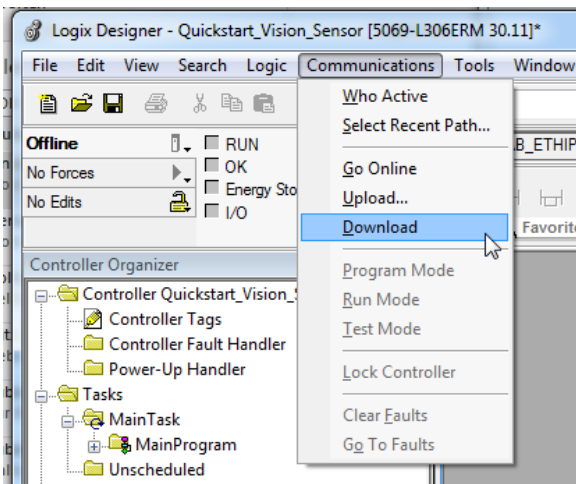


Abb. 60: Download

2. Die Hinweise prüfen und mit "Download" bestätigen.

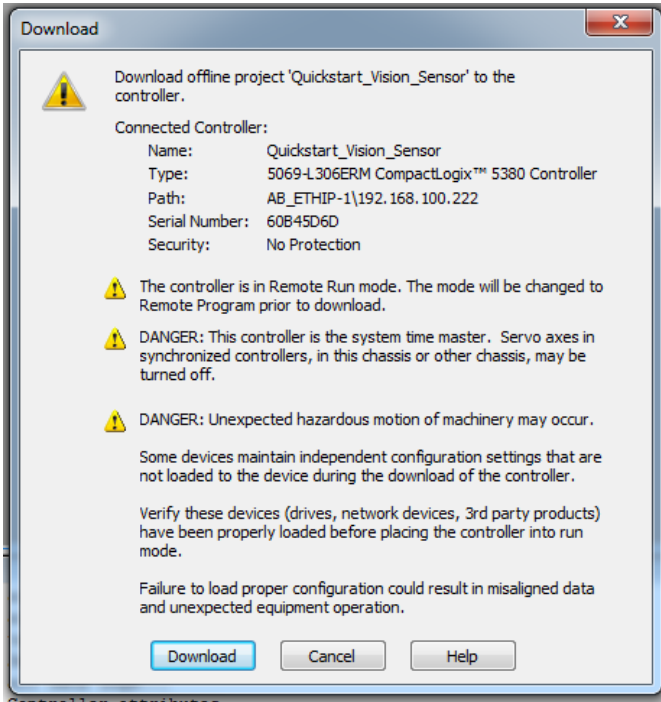


Abb. 61: Hinweise

3. Nach erfolgreichem Download ist der SBS Vision-Sensor im Status "Running".

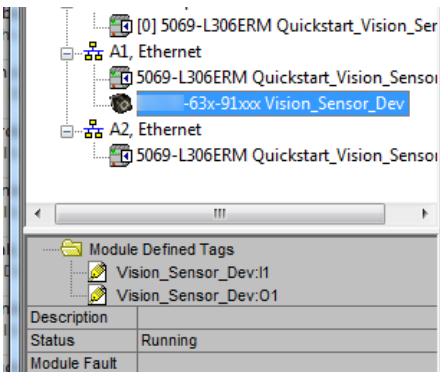


Abb. 62: Status "Running"

## 10.5 Zuordnung der Ausgangsdaten

Die Inputdaten werden wie folgt zugewiesen: (Auswahl Modul CNTL + STAT + Data (2+128))

.... I1.Data[0] – .... I1.Data [5] Modul "Status" (siehe Beschreibung [Modul 2: „Status“ \(Von SBS an Steuerung\)](#))

z.B. .... I1.Data [3] = Jobnumber

.... I1.Data[4] = Image\_ID

Das Datamodul wird direkt angefügt. Start Data Modul ab .... I1.Data[6] - .... I1.Data[135]

Hier werden die Daten eingefügt wie in Vision Sensor Configuration Studio unter "Ausgabe" / "Datenausgabe" angegeben.

Weitere Informationen: [Telegramme / Datenausgabe definieren in Vision Sensor Configuration Studio](#)

Name	Value	Force Mask	Style	De
- Vision_Sensor_Dev:1.Data	{...}	{...}	Decimal	SII
+ Vision_Sensor_Dev:1.Data[0]	1		Decimal	SII
+ Vision_Sensor_Dev:1.Data[1]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[2]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[3]	1		Decimal	SII
+ Vision_Sensor_Dev:1.Data[4]	6		Decimal	SII
+ Vision_Sensor_Dev:1.Data[5]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[6]	6		Decimal	SII
+ Vision_Sensor_Dev:1.Data[7]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[8]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[9]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[10]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[11]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[12]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[13]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[14]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[15]	0		Decimal	SII
+ Vision_Sensor_Dev:1.Data[16]	0		Decimal	SII

Abb. 63: Ausgangsdaten

### Umrechnung von Binär- Werten

Alle detektorspezifischen Nutzdaten mit Nachkommastellen werden als ganze Zahlen, mit 1000 multipliziert, übertragen, und müssen nach Datenempfang deshalb wieder durch 1000 geteilt werden. Die Werte werden im Format „Big Endian“ übertragen. Die Länge richtet sich nach dem Wert, z.B. Score 32 Bit (DWord).

The screenshot displays the Logix Designer interface for a vision sensor application. The main workspace shows a ladder logic program with three rungs. Each rung contains a 'Copy File' block followed by a 'Swap Byte' block. The 'Copy File' blocks copy data from 'Vision\_Sensor\_Dev:1:Data[9]', 'Vision\_Sensor\_Dev:1:Data[13]', and 'Vision\_Sensor\_Dev:1:Data[17]' to temporary variables 'Pos\_X\_tmp', 'Pos\_Y\_tmp', and 'Angle\_tmp'. The 'Swap Byte' blocks then swap the byte order of these variables, with 'Order Mode' set to 'REVERSE'. The 'Watch' window at the bottom shows the current values: Angle is -116, Pos\_X is 409395, and Pos\_Y is 422919.

Name	Scope	Value	Force Mask	Description
Angle	MainProgram	-116		
Pos_X	MainProgram	409395		
Pos_Y	MainProgram	422919		
Vision_Sensor...	Controller	[...]	[...]	
Vision_Sensor...	Controller	[...]	[...]	

Abb. 64: Tauschen der Byte-Reihenfolge

The Results window displays the following data:

Detector	Score	Time	Detector t
1	99.8	27ms	Contour

Statistics:

Count	Reset
1	
Pass	100.00%
Fail	0.00%
Minimum execution time	43ms
Maximum execution time	43ms
Average execution time	43ms



## 11 Telegramme und Datenausgabe

11.1 Übersicht Telegramme .....	81
11.2 Telegramme: Verfügbarkeit und unterstützte Schnittstellen .....	86
11.3 Fehlercodes .....	89
11.4 Beschreibung Telegramme ASCII .....	92
11.5 Beschreibung Telegramme BINÄR .....	192

In diesem Kapitel werden die Telegramme beschrieben, welche für den SBS Vision-Sensor zur Verfügung stehen. Die Telegramme können über verschiedene Schnittstellen an den SBS Vision-Sensor gesendet werden.

- EtherNet/IP
- TCP/IP
- PROFINET (Request / Response Modul)

Die Telegramme stehen in ASCII sowie Binär Format zur Verfügung. Das Format wird im Modul "Vision Sensor Configuration Studio" im Reiter "Datenausgabe" des Bedienschritts "Ausgabe" festgelegt.

Folgende Einstellungen sind möglich:

Kommunikation	TCP / IP	EtherNet/IP	PROFINET
Telegramm Format	ASCII / Binär	Binär	Binär

### 11.1 Übersicht Telegramme

#### [Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

##### SBS Allgemein

- **Statistik zurücksetzen (RST)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Statistik zurücksetzen" kann der interne Statistikzähler des SBS Vision-Sensors zurückgesetzt werden.
- **Bild aus Datei verarbeiten (PIF)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Bild aus Datei verarbeiten" kann der SBS Vision-Sensor für Simulations- und Testzwecke eine Bilddatei anstelle eines Live-Bildes verarbeiten.

##### SBS Kontrolle

- **Trigger (TRG)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Trigger" kann eine Bildaufnahme gestartet werden. Einige Befehle

benötigen eine zusätzliche Bildaufnahme. Ergebnisdaten der Auswertung werden über den "Out"-Port ausgegeben.

- **Erweiterter Trigger (TRX)** ([ASCII](#) / [Binär](#))  
Das Telegramm "Erweiterter Trigger" ist eine Erweiterung zum "Trigger"-Telegramm. Neben den Ergebnisdaten besteht zusätzlich die Möglichkeit, eine ID zu vergeben oder Informationen über den Betriebszustand (Run/Config) zu erhalten. Im Unterschied zum "Trigger"-Telegramm werden beim "Erweiterten Trigger" die Ergebnisdaten auch über den "In"-Port gesendet.
- **Trigger Robotics (TRR)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Trigger Robotics" kann eine Bildaufnahme gestartet werden. Zusätzlich zur Bildaufnahme kann die Roboterwerkzeug-Position (TCP) übergeben werden. Diese wird verwendet, um die Positionswerte zu berechnen.
- **Trigger ID setzen (STI)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Trigger ID setzen" kann ein Trigger Identifier gesetzt werden. Der Identifier wird für die nächste Bildaufnahme verwendet und kann bspw. als Dateiname gesetzt werden.
- **Jobwechsel (CJB)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Jobwechsel" wird ein Jobwechsel auf dem SBS Vision-Sensor angestoßen.
- **Jobwechsel Permanent (CJP)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Jobwechsel Permanent" wird ein permanenter Jobwechsel auf dem SBS Vision-Sensor angestoßen. Der Job wird nach dem Neustart wieder ausgeführt.
- **Jobwechsel nach Jobname (CJN)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Jobwechsel nach Jobname" wird ein Jobwechsel auf dem SBS Vision-Sensor angestoßen. Der Job wird nach dem Jobnamen ausgeführt. Die Jobnamen können z.B. mit dem Telegramm "Jobliste lesen" ausgelesen werden.

## SBS Jobeinstellungen

- **Auto Arbeitsabstand (AFC)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Auto Arbeitsabstand" kann der Arbeitsabstand des Jobs automatisch bestimmt werden.
- **Arbeitsabstand setzen (SFC)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Arbeitsabstand setzen" kann der Arbeitsabstand des Jobs geändert werden.
- **Arbeitsabstand lesen (GFC)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Arbeitsabstand lesen" kann der aktuelle Arbeitsabstand des Jobs gelesen werden.

- **Auto Verschlusszeit (ASH)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Auto Verschlusszeit" kann die Verschlusszeit des Jobs automatisch bestimmt werden.
- **Verschlusszeit setzen (SSP/SST)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Verschlusszeit setzen" kann die Verschlusszeit des Jobs geändert werden. Dieses Telegramm kann z.B. für einen Helligkeitsausgleich verwendet werden.
- **Verschlusszeit lesen (GSH)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Verschlusszeit lesen" kann die gesetzte Verschlusszeit des Jobs gelesen werden.
- **Verstärkung setzen (SGA)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Verstärkung setzen" kann die Verstärkung des Jobs geändert werden. Dieses Telegramm kann z.B. für einen Helligkeitsausgleich verwendet werden.
- **Verstärkungswert lesen (GGA)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Verstärkung lesen" kann die gesetzte Verstärkung des Jobs gelesen werden.
- **Parameter setzen (SPP/SPT)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Parameter setzen" können Parameter von Detektoren angepasst werden, z.B. Referenzstrings, Schwellenwerte für Detektoren.
- **Parameter lesen (GPA)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Parameter lesen" können die gesetzten Parameter der Detektoren gelesen werden.
- **Suchbereich / ROI setzen (SRP/SRT)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "ROI setzen" kann die Position des angegeben Detektors geändert werden.
- **Suchbereich / ROI lesen (GRI)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "ROI lesen" kann die Position des angegeben Detektors ausgelesen werden.
- **Jobliste lesen (GJL)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Jobliste lesen" kann eine Auflistung aller verfügbaren Jobs auf dem SBS Vision-Sensor ausgegeben werden.
- **Detektorliste lesen (GDL)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Detektorliste lesen" kann eine Auflistung aller Detektoren im aktuellen Job ausgegeben werden.
- **Detektor einlernen (TED)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Detektor einlernen" wird der angegeben Detektor neu eingelernt (nur für Mustervergleich, Kontur und Barcode verfügbar).
- **Trigger-Verzögerung setzen (STD)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Trigge-Verzögerung setzen" kann eine Verzögerung für das Auslösen eines Triggers eingestellt werden (in Zeit (ms) oder Encoder-Schritten).

- **Trigger-Verzögerung lesen (GTD)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Trigger-Verzögerung lesen" kann die eingestellte Verzögerung für das Auslösen eines Triggers ausgelesen werden.
- **Speichern Job Permanent (SJP)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Speichern Job Permanent" werden alle zuvor temporär gesetzten Parameter in einen Jobsatz übernommen.

## SBS Kalibrierung

- **Kalibrierung: Initialisieren (CCD)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Punktpaarliste initialisieren" wird die Punktpaarliste initialisiert.
- **Kalibrierung: Weltpunkt hinzufügen (CAW)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Weltpunkt hinzufügen" wird ein Weltpunkt (Referenzmarke oder Punktpaar) der Punktpaarliste hinzugefügt. Das Telegramm kann für Kalibriermethode Punktpaarliste (Roboter) und Kalibrierplatte (Roboter) verwendet werden.
- **Kalibrierung: Punktpaarliste (CCL)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Punktpaarliste" wird die Kalibrierung anhand der Punktpaarliste im aktuellen Job durchgeführt.
- **Kalibrierung: Punktpaarliste validieren (CVL)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Punktpaarliste validieren" wird die Kalibrierung anhand der Punktpaarliste validiert.
- **Kalibrierung: Kalibrierplatte (CCP)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Kalibrierplatte" wird die Kalibrierung anhand der Kalibrierplatte durchgeführt.
- **Referenzmarke setzen (CSF)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Referenzmarke setzen" werden die Referenzmarken anhand der Punktpaarliste im aktuellen Job gesetzt.
- **Kalibrierung: Bild hinzufügen (CAI)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Bild hinzufügen" wird eine Bildaufnahme ausgelöst und wenn eine Kalibrierplatte gefunden wird, wird dem Kalibrierobjekt eine Aufnahme hinzugefügt. Das Telegramm kann für Kalibriermethode Mehrbild-Kalibrierung und Kalibriermethode Kalibrierplatte (Roboter) verwendet werden.
- **Kalibrierung: Mehrbild (CMP)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Mehrbild" wird eine Kalibrierung durchgeführt und dabei auf ein vorhandenes Kalibrierobjekt zugegriffen.
- **Kalibrierung: Roboter Mehrbild (CRP)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Roboter Mehrbild" wird eine Kalibrierung anhand der Kalibrierplatte durchgeführt.

- **Kalibrierung: Kalibrierung kopieren (CCC)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Kalibrierung kopieren" wird die Kalibrierung aus dem aktuellen Job auf das angegebene Ziel kopiert.
- **Kalibrierung: Parameter setzen (CSP)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Parameter setzen" können Parameterwerte für die Kalibrierung gesetzt werden.
- **Kalibrierung: Parameter lesen (CGP)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Kalibrierung: Parameter lesen" können eingestellte Parameterwerte der Kalibrierung gelesen werden.

## SBS Visualisierung

- **Bild holen (GIM)** ([ASCII](#) / [Binär](#))  
Mit dem Telegramm "Bild holen" kann das Bild vom SBS Vision-Sensor erhalten werden.

## SBS Service (nur auf Port 1998 und in ASCII Format verfügbar)

- **Update Visualisierungsdaten (UVR)** ([ASCII](#))  
Mit dem Telegramm "Visualisierungsdaten aktualisieren" werden Visualisierungsdaten, wie z.B. Bild, Detektor Informationen und Ergebnisse aktualisiert.
- **Sensoridentität lesen (GSI)** ([ASCII](#))  
Mit dem Telegramm "Sensoridentität lesen" kann der aktuelle Firmwarestand sowie der Hardwaretyp abgefragt werden.
- **Firmware aktualisieren (UFW)** ([ASCII](#))  
Mit dem Telegramm "Firmware aktualisieren" wird ein Firmwareupdate gestartet. Die Firmware-Datei muss zuvor auf den SBS Vision-Sensor geladen werden.
- **Jobsatz einlesen (SJS)** ([ASCII](#))  
Mit dem Telegramm "Jobsatz einlesen" kann der Jobsatz des SBS Vision-Sensors getauscht werden. Die Jobsatz-Datei muss zuvor auf den SBS geladen werden.
- **Jobsatz sichern (GJS)** ([ASCII](#))  
Mit dem Telegramm "Jobsatz sichern" kann der Jobsatz des SBS Vision-Sensors gelesen werden.

## Datenausgabe

In diesem Abschnitt finden sich Informationen über die Datenausgabe (z. B. welches Format die einzelnen Ergebnisse erhalten).

## Datenausgabe ASCII

- [Allgemein](#)
- [Basiswerte](#)
- [Position / Lage](#)
- [Messen](#)
- [Identifikation](#)
- [Identifikation - Qualität](#)
- [Farbe](#)
- [Zählen / Anzahl](#)
- [Erweitert](#)

## Datenausgabe Binär

- [Allgemein](#)
- [Basiswerte](#)
- [Position / Lage](#)
- [Messen](#)
- [Identifikation](#)
- [Identifikation - Qualität](#)
- [Farbe](#)
- [Zählen / Anzahl](#)
- [Erweitert](#)



### HINWEIS:

Für die Übertragung von Dateien vom bzw. zum SBS Vision-Sensor wird das Verzeichnis **/tmp** auf dem SBS verwendet. Dateien können von hier / hierhin mit einem SFTP-Client übertragen werden. Zugangsdaten für SFTP-Client: Benutzername: *user*, Passwort: *user*.

## 11.2 Telegramme: Verfügbarkeit und unterstützte Schnittstellen

### Gerätevariante

<b>U</b>	Universal
<b>Q</b>	Object
<b>B</b>	Code Reader
<b>R</b>	Robotic

✓ verfügbar

[ ] Eingeschränkt verfügbar: Unterschiede zwischen Versionen < 2 und ≥ 2

### Gerätetyp

	Standard
<b>AF</b>	Erweitert
<b>PF</b>	Professional

### Schnittstellen

<b>1</b>	Ethernet TCP IN (2006)
<b>2</b>	PROFINET
<b>3</b>	EtherNet/IP
<b>4</b>	Service Port (1998)

Telegramm	U		Q		B		R		Schnittstellen				Ab Version	
	AF	PF		AF		AF	PF	AF	PF	1	2	3		4
<b>SBS Allgemein</b>														
Statistik zurücksetzen (RST)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.18
Bild aus Datei verarbeiten (PIF)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	2.0

Telegramm	U		Q		B		R		Schnittstellen				Ab Ver- sion	
	AF	PF		AF		AF	PF	AF	PF	1	2	3		4
<b>SBS Kontrolle</b>														
Trigger (TRG)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
Erweiterter Trigger (TRX)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			1.6
Trigger Robotics (TRR)		✓						✓	✓	✓	✓	✓		2.2
Trigger ID setzen (STI)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		2.2
Jobwechsel (CJB)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
Jobwechsel Per- manent (CJP)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.18
Jobwechsel nach Jobname (CJN)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		2.0
<b>SBS Jobeinstellungen</b>														
Auto Arbeitsabstand (AFC)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		2.0
Arbeitsabstand setzen (SFC)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		2.0
Arbeitsabstand lesen (GFC)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		2.0
Auto Verschlusszeit (ASH)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		2.0
Verschlusszeit set- zen (SSP/SST)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
Verschlusszeit lesen (GSH)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
Verstärkung setzen (SGA)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.6
Verstärkungswert lesen (GGA)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.6

Telegramm	U		Q		B		R		Schnittstellen				Ab Ver- sion	
	AF	PF		AF		AF	PF	AF	PF	1	2	3		4
Parameter setzen (SPP/SPT)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
Parameter lesen (GPA)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
ROI setzen (SRP/SRT)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
ROI lesen (GRI)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
Jobliste lesen (GJL)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.18
Detektorliste lesen (GDL)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.18
Detektor einlernen (TED)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.0
Trigger-Verzögerung setzen (STD)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.22
Trigger-Verzögerung lesen (GTD)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		1.22
Job Permanent Speichern (SJP)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		2.0
<b>SBS Kalibrierung</b>														
Initialisieren (CCD)	✓	✓						✓	✓	✓	✓	✓		1.18
Welpunkt hinzufügen (CAW)	✓	✓						✓	✓	✓	✓	✓		1.22
Kalibrierung Punktpaarliste (CCL)	✓	✓						✓	✓	✓	✓	✓		1.18
Kalibrierung validieren (CVL)	✓	✓						✓	✓	✓	✓	✓		1.18
Kalibrierung Kalibrierplatte (CCP)	[1]	✓		[1]				✓	✓	✓	✓	✓		1.19
Referenzmarken setzen (CSF)		✓						✓	✓	✓	✓	✓		1.22



Telegramm	U		Q		B		R		Schnittstellen				Ab Ver- sion	
	AF	PF		AF		AF	PF	AF	PF	1	2	3		4
Bild hinzufügen (CAI)	✓	✓						✓	✓	✓	✓	✓		2.2
Mehrbild (CMP)	✓	✓						✓	✓	✓	✓	✓		2.2
Roboter Mehrbild (CRP)		✓							✓	✓	✓	✓		2.2
Kalibrierung kopieren (CCC)		✓						✓	✓	✓	✓	✓		1.19
Parameter setzen (CSP)	[ ]	✓		[ ]				✓	✓	✓	✓	✓		1.22
Parameter lesen (CGP)	[ ]	✓		[ ]				✓	✓	✓	✓	✓		1.22
<b>SBS Visualisierung</b>														
Bild holen (GIM)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	1.0
<b>SBS Service</b>														
Update Visualisierungsdaten (UVR)	✓	✓		✓		✓	✓	✓	✓				✓	1.22
Sensoridentität lesen (GSI)	✓	✓		✓		✓	✓	✓	✓				✓	1.19
Firmware aktualisieren (UFW)	✓	✓		✓		✓	✓	✓	✓				✓	1.19
Jobsatz einlesen (SJS)	✓	✓		✓		✓	✓	✓	✓				✓	1.19
Jobsatz sichern (GJS)	✓	✓		✓		✓	✓	✓	✓				✓	1.19

Siehe auch: [Übersicht Telegramme](#)

### 11.3 Fehlercodes

Fehlercode	Fehlercode HEX	Beschreibung
000	0x00	Erfolgreich
001	0x01	Fehler
003	0x03	Ungültige Parameterdaten

<b>Fehlercode</b>	<b>Fehlercode HEX</b>	<b>Beschreibung</b>
005	0x05	Ungültiges Telegramm
006	0x06	Eingabeparameter mit ungültiger Größe oder ungültigem Wert
007	0x07	Die Datei existiert nicht
008	0x08	Rekorder ausgeschaltet
009	0x09	Das passende Bild des angeforderten Typs wurde nicht gefunden
010	0x0A	Ungültiger Dateiname oder Länge
011	0x0B	Ungültige Datenlänge
012	0x0C	Nicht zulässig aufgrund von nicht übereinstimmendem Jobsatz
013	0x0D	Fehler beim Starten eines neuen Jobs aus dem Jobsatz
016	0x10	Nicht übereinstimmende Firmware-Version
018	0x12	Kalibrierplattendatei nicht vorhanden
020	0x14	Mehr als eine vis-Datei vorhanden
021	0x15	Sensortyp unpassend für vis-Datei
029	0x1D	Temporäre Jobänderung abgelehnt, da Job-Prüfsumme aktiv ist.
030	0x1E	Kalibrierung nicht aktiviert / Kalibrierung nicht unterstützt
031	0x1F	Fehler beim Kopieren
032	0x20	Nicht übereinstimmende Eingangsbedingung für Ziel-Job
033	0x21	Kalibrierungs-/ Validierungsfehler
034	0x22	Ungültige Anzahl an Punkten
035	0x23	Fehler Kalibrierung: Punkt hinzufügen (z.B. letztes Jobergebnis fehlgeschlagen)
036	0x24	Ungültige Referenzmarke
037	0x25	Jobsatz geschützt: Permanente Änderungen an Job nicht zulässig
038	0x26	Parameter-Werte sind nicht verfügbar zum Schreiben / Lesen
039	0x27	Sensor ist im Konfigurations-Modus, Telegramm wurde abgelehnt
040	0x28	Fehler beim Schreiben / Lesen von Parameter-Wert
041	0x29	Kein übereinstimmender Job gefunden
042	0x2A	Formatfehler
043	0x2B	Jobset- / Job-Speicherfehler
044	0x2C	Fokussperrzeit verletzt

<b>Fehlercode</b>	<b>Fehlercode HEX</b>	<b>Beschreibung</b>
045	0x2D	Fehler bei mehreren Dateien
046	0x2E	Arbeitsabstand konnte nicht ermittelt werden
047	0x2F	"Min. Bearbeitungszeit pro Bild" wurde nicht eingehalten
048	0x30	Suchbereichgröße (ROI) stimmt nicht überein
049	0x31	Suchbereich (ROI) Freiform nicht gewählt
050	0x32	Kalibriermethode stimmt nicht überein
051	0x33	Keine Kalibrierplatte gefunden
052	0x34	Bildanzahl zu gering
053	0x35	Keine Kalibrierung möglich: Abstand Werkzeugpositionen nicht plausibel
054	0x36	Rotation zwischen den Bildern nicht ausreichend
055	0x37	Neigung zwischen den Bildern nicht ausreichend

## 11.4 Beschreibung Telegramme ASCII

### 11.4.1 Allgemein

#### Statistik zurücksetzen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Statistik zurücksetzen (RST) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>R</b>	Statistik zurücksetzen (Reset Statistics)
2	<b>S</b>	
3	<b>T</b>	
Beispiel:	RST	
Statistik zurücksetzen (RST) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>R</b>	Statistik zurücksetzen (Reset Statistics)
2	<b>S</b>	
3	<b>T</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
Beispiel:	RSTP	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)



#### HINWEIS:

Statistik-Werte können ausgegeben werden im Bedienschnitt Ausgabe / Reiter Datenausgabe / "Detektorspezifische Nutzdaten", Auswahl "GENERAL".

Weitere Informationen: siehe Datenausgabe ASCII / [Allgemein](#)

## Bild aus Datei verarbeiten (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Bild aus Datei verarbeiten (PIF) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>P</b>	Bild aus Datei verarbeiten (Process image from file)
2	<b>I</b>	
3	<b>F</b>	
4	X	Version des Request
5	0	Reserviert
6	1	Reserviert
7-9	X	Länge des darauffolgenden Dateinamens der Datei (Tatsächliche Byte-Anzahl mit Datei-erweiterung, max. 255 Bytes)
10 ... n	X	Dateiname (UTF-8) und Format verfügbar auf dem Gerät im Verzeichnis "/tmp/". Erlaubte Erweiterungen: Monochromer Sensor: .pgm Farbsensor: .ppm (RGB) or .pgm (Bayer)
Beispiel:	PIF1 0 1 009 Image.pgm PIF1 0 1 008 test.pgm	
<b>Bild aus Datei verarbeiten (PIF) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>P</b>	Bild aus Datei verarbeiten (Process image from file)
2	<b>I</b>	
3	<b>F</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5-7	X	<a href="#">Fehlercodes</a>
8	0	Reserviert
9-16	X	Länge der impliziten Ergebnisausgabe
17 ... n	X	Ausgabe des impliziten Ergebnis
Beispiel:	PIF P 000 00000010 0123456789 PIF P 000 0 00000014 [15;P;1;53371] PIF P 000 1 00000005 [2;7]	

Weitere Informationen:	
Akzeptiert im Runmode:	Ja
Akzeptiert im Konfigurationsmodus:	Nein
Akzeptiert wenn Ready Low:	Ja
Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:	Max. 4 Byte (optional)



**HINWEIS:**

Bildgröße des Testbilds muss übereinstimmen mit der Bildgröße des derzeit aktiven Jobs auf dem Gerät.

## 11.4.2 Kontrolle

### Trigger (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Trigger (TRG) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>T</b>	Trigger (einfacher Trigger, In-Port)
2	<b>R</b>	
3	<b>G</b>	
Beispiel:	TRG	
Trigger (TRG) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>T</b>	Trigger, (Response auf Kommando Trigger ohne Index, via Port 2006. Falls definiert: Ergebnisdaten ohne Index via Port 2005)
2	<b>R</b>	
3	<b>G</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
Beispiel:	TRGP	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Ja
Akzeptiert wenn Ready Low:		Nein
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Erweiterter Trigger (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

Erweiterter Trigger (TRX) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>T</b>	Erweiterter Trigger, (Trigger mit Index, zur Zuordnung Trigger zu entsprechenden Ergebnisdaten, via Port 2006)
2	<b>R</b>	
3	<b>X</b>	
4 - 5	X	Länge nachfolgender Daten (0-99)
6 ... n	X	Daten
Beispiel:	TRX06MyPart	
Erweiterter Trigger (TRX) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>T</b>	Erweiterter Trigger, (Response auf Trigger mit Index und Ergebnisdaten, via Port 2006, zur Zuordnung von Trigger zum Ergebnis. Ergebnisdaten außerdem ohne Index via Port 2005)
2	<b>R</b>	
3	<b>X</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 6	X	Länge nachfolgender Daten (n)
7 ... n	X	Daten aus dem Sendebefehl
n+1	C R	C = Config R = Run
n+2 ... n+9	X	Länge nachfolgender Ergebnisdaten (m)
n+9 ... m	X	Ergebnisdaten
Beispiel:	TRX06MyPartR00000000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Ja
Akzeptiert wenn Ready Low:		Nein
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)



## Trigger Robotics (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Trigger Robotics (TRR) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>T</b>	Trigger Robotics
2	<b>R</b>	
3	<b>R</b>	
4	1	Version des Request
5-6	X	Länge des Trigger-Identifiers
7 ... n	X	Trigger Identifier
n+1...n+8	X	Pose_TCP Pos. X (in Kundeneinheit * 1000)
n+9...n+16	X	Pose_TCP Pos. Y (in Kundeneinheit * 1000)
n+17...n+24	X	Pose_TCP Pos. Z (in Kundeneinheit * 1000)
n+25...n+32	X	Pose_TCP Winkel X (in Grad * 1000)
n+33...n+40	X	Pose_TCP Winkel Y (in Grad * 1000)
n+41...n+48	X	Pose_TCP Winkel Z (in Grad * 1000)
Beispiel:	TRR104Part000040040000500500006006000070070000 800800009009	
<b>Trigger Robotics (TRR) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>T</b>	Trigger, (Response auf Kommando Trigger ohne Index, via Port 2006. Falls definiert: Ergebnisdaten ohne Index via Port 2005)
2	<b>R</b>	
3	<b>R</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5-7	X	<a href="#">Fehlercodes</a>
7-8	X	Länge des Trigger-Identifiers
9 ... n	X	Trigger Identifier

n+1	X	Operation Mode C = Config R = Run
n+2...n+9	X	Länge der Ergebnisdaten
n+10...m	X	Ergebnisdaten
Beispiel:	TRRP00004PartR00000000	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Ja	
Akzeptiert wenn Ready Low:	Nein	
Status des Signals Ready während Bearbeitung:	Low	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:		

Hinweis: Für "Kalibrierplatte (Roboter)" und "Punktpaarliste (Roboter)" werden nur die X- und Y-Position berücksichtigt. Die anderen Werte (Position Z und Rotationen) müssen 0 sein.

### Trigger ID setzen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Trigger ID setzen (STI) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Trigger ID setzen (Set Trigger ID)
2	<b>T</b>	
3	<b>I</b>	
4	1	Version des Request
5-6	x	Länge der nachfolgenden Daten (max 99)
7 ... n	x	Trigger ID
Beispiel:	STI106MyPart	
Trigger ID setzen (STI) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Trigger ID setzen (Set Trigger ID)
2	<b>T</b>	
3	<b>I</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5-7	x	<a href="#">Fehlercodes</a>
Beispiel:	STIP000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Ja
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		

## Jobwechsel (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

Jobwechsel (CJB) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Jobwechsel (Change Job)
2	<b>J</b>	
3	<b>B</b>	
4 - 6	X	Job Nummer
Beispiel:	CJB005	
Jobwechsel (CJB) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Jobwechsel (Change Job)
2	<b>J</b>	
3	<b>B</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5	T F	Triggered Freerun
6 - 8	X	Job Nummer
Beispiel 1:	CJBPT005	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Low	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	



### HINWEIS:

Tritt ein Fehler beim Jobwechsel auf, besteht die Möglichkeit in Job 1 zu wechseln.

## Jobwechsel Permanent (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Jobwechsel Permanent (CJP) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Jobwechsel Permanent (Change Job Permanently)
2	<b>J</b>	
3	<b>P</b>	
4 - 6	X	Job Nummer
Beispiel:	CJP005	
Jobwechsel Permanent (CJP) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Jobwechsel Permanent (Change Job Permanently)
2	<b>J</b>	
3	<b>P</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5	T F	Triggered Freerun
6 - 8	X	Job Nummer
Beispiel 1:	CJPPT005	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)



### HINWEIS:

Tritt ein Fehler beim Jobwechsel auf, besteht die Möglichkeit in Job 1 zu wechseln.

## Jobwechsel nach Jobname (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)


Jobwechsel nach Jobname (CJN) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Jobwechsel nach Jobname (Change Job by Name)
2	<b>J</b>	
3	<b>N</b>	
4	1	Version des Request
5 - 7	X	Länge Jobname
8 ... n	X	Jobname
Beispiel:	CJN1005Myjob	
Jobwechsel nach Jobname (CJN) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Jobwechsel nach Jobname (Change Job by Name)
2	<b>J</b>	
3	<b>N</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8	X	Triggermodus T: Trigger F: Freilauf
Beispiel:	CJNP000T	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)


### 11.4.3 Jobeinstellungen

#### Auto Arbeitsabstand (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Auto Arbeitsabstand (AFC) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>A</b>	Auto Arbeitsabstand (Auto Focus)
2	<b>F</b>	
3	<b>C</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
6	X	Schrittlänge 1-5
7 - 9	X	Fokusauswahl 0: Höchster Score 1: Min. Arbeitsabstand 2: Max. Arbeitsabstand 3: Mittelwert Arbeitsabstand 4: Median Arbeitsabstand 5: Höchste Score und alle Ebenen
10	X	Fokuseinheit 0: Millimeter 1: Schritte
11	X	Auswahl Arbeitsabstand 0: Standardbereich 1: Spezifizierter Bereich
	 <b>HINWEIS:</b> Folgende Bytefolge ist nur relevant wenn "Auswahl Abstandsbereich" auf 1 gesetzt wurde.	
12 - 19	X	Anfang Arbeitsbereich (nah)
20 - 27	X	Ende Arbeitsbereich (fern)
Beispiel:	Beispiel 1: AFC11100500 Beispiel 2: AFC111005010001000000100000	
Auto Arbeitsabstand (AFC) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung

1	A	Auto Arbeitsabstand (Auto Focus)
2	F	
3	C	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 10	X	Auswahl Fokus = 5 ; Anzahl gefundener Abstände Auswahl Fokus = 1-4 ; 1
	 <b>HINWEIS:</b> Die Folgenden Felder [Abstandswert / Score-Wert] werden pro Anzahl Gefundener Abstände wiederholt.	
11 - 18	X	Abstandswert (in mm *1000) oder in Schritten
19 - 26	X	Score-Wert in %*1000
Beispiel:	AFCP00000200000095000000900009300000089000	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	



## Arbeitsabstand setzen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Arbeitsabstand setzen (SFC) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Arbeitsabstand (Set Focus)
2	<b>F</b>	
3	<b>C</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
6	X	Bewegung 0: Absolut 1: Relativ 2: Absolut mit Neuinitialisierung
7	X	Einheit 0: 1/1000 Millimeter 4: Schritte
8 - 15	X	Abstandswert (in mm *1000) oder in Schritten
Beispiel:	SFC111400000010	
Arbeitsabstand setzen (SFC) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Arbeitsabstand (Set Focus)
2	<b>F</b>	
3	<b>C</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 15	X	Abstandswert (in mm *1000) oder in Schritten
Beispiel:	SFCP00000000050	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja

Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:	Max. 4 Byte (optional)



## Arbeitsabstand lesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Arbeitsabstand lesen (GFC) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Arbeitsabstand lesen (Get Focus)
2	<b>F</b>	
3	<b>C</b>	
4	1	Version des Request
5	X	Einheit 0 - 1/1000 Millimeter 4 - Schritte
Beispiel:	GFC10	
Arbeitsabstand lesen (GFC) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Arbeitsabstand lesen (Get Focus)
2	<b>F</b>	
3	<b>C</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 15	X	Abstandswert (in mm *1000) oder in Schritten
Beispiel:	GFCP00000092500	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Auto Verschlusszeit (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Auto Verschlusszeit (ASH) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>A</b>	Auto Verschlusszeit (Auto Shutter)
2	<b>S</b>	
3	<b>H</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
Beispiel:	ASH11	
Auto Verschlusszeit (ASH) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>A</b>	Auto Verschlusszeit (Auto Shutter)
2	<b>S</b>	
3	<b>H</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 15	X	Auto Verschlusszeit-Wert (in mm *1000)
16 - 23	X	Score in % * 1000
Beispiel:	ASHP0000000178000057500	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Verschlusszeit setzen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Verschlusszeit setzen (SSP/SST) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Verschlusszeit setzen (Set Shutter Speed)
2	<b>S</b>	
3	<b>P</b> <b>T</b>	Permanent Temporär
4 - 5	X	Anzahl der Stellen des Verschlusszeitwerts, z.B. 04
6 - 9	X	Neuer Verschlusszeitwert in ms * 1000 z.B. 8000 = 8 ms
Beispiel:	SSP048000	
Verschlusszeit setzen (SSP/SST) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Verschlusszeit setzen (Set Shutter Speed )
2	<b>S</b>	
3	<b>P</b> <b>T</b>	Permanent Temporär
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
Beispiel:	SSPP	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

### Verschlusszeitwert lesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Verschlusszeitwert lesen (GSH) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Verschlusszeitwert lesen (Get Shutter Speed) (von aktivem Job)
2	<b>S</b>	
3	<b>H</b>	
Beispiel:	GSH	
Verschlusszeitwert lesen (GSH) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Verschlusszeitwert lesen (Get Shutter Speed)
2	<b>S</b>	
3	<b>H</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5	X	Verschlusszeitwert, Länge
6 ... n	X	Verschlusszeitwert in ms * 1000
Beispiel Run Mode:	GSHP41200	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

## Verstärkung setzen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

Verstärkung setzen (SGA) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Verstärkung setzen (Set Gain)
2	<b>G</b>	
3	<b>A</b>	
4	X	0: Temporär 1: Permanent
5 - 9	X	Neuer Verstärkungswert (in Wert *1000), z.B. 2,0 = 02000
Beispiel:	SGA102000	
Verstärkung setzen (SGA) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Verstärkung setzen (Set Gain)
2	<b>G</b>	
3	<b>A</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 9	X	Aktueller Verstärkungswert * 1000
Beispiel:	SGAP02000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)



## Verstärkungswert auslesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Verstärkungswert lesen (GGA) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Verstärkungswert lesen (Get Gain)
2	<b>G</b>	
3	<b>A</b>	
Beispiel:	GGA	
Verstärkungswert lesen (GGA) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Verstärkungswert lesen (Get Gain)
2	<b>G</b>	
3	<b>A</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 9	X	Aktueller Verstärkungswert (in Wert * 1000), z.B. 1,0 = 01000
Beispiel:	GGAP01000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Parameter setzen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Parameter setzen (SPP/SPT) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Parameter setzen (Set Parameter)
2	<b>P</b>	
3	<b>P</b> <b>T</b>	P: Permanent T: Temporär
4 - 6	X	Detektor Nummer
7 - 9	X	Kommando: Parameter-Nummer, siehe unten, Tabelle <a href="#">Übersicht Detektor-Parameter</a>
10 - 14	X	Länge des Werts (max. 512 Byte)
15 ... n	X	Wert
Beispiel:	SPP0010010000560000	
Parameter setzen (SPP/SPT) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Parameter setzen (Set Parameter)
2	<b>P</b>	
3	<b>P</b> <b>T</b>	P: Permanent T: Temporär
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler

5 - 8	X	SI08 - Signed Integer 08 UI08 - Unsigned Integer 08 SI16 - Signed Integer 16 UI16 - Unsigned Integer 16 SI32 - Signed Integer 32 UI32 - Unsigned Integer 32 SI40 - Signed Integer 40 UI40 - Unsigned Integer 40 FLOT - Float DOBL - Double STRG - String BOOL - Boolean SP08 - Special Signed 8 UDEF - Undefined IARR - Integer Array ZERO - Default Zero Parameter
Beispiel:	SPPPSTRG	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Low	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

## Übersicht Detektor-Parameter (setzen / lesen)

Detektor	Funktion	Wert	Multiplikator	Länge
<b>Lagenachführung</b>				
Mustervergleich Konturvergleich	Schwellenwert Min.	1	1000	n
	Schwellenwert Max.	2	1000	n
	Ergebnisoffset 0: "Aus" 1: "Bildebene (in Pixel)" 2: "Positionierung (2D)" 3: "Roboter (3D)"	30	1	n
	Ergebnisoffset Bildebene: Pos. X	31	1000	n
	Ergebnisoffset Bildebene: Pos. Y	32	1000	n
	Ergebnisoffset Bildebene: Winkel	33	1000	n
	Ergebnisoffset Positionierung (2D), Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z	34	1000	48 (6 * 8 Byte je Wert)
	Ergebnisoffset berechnen* mit übertragener Position  <ul style="list-style-type: none"> <li>• Positionierung (2D): Pos. X, Pos. Y, 0, 0, 0, Winkel Z</li> <li>• Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul> *Es muss eine gültige Posi- tion für den Detektor vor- handen sein	35	1000	48 (6 * 8 Byte je Wert)
Kantenantastung	Antaster 1: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	101	1	n
	Antaster 2: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	102	1	n

Detektor	Funktion	Wert	Multiplikator	Länge
	Antaster 3: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	103	1	n
	Antaster 1: Schwellenwert Min.	104	1000	n
	Antaster 2: Schwellenwert Min.	105	1000	n
	Antaster 3: Schwellenwert Min.	106	1000	n
<b>Detektoren</b>				
Mustervergleich Kontur Kontur 3D	Schwellenwert Min.	1	1000	n
	Schwellenwert Max.	2	1000	n
	Ergebnisoffset 0: "Aus" 1: "Bildebene (in Pixel)" 2: "Positionierung (2D)" 3: "Roboter (3D)"	30	1	n
	Ergebnisoffset Bildebene: Pos. X	31	1000	n
	Ergebnisoffset Bildebene: Pos. Y	32	1000	n
	Ergebnisoffset Bildebene: Winkel	33	1000	n
	Ergebnisoffset <ul style="list-style-type: none"> <li>Positionierung (2D): Pos. X, Pos. Y, 0, 0, 0, Winkel Z</li> <li>Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul>	34	1000	48 (6 * 8 Byte je Wert)

Detektor	Funktion	Wert	Multiplikator	Länge
	Ergebnisoffset berechnen* mit übertragener Position <ul style="list-style-type: none"> <li>• Positionierung (2D): Pos. X, Pos. Y, 0, 0, 0, Winkel Z</li> <li>• Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul> *Es muss eine gültige Position für den Detektor vorhanden sein	35	1000	48 (6 * 8 Byte je Wert)
Zielmarke 3D	Ergebnisoffset <ul style="list-style-type: none"> <li>• Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul>	34	1000	48 (6 * 8 Byte je Wert)
	Ergebnisoffset berechnen* mit übertragener Position <ul style="list-style-type: none"> <li>• Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul> *Es muss eine gültige Position für den Detektor vorhanden sein	35	1000	48 (6 * 8 Byte je Wert)
	ID der aktiven Zielmarke	101	-	n
	Zielmarkenname	102	-	n
	Aktuelle Zielmarke zur Liste der Zielmarken hinzufügen (Parameter kann nur gesetzt werden!) Es muss eine Zielmarke detektiert worden sein.	103	-	n
Graustufe	Schwellenwert Min.	1	1000	n
	Schwellenwert Max.	2	1000	n
	Graustufenwert Min.	101	1000	n
	Graustufenwert Max.	102	1000	n

Detektor	Funktion	Wert	Multiplikator	Länge
	Graustufenwert invertieren	103	1	n
Kontrast Helligkeit	Schwellenwert Min.	1	1000	n
	Schwellenwert Max.	2	1000	n
Messschieber	Schwellenwert Abstand Min.	101	1000	n
	Schwellenwert Abstand Max.	102	1000	n
	Schwellenwert Abstand invertieren 0: nicht invertiert 1: invertiert	103	1	1
	Abstandsart 0: Minimum 1: Maximum 2: Mittlerer 3: Median 4: Kleinster Gegenüberliegender 5: Größter Gegenüberliegender	104	1	n
	Antaster 1: Schwellenwert Min.	105	1000	n
	Antaster 2: Schwellenwert Min.	106	1000	n
	Antaster 1: Glättung	107	1000	n
	Antaster 2: Glättung	108	1000	n
	Antaster 1: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	109	1	n
	Antaster 2: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	110	1	n
	Antaster 1: Anzahl Suchstreifen	111	1	n
	Antaster 2: Anzahl Suchstreifen	112	1	n
BLOB	Graustufenwert Min.	101	1000	n
	Graustufenwert Max.	102	1000	n

Detektor	Funktion	Wert	Multiplikator	Länge
	Graustufenwert invertieren 0: nicht invertiert 1: invertiert	103	1	1
	Schwellenwert Anzahl BLOB Min.	120	1	n
	Schwellenwert Anzahl BLOB Max.	121	1	n
	Schwellenwert Anzahl invertieren 0: nicht invertiert 1: invertiert	122	1	1
	Anzahl gesetzter Merkmale (nur lesend)	123	1	n
	Auswahl Merkmal aus Liste	124	1	n
	Schwellenwert Merkmal Min.	125	1000	n
	Schwellenwert Merkmal Max.	126	1000	n
	Schwellenwert Merkmal invertieren	127	1	1
Barcode Datacode OCR	Referenzstring	101	-	n (Länge des Strings)
Farbwert Farbfläche	Farbraum (nur lesend)	21	0 = RGB 1 = HSV 2 = LAB	3
	Kanalauswahl (nur lesend)	22	Bitfeld jeweils eine Stelle pro Farbkanal	4
	Farbkanal 1: Schwellenwert Min.	101	1000	n
	Farbkanal 1: Schwellenwert Max.	102	1000	n
	Farbkanal 1: Schwellenwert invertieren	103	1	n
	Farbkanal 2: Schwellenwert Min.	104	1000	n



Detektor	Funktion	Wert	Multiplikator	Länge
	Farbkanal 2: Schwellenwert Max.	105	1000	n
	Farbkanal 2: Schwellenwert invertieren	106	1	n
	Farbkanal 3: Schwellenwert Min.	107	1000	n
	Farbkanal 3: Schwellenwert Max.	108	1000	n
	Farbkanal 3: Schwellenwert invertieren	109	1	n
Farbliste	Farbraum (nur lesend)	21	0 = RGB 1 = HSV 2 = LAB	3
	Kanalauswahl (nur lesend)	22	Bitfeld jeweils eine Stelle pro Farbkanal	4
	Schwellenwert Farbabstand	101	1000	n
	Schwellenwert Farbabstand aktiv setzen	102	1	n
	Anzahl Farben in Liste	103	1	n
	Auswahl Farbe aus Liste	104	1	n
	Farbwert der ausgewählten Farbe (Farbkanal 1, Farbkanal 2, Farbkanal 3, Farbkanal 4 [konstant 0])	105	1000	32
Busbar Wafer	Schwellenwert Min.	1	1000	n
	Schwellenwert Max.	2	1000	n
Ergebnisverarbeitung	Name des aktiven Ausdrucks	122	-	n (Länge des Strings)
	Aktueller Ausdruck	124	-	n (Länge des Strings)

## Parameter lesen (ASCII)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

### Übersicht Telegramme

Parameter lesen (GPA) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Parameter lesen (Get Parameter)
2	<b>P</b>	
3	<b>A</b>	
4 - 6	X	Detektor Nummer z.B. 001
7 - 9	X	Kommando: Parameter-Nummer, siehe Tabelle <a href="#">Übersicht Detektor-Parameter</a>
Beispiel:	GPA001001	
Parameter lesen (GPA) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Parameter lesen (Get Parameter)
2	<b>P</b>	
3	<b>A</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 8	X	SI08 - Signed Integer 08 UI08 - Unsigned Integer 08 SI16 - Signed Integer 16 UI16 - Unsigned Integer 16 SI32 - Signed Integer 32 UI32 - Unsigned Integer 32 SI40 - Signed Integer 40 UI40 - Unsigned Integer 40 FLOT - Float DOBL - Double STRG - String BOOL - Boolean SP08 - Special Signed 8 UDEF - Undefined IARR - Integer Array ZERO - Default Zero Parameter
9 - 13	X	Länge des Werts (n) z.B. 00005

14 ... n	X	Wert
Beispiel:	GPAPSTRG00005Test1	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

## Suchbereich (ROI) setzen (ASCII)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme

ROI setzen (SRP/SRT) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Suchbereich setzen (Set ROI)
2	<b>R</b>	
3	<b>P</b> <b>T</b>	P = Permanent T = Temporär
4 - 11	X	ROI Info Länge in Bytes, ab Byte 5 bis Ende 39 Byte: Kreis 55 Byte: Rechteck, Ellipse, Freiform
12 - 14	X	Detektor Nr. z.B. 001
15 - 16	X	ROI Index 00: für gelben Suchbereich 01: für roten Teachbereich 02: Positionskontrolle
17 - 18	X	ROI Form 01: Kreis 02: Rechteck 03: Ellipse 04: Freiform
19 - 26	X	Zentrum X (Wert in Pixeln * 1000), z.B. 160 Pixel = 00160000
27 - 34	X	Zentrum Y (Wert in Pixeln * 1000), z.B. 120 Pixel = 00120000
35 - 42	X	Halbe Breite / X-Radius (Wert in Pixeln * 1000), z.B. 80 Pixel = 00080000
43 - 50	X	Halbe Höhe (nicht bei Kreis) (Wert in Pixel * 1000), z.B. 40 Pixel = 00040000
51 - 58	X	Winkel (nicht bei Kreis) (Wert in ° * 1000), z.B. 180° = 00180000
Beispiel:	SRP000000550010002001600000012000000 0800000004000000180000 Länge=55, Detektor=1, gelber Suchbereich, Rechteck, Zentrum X=160, Zentrum Y=120, Halbe Breite= 80, Halbe Höhe=40, Orientierung=180	

<b>ROI setzen (SRP/SRT) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Suchbereich setzen (Set ROI)
2	<b>R</b>	
3	<b>P</b> <b>T</b>	Permanent Temporär
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
Beispiel:	SRPP	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)
Parameter:		Die Parameter werden im Koordinatensystem der Lagenachführung und nicht im Koordinatensystem des Bildes angegeben.

## Suchbereich (ROI) lesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>ROI lesen (GRI) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Suchbereich lesen (Get ROI)
2	<b>R</b>	
3	<b>I</b>	
4 - 6	X	Detektor Nr. z.B. 001
7 - 8	X	ROI Index 00: für gelben Suchbereich 01: für roten Teachbereich 02: Positionskontrolle
Beispiel:	GRI00100	
<b>ROI lesen (GRI) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Suchbereich lesen (Get ROI)
2	<b>R</b>	
3	<b>I</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 12	X	ROI Info Länge in Bytes, ab Byte 5 bis Ende 39 Byte: Kreis 55 Byte: Rechteck, Ellipse, Freiform
13 - 15	X	Detektor Nr. z.B. 001
16 - 17	X	ROI Index 00: für gelben Suchbereich 01: für roten Teachbereich 02: Positionskontrolle
18 - 19	X	ROI Form 01: Kreis 02: Rechteck 03: Ellipse 04: Freiform
20 - 27	X	Zentrum X (Wert in Pixeln * 1000)

28 - 35	X	Zentrum Y (Wert in Pixeln * 1000)
36 - 43	X	Halbe Breite / X-Radius (Wert in Pixeln * 1000)
44 - 51	X	Halbe Höhe (nicht bei Kreis) (Wert in Pixel * 1000), z.B. 40 Pixel = 00040000
52 - 59	X	Winkel (nicht bei Kreis) (Wert in ° * 1000), z.B. 180° = 00180000
Beispiel:	GRIP000000550010002001600000012000000 0800000004000000090000 (Länge= 55, Detektor 1, Suchbereich, Rechteck, Zentrum X=160, Zentrum Y=120, Halb Breite=80, Halb Höhe=40, Winkel = 90)	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Low	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

## Suchbereichsinhalt setzen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Suchbereichsinhalt setzen (SRC) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Suchbereichsinhalt setzen (Set ROI Content)
2	<b>R</b>	
3	<b>C</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
6-8	X	Detektor Nr. 0: Lagenachführung >0: Standarddetektor im Job
9-10	00	Reserviert
11	X	0: Maskierungsdatei nicht verwenden 1: Maskierungsdatei verwenden
12-16	00000	Reserviert
17-19	X	Länge des darauffolgenden Dateinamens der Datei (Tatsächliche Byte-Anzahl mit Dateierweiterung, max. 255 Bytes) Oder für 000: Standardname mask.pgm
20 ... n	X	Dateiname (UTF-8) und Format verfügbar auf dem Gerät im Verzeichnis "/tmp/". Standardname mask.pgm Dateiformat: PGM
Beispiel:	SRC 1 1 001 00 100000 008 Test.pgm (für Datei "Test.pgm") SRC 1 1 001 00 100000 000 (für Datei: "mask.pgm")	
Suchbereichsinhalt setzen (SRC) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Suchbereichsinhalt setzen (Set ROI Content)
2	<b>R</b>	
3	<b>C</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5-7	X	<a href="#">Fehlercodes</a>



Beispiel:	SRC P 000
Weitere Informationen:	
Akzeptiert im Runmode:	Ja
Akzeptiert im Konfigurationsmodus:	Nein
Akzeptiert wenn Ready Low:	Ja
Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:	Max. 4 Byte (optional)

## Suchbereichsinhalt lesen (ASCII)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme


Suchbereichsinhalt lesen (GRC) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Suchbereichsinhalt lesen (Get ROI Content)
2	<b>R</b>	
3	<b>C</b>	
4	1	Version des Request
5-7	X	Detektor Nr. 0: Lagenachführung >0: Standarddetektor im Job
8-9	00	Reserviert
10	X	0: Maskierungsdatei nicht erzeugen 1: Maskierungsdatei erzeugen (sofern Maskierung im Job vorhanden)
11	X	0: Musterdatei nicht erzeugen 1: Musterdatei erzeugen
12	X	0: Konturdatei nicht erzeugen 1: Konturdatei erzeugen
13-15	000	Reserviert
16-18	X	Länge des darauffolgenden Dateinamens der Datei (Tatsächliche Byte-Anzahl mit Dateierweiterung, max. 255 Bytes) Oder für 000: Standardname mask.pgm / pattern.pgm / contour.pgm
19 ... n	X	Dateiname (UTF-8) und Format verfügbar auf dem Gerät im Verzeichnis "/tmp/". Standardname mask.pgm / pattern.pgm / contour.pgm Dateiformat: PGM
Beispiel:	GRC 1 001 00 100000 008 Test.pgm (Datei mit Standarderweiterung: Testmask.pgm) GRC 1 001 00 111000 008 Test.pgm (Dateien mit Standarderweiterung: Testmask.pgm, Testpattern.pgm & Testcontour.pgm) GRC 1 001 00 111000 000 (Datei nur mit Standarderweiterung: mask.pgm, pattern.pgm & contour.pgm)	

<b>Suchbereichsinhalt lesen (GRC) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Suchbereichsinhalt lesen (Get ROI Content)
2	<b>R</b>	
3	<b>C</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5-7	X	<a href="#">Fehlercodes</a>
8-12	X	Breite des Suchbereichs
13-17	X	Höhe des Suchbereichs
Beispiel:	GRC P 000 01080 01440	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Jobliste lesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)


Jobliste lesen (GJL) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Jobliste lesen (Get Job List)
2	<b>J</b>	
3	<b>L</b>	
Beispiel:	GJL	
Jobliste lesen (GJL) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Jobliste lesen (Get Job List)
2	<b>J</b>	
3	<b>L</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	Version des Response
8 - 10	X	Anzahl der Jobs
11 - 13	X	Aktive Job Nummer
 <b>HINWEIS:</b> Folgende Bytefolge wird für jeden Job von 1 bis "Anzahl der Jobs" wiederholt. Die Byte Nummern verschieben sich entsprechend.		
14 - 16	X	Anzahl der Zeichen für den Jobnamen. Damit kann ein eindeutiger Name für den Job n angegeben werden.
17 ... n	X	Ab dieser Position folgt, in der angegebenen Länge, der Name für Job n.
n+1 ... n + 3	X	Anzahl der folgenden Bytes. Damit kann eine Beschreibung für den Job n angegeben werden.
n + 4 ... m	X	Ab dieser Position folgt, in der angegebenen Länge, die Beschreibung für Job n.
m + 1 ... m + 3	X	Anzahl der folgenden Bytes. Damit kann ein eindeutiger Name für den Autor des Job n angegeben werden.

m + 4 ... k	X	Ab dieser Position folgt, in der angegebenen Länge, der Name für den Autor aus Job n.
k + 1 ... k + 19	X	Datum der Erstellung von Job n (19 Byte)
k + 20 ... k + 39	X	Datum der letzten Änderung von Job n (19 Byte)
Beispiel:	GJLP001001001007testjob010DefaultJob 004Test2014112720141128	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

## Detektorliste lesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Detektorliste lesen (GDL) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Get Detector List
2	<b>D</b>	
3	<b>L</b>	
Beispiel:	GDL	
Detektorliste lesen (GDL) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Get Detector List
2	<b>D</b>	
3	<b>L</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	Job Nummer des aktuellen Jobs
8 - 10	X	Anzahl der Detektoren im aktuellen Job
	 <b>HINWEIS:</b> Folgende Bytefolge wird für jeden Detektor im Job wiederholt. Die Byte Nummern verschieben sich entsprechend.	
11 - 13	X	Anzahl der folgenden Bytes. Damit kann ein eindeutiger Name für den Detektor n angegeben werden.
14 ... n	X	Ab dieser Position folgt, in der angegebenen Länge, der Name für Detektor n.

n + 1 ... n+ 5	X	001 - Mustervergleich 004 - Kontur 005 - Graustufe 006 - Kontrast 007 - Helligkeit 011 - OCR 013 - Datacode 014 - Barcode 018 - Farbwert 019 - Farbfläche 020 - Farbliste 021 - Messschieber 022 - BLOB 024 - Kontur 3D 027 - Ergebnisverarbeitung 028 - Zielmarke 3D
Beispiel:	GDLP001001012testdetector00005	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

## Detektor einlernen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Detektor einlernen (TED) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>T</b>	Detektor einlernen (Teach Detector)
2	<b>E</b>	
3	<b>D</b>	
4 - 6	X	0 = Lagenachführung ≥ 1 Detektoren
7	X	0: Temporär 1: Permanent
8	X	0: kein Trigger, Einlernen mit nächster Bildaufnahme 1: Trigger wird ausgeführt zum Einlernen
Beispiel:	TED00111	
<b>Detektor einlernen (TED) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>T</b>	Detektor einlernen (Teach Detector)
2	<b>E</b>	
3	<b>D</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
Beispiel:	TEDP	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)



**Trigger-Verzögerung setzen (ASCII)**

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Trigger-Verzögerung setzen (STD) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Trigger-Verzögerung setzen (Set Trigger Delay)
2	<b>T</b>	
3	<b>D</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
6 - 13	X	Trigger-Verzögerung in msec (max. 3000 msec) in Encoder-Schritten (max. 65535 Schritte)
Beispiel:	STD1100001000	
<b>Trigger-Verzögerung setzen (STD) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Trigger-Verzögerung setzen (Set Trigger Delay)
2	<b>T</b>	
3	<b>D</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
Beispiel:	STDP000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Trigger-Verzögerung lesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Trigger-Verzögerung lesen (GTD) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Trigger-Verzögerung lesen (Get Trigger Delay)
2	<b>T</b>	
3	<b>D</b>	
4	1	Version des Request
Beispiel:	GTD1	
<b>Trigger-Verzögerung lesen (GTD) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Trigger-Verzögerung lesen (Get Trigger Delay)
2	<b>T</b>	
3	<b>D</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	Fehlercode
8 - 15	X	Trigger-Verzögerung in msec (max. 3000 msec) in Encoder-Schritten (max. 65535 Schritte)
Beispiel:	GTDP00000001000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Job Permanent speichern (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Job Permanent speichern (SJP) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Job Permanent speichern (Store Job Permanently)
2	<b>J</b>	
3	<b>P</b>	
Beispiel:	SJP	
<b>Job Permanent speichern (SJP) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Job Permanent speichern (Store Job Permanently)
2	<b>J</b>	
3	<b>P</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
Beispiel:	SJPP	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## 11.4.4 Kalibrierung

### Kalibrierung: Initialisieren (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Initialisieren (CCD) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	C	Initialisieren (Calibration: Clear Data)
2	C	
3	D	
Beispiel:	CCD	
Initialisieren (CCD) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	C	Initialisieren (Calibration: Clear Data)
2	C	
3	D	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
Beispiel:	CCDP	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

### Kalibrierung: Weltpunkt hinzufügen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Kalibrierung: Weltpunkt hinzufügen (CAW) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Weltpunkt hinzufügen (Calibration: Add World Point)
2	<b>A</b>	
3	<b>W</b>	
4	1	Version des Request
5	X	1: Nur Referenzmarken Kalibrierplatte (Roboter) 4: Weltpunkt und Bildpunkt Punktpaarliste (Roboter)
6 - 10	0	Konstant (5 Byte)
11 - 18	X	Welt-X (in Kundeneinheit * 1000)
19 - 26	X	Welt-Y (in Kundeneinheit * 1000)
27 - 34	0	Konstant (8 Byte)
Beispiel:	CAW10000100100000002000000000000000 (Welt-X = 100 mm; Welt-Y = 200 mm)	
Kalibrierung: Weltpunkt hinzufügen (CAW) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Weltpunkt hinzufügen (Calibration: Add World Point)
2	<b>A</b>	
3	<b>W</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 12	X	Aktuelle Anzahl Punkte
13 - 20	X	Bildpunkt X
21 - 28	X	Bildpunkt Y
Beispiel:	CAWP000000010028800000566000 (Referenzpunkt 1; Bild-X=288; Bild-Y=566)	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja

Akzeptiert im Konfigurationsmodus:	Nein
Akzeptiert wenn Ready Low:	Ja
Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:	Max. 4 Byte (optional)

Hinweis: Für den Request CAW muss das Gesamt-Jobergebnis positiv sein.

## Kalibrierung: Punktpaarliste (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Kalibrierung per Punktpaarliste (CCL) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	C	Kalibrierung: Punktpaarliste (Calibration: Calibrate by Point List)
2	C	
3	L	
4	X	0: Temporär 1: Permanent
Beispiel:	CCL1	
Kalibrierung Punktpaarliste (CCL) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	C	Kalibrierung: Punktpaarliste (Calibration: Calibrate by Point List)
2	C	
3	L	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 9	X	Aktuell höchster Punktpaarindex
10 - 17	X	Abweichung Kalibrierung, RMSE
18 - 25	X	Abweichung Kalibrierung, Mittelwert
26 - 33	X	Abweichung Kalibrierung, Max.
34 - 41	X	Abweichung Kalibrierung, Min.
Beispiel:	CCLP0001012345678123456781234567812345678	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

## Kalibrierung: Punktpaarliste validieren (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Kalibrierung: Punktpaarliste validieren (CVL) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Punktpaarliste validieren (Calibration: Validate by Point List)
2	<b>V</b>	
3	<b>L</b>	
Beispiel:	CVL	
Kalibrierung: Punktpaarliste validieren (CVL) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Punktpaarliste validieren (Calibration: Validate by Point List)
2	<b>V</b>	
3	<b>L</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 9	X	Aktuell höchster Punktpaarindex
10 - 17	X	Abweichung Kalibrierung, RMSE
18 - 25	X	Abweichung Kalibrierung, Mittelwert
26 - 33	X	Abweichung Kalibrierung, Max.
34 - 41	X	Abweichung Kalibrierung, Min.
Beispiel:	CVLP0001012345678123456781234567812345678	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)



## Kalibrierung: Kalibrierplatte (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Kalibrierung: Kalibrierplatte (CCP) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Kalibrierplatte (Calibration: Calibrate by Plate)
2	<b>C</b>	
3	<b>P</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
6	X	0: Es werden keine Referenzmarken verwendet. Ursprung Mess-Koordinatensystem identisch mit Ursprung Kalibrierplatten-Koordinatensystem. 1: Es werden keine Referenzmarken verwendet. Mess-Koordinatensystem identisch mit Kamera-Koordinatensystem. 2: Verwendet Weltsystem, Referenzmarke Job 3: Verwendet Weltsystem, Referenzmarke Befehl <b>CAW</b>
7	X	0: Kalibrierung interne und externe Sensor-Parameter 1: Validierung der Kalibrierung 2: Kalibrierung interne Sensor-Parameter 5: Kalibrierung Transformation Mess-Koordinatensystem
Beispiel:	CCP1110	
Kalibrierung: Kalibrierplatte (CCP) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Kalibrierplatte (Calibration: Calibrate by Plate)
2	<b>C</b>	
3	<b>P</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>

8 - 12	X	Anzahl aktuell detektierter Kalibrierpunkte
13 - 20	X	Abweichung Kalibrierung, RMSE
21 - 28	X	Abweichung Kalibrierung, Mittelwert
29 - 36	X	Abweichung Kalibrierung, Max.
37 - 44	X	Abweichung Kalibrierung, Min.
45 - 52	X	CPF_MF X (in Kundeneinheit * 1000)
53 - 60	X	CPF_MF Y (in Kundeneinheit * 1000)
61 - 68	0	CPF_MF Z (in Kundeneinheit * 1000)
69 - 76	0	CPF_MF Winkel X (in Grad * 1000)
77 - 84	0	CPF_MF Winkel Y (in Grad * 1000)
85 - 92	X	CPF_MF Winkel Z (in Grad * 1000)
93 - 100	X	Abweichung Referenzmarken, Mittelwert
101 - 108	X	Abweichung Referenzmarken, Max.
109 - 116	X	Abweichung Referenzmarken, Min
Beispiel:	CCPP00000012000010010000200200003003000040040 00050050000600600007007000080080000900900001001	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

**Kalibrierung: Referenzmarke setzen (ASCII)**

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme

<b>Kalibrierung: Referenzmarke setzen (CSF) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Referenzmarke setzen (Calibration: Set Fiducial)
2	<b>S</b>	
3	<b>F</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
Beispiel:	CSF11	
<b>Kalibrierung: Referenzmarke setzen (CSF) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Referenzmarke setzen (Calibration: Set Fiducial)
2	<b>S</b>	
3	<b>F</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 15	X	X-Wert (in Kundeneinheit * 1000)
16 - 23	X	Y-Wert (in Kundeneinheit * 1000)
24 - 31	X	Z-Wert (in Kundeneinheit * 1000)
32 - 39	X	Winkel X-Wert (in Grad * 1000)
40 - 47	X	Winkel Y-Wert (in Grad * 1000)
48 - 55	X	Winkel Z-Wert (in Grad * 1000)
56 - 63	X	Abweichung Referenzmarken, Mittelwert
64 - 71	X	Abweichung Referenzmarken, Max.
72 - 79	X	Abweichung Referenzmarken, Min
Beispiel:	CSFP000000010010000200200003003000040040 00050050000600600001001000020200003003	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	

Akzeptiert im Konfigurationsmodus:	Nein
Akzeptiert wenn Ready Low:	Ja
Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:	Max. 4 Byte (optional)

## Kalibrierung: Bild hinzufügen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Kalibrierung: Bild hinzufügen (CAI) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Bild hinzufügen (Calibration: Add Image)
2	<b>A</b>	
3	<b>I</b>	
4	1	Version des Request
5	X	Modus 1: Mehrbild-Kalibrierung 2: Hand-Eye-Kalibrierung (Roboter) 3: Base-Eye-Kalibrierung (Roboter)
6-8	0	Anfügen am Ende der Liste (3 Byte)
9	X	Messebene definieren 0: Bild nicht verwenden, um Messebene zu definieren 1: Bild verwenden, um Messebene zu definieren
10-11	X	"Roboter: Rotationsreihenfolge" 00: Im Job angegebene Rotationsreihenfolge verwenden 01: Yaw-Pitch-Roll (z.B. Stäubli) 02: Roll-Pitch-Yaw (z.B. Kuka, Fanuc, Hanwha, ABB**, UR**) **bei Nutzung von entsprechender Umwandlungsfunktion
12-19	X	Pose_TCP Pos. X (in Kundeneinheit * 1000)
20-27	X	Pose_TCP Pos. Y (in Kundeneinheit * 1000)
28-35	X	Pose_TCP Pos. Z (in Kundeneinheit * 1000)
36-43	X	Pose_TCP Winkel X (in Grad * 1000)
44-51	X	Pose_TCP Winkel Y (in Grad * 1000)

52-59	X	Pose_TCP Winkel Z (in Grad * 1000)
Beispiel:	CAI11 001 1 02 000040040000500500006006000070070000800800009009	
<b>Kalibrierung: Bild hinzufügen (CAI) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Bild hinzufügen (Calibration: Add Image)
2	<b>A</b>	
3	<b>I</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5-7	X	<a href="#">Fehlercodes</a>
8-10	X	Aktuelle Anzahl Bilder in Liste
11-15	X	Anzahl insgesamt detektierter Punkte
Beispiel:	CAIP 000 001 00021	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Kalibrierung: Mehrbild (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Kalibrierung: Mehrbild (CMP) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Calibration Calibrate Multi-Image Plate
2	<b>M</b>	
3	<b>P</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
6	X	Ursprung des Welt-Koordinatensystems: 0: Welt-Koordinatensystem identisch mit dem Kalibrierplatten-Koordinatensystem (Zentrum der Platte). 1: Ursprung des Welt-Koordinatensystems so, dass dieser mit Ursprung des Bild-Koordinatensystems übereinstimmt (oberes linkes Pixel). 2: Welt-Koordinatensystem von Referenzpunkten verwenden, wie in der Jobdatei angegeben. 3: Welt-Koordinatensystem von Referenzpunkten verwenden, wie im Request CAW eingestellt.
7	X	Modus 0: Kalibrieren (interne und externe Parameter) 1: Validieren (vorhandene Kalibrierung verwenden; mindestens ein Kalibrierpunkt wird hinzugefügt. Über Rückprojektion kann zurückgeschlossen werden, ob der Punkt zur aktuellen Kalibrierung passt, oder verschoben ist) 2: Kalibrieren (nur interne Parameter) 3: Kalibrieren (nur externe Parameter unter Verwendung der neuen internen Parameter) 4: Kalibrieren (nur externe Parameter) 5: Nur Messebene (CPF_MF) kalibrieren
Beispiel:	CMP1105	

Kalibrierung: Mehrbild (CMP) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Calibration Calibrate Multi-Image
2	<b>M</b>	
3	<b>P</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5-7	X	<a href="#">Fehlercodes</a>
8-10	X	Abdeckung Sichtfeld in %
11-15	X	Anzahl insgesamt detektierter Punkte
16-18	X	Anzahl verwendeter Bilder
19-21	X	Anzahl ungültiger Bilder
22	X	Neigung zwischen Kalibrierplatten-Posen 0: ausreichend 1: nicht ausreichend
23-30	X	Abweichung Kalibrierung, RMSE [px]
31-38	X	Abweichung Kalibrierung, Max. [px]
39-46	X	Abweichung Referenzmarken, RMSE (in Kundeneinheit * 1000)
47-54	X	Abweichung Referenzmarken, Max. (in Kundeneinheit * 1000)
Beispiel:	CMPP 000 089 00312 011 002 0 00001001000020020000300300004004	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	



**Kalibrierung: Roboter Mehrbild (ASCII)**

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

Übersicht Telegramme

<b>Kalibrierung: Roboter Mehrbild (CRP) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Kalibrierplatte Roboter (Calibration: Calibrate Robotics Plate)
2	<b>R</b>	
3	<b>P</b>	
4	1	Version des Request
5	x	0: Temporär 1: Permanent
6	X	Ursprung des Welt-Koordinatensystems: 4: Worldframe auf Benutzer-Roboterframe setzen
7	X	Modus 0: Kalibrieren (interne und externe Parameter) 1: Validieren (vorhandene Kalibrierung verwenden; mindestens ein Kalibrierpunkt wird hinzugefügt. Über Rückprojektion kann zurückgeschlossen werden, ob der Punkt zur aktuellen Kalibrierung passt, oder verschoben ist) 2: Kalibrieren (nur interne Parameter) 3: Kalibrieren (nur externe Parameter unter Verwendung der neuen internen Parameter) 4: Kalibrieren (nur externe Parameter) 5: Nur Messebene (CPF_MF) kalibrieren 6: Nur Hand-Eye (TCP_CF)/Base-Eye (RF_CF) kalibrieren
Beispiel:	CRP1140	
<b>Kalibrierung: Roboter Mehrbild (CRP) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Kalibrierplatte Roboter (Calibration: Calibrate Robotics Plate)
2	<b>R</b>	
3	<b>P</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler

5-7	X	<a href="#">Fehlercodes</a>
8-10	X	Abdeckung Sichtfeld (%)
11-15	X	Anzahl insgesamt detektierter Punkte
16-18	X	Anzahl verwendeter Bilder
19-21	X	Anzahl ungültiger Bilder
22-29	X	Abweichung Kalibrierung, RMSE [px]
30-37	X	Abweichung Kalibrierung, Max. [px]
38-45	X	Abweichungen Kalibrierplatten-Pose Translation RMSE (in Kundeneinheit * 1000)
46-53	X	Abweichungen Kalibrierplatten-Pose Translation Max. (in Kundeneinheit * 1000)
54-61	X	Abweichungen Kalibrierplatten-Pose Rotation RMSE (in Grad * 1000)
62-69	X	Abweichungen Kalibrierplatten-Pose Rotation Max. (in Grad * 1000)
Beispiel:	CRPP 000 092 01349 012 004 0000100100002002 00003003000040040000500500006006	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Keine Veränderung	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

## Kalibrierung: Kalibrierung kopieren (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Kalibrierung: Kalibrierung kopieren (CCC) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	C	Kalibrierung: Kalibrierung kopieren (Calibration: Copy Calibration)
2	C	
3	C	
4	1	Version des Request
5	1	Konstant
6 - 8	X	Destination 0 : In alle Jobs kopieren >0 : In angegebenen Job kopieren
9	X	0: Immer kopieren, wenn die Kalibrierung aktiv ist. 1: Nur kopieren, wenn die Kalibrieremethode identisch ist.
Beispiel:	CCC110021	
<b>Kalibrierung: Kalibrierung kopieren (CCC) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	C	Kalibrierung: Kalibrierung kopieren (Calibration: Copy Calibration)
2	C	
3	C	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 10	X	Jobnummer des Jobs, indem der Fehler auftritt 00: Erfolgreich >0 - Jobnummer des Jobs, indem der Fehler zuerst auftritt
Beispiel:	CCCP000000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja

Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:	Max. 4 Byte (optional)

## Kalibrierung: Parameter setzen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Kalibrierung: Parameter setzen (CSP) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Parameter setzen (Calibration: Set Parameter)
2	<b>S</b>	
3	<b>P</b>	
4	1	Version des Request
5	X	0: Temporär 1: Permanent
6 - 8	X	Parameter-Nummer, siehe Tabelle <a href="#">Kalibrierparameter CSP und CGP</a>
9 - 16	X	Länge des Werts
17 ... n	X	Wert für gewählten Parameter, siehe Tabelle <a href="#">Kalibrierparameter CSP und CGP</a>
Beispiel:	CSP1100200000019	
Kalibrierung: Parameter setzen (CSP) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Parameter setzen (Calibration: Set Parameter)
2	<b>S</b>	
3	<b>P</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
Beispiel:	CSPP000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

Kalibrierparameter: siehe Tabelle [Kalibrierparameter für Telegramme CSP und CGP](#)

## Kalibrierung: Parameter lesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Kalibrierung: Parameter lesen (CGP) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Parameter lesen (Calibration: Get Parameter)
2	<b>G</b>	
3	<b>P</b>	
4	1	Version des Request
5 - 7	X	Parameter-Nummer, siehe <a href="#">Kalibrierparameter CSP und CGP</a>
Beispiel:	CGP1001	
<b>Kalibrierung: Parameter lesen (CGP) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>C</b>	Kalibrierung: Parameter lesen (Calibration: Get Parameter)
2	<b>G</b>	
3	<b>P</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 10	X	Parameter-Nummer, siehe <a href="#">Kalibrierparameter CSP und CGP</a>
11 - 18	X	Länge der nachfolgenden Daten
19 ... n	X	Parameter Werte, abhängig von gewähltem Parameter
Beispiel:	CGPP000001000000011	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Kalibrierparameter für Telegramme CSP und CGP

Parameter-Bezeichnung	Parameter Nummer	Wert	Länge	Status Kalibrierung nach CSP
Status Kalibrierung	001	0: Ungültig 1: Gültig	1 Byte	-*
Kalibriermethode	002	0: Keine 2: Punktpaarliste (Roboter) 3: Kalibrierplatte (Messen) 4: Kalibrierplatte (Roboter) 5: Hand-Eye-Kalibrierung (Roboter) 6: Base-Eye-Kalibrierung (Roboter)	1 Byte	ungültig
Einheit (Kundeneinheit)	004	0: Millimeter [mm] 1: Zentimeter [cm] 2: Meter [m] 3: Inch ["] 4: Beliebige Einheit [au]	1 Byte	keine Änderung
Interne Parameter	010	Brennweite (in mm *1000) Kappa (*1000) Pixelabstand X (in $\mu\text{m}$ * 1000) Pixelabstand Y (in $\mu\text{m}$ * 1000) Koordinatenursprung X (in Pixel * 1000) Koordinatenursprung Y (in Pixel * 1000) Bildgröße X (Anzahl Pixel) Bildgröße Y (Anzahl Pixel)	64 (8 * 8 Byte je Wert)	-*
Bezug Kamera- zu Mess-Koordinatensystem (CF_MF)	011	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	48 (6 * 8 Byte je Wert)	-*
Bezug Kamera- zu Kalibrierplatten-Koordinatensystem (CF_CPF)	012	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	48 (6 * 8 Byte je Wert)	-*



Parameter-Bezeichnung	Parameter Nummer	Wert	Länge	Status Kalibrierung nach CSP
Bezug Roboter- zu Kamera-Koordinatensystem (RF_CF)	013	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	48 (6 * 8 Byte je Wert)	–*
Bezug Kalibrierplatten- zu Mess-Koordinatensystem (CPF_MF)	014	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	48 (6 * 8 Byte je Wert)	–*
Bezug Roboter- zu Mess-Koordinatensystem (RF_MF)	015	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	48 (6 * 8 Byte je Wert)	–*
Bezug Werkzeug- zu Kamera-Koordinatensystem (TCP_CF)	016	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	48 (6 * 8 Byte je Wert)	–*
Bezug Roboter- zu Werkzeug-Koordinatensystem (RF_TCP)	017	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	48 (6 * 8 Byte je Wert)	keine Änderung
Z-Verschiebung Mesesebene	021	Wert (in Kundeneinheit * 1000)	8 Byte	keine Änderung
Brennweite in [mm]	022	[mm * 1000]	8 Byte	ungültig (CSP nur für C-Mount)
Kalibrierplatten-Typ	023	Zeichenkette mit Namen der Beschreibungsdatei	n	ungültig
Referenzmarke 1	024	Translation X, Y, Z (in Kundeneinheit * 1000)	24 (3 * 8 Byte je Wert)	ungültig
Referenzmarke 2	025			
Referenzmarke 3	026			
Referenzmarke 4	027			

Parameter-Bezeichnung	Parameter Nummer	Wert	Länge	Status Kalibrierung nach CSP
Anzahl vorhandener Kalibrierplatten-Typen	037	Request – Auswahl Typ: 0: Alle 1: Messen 2: Roboter Response: Anzahl Platten	Request: 1 Response: 5	–*
Vorhandene Kalibrierplatten-Typen (Dateinamen)	038	Request – Auswahl Typ: 0: Alle 1: Messen 2: Roboter Request – Index: 0: Alle Dateinamen >0: Auswahl Index Antwort: Dateinamen Kalibrierplatten	Request: 1 / 5 Response: n (String)	–*
Roboter: Rotationsreihenfolge	039	"Roboter: Rotationsreihenfolge" 00: Im Job angegebene Rotationsreihenfolge verwenden 01: Yaw-Pitch-Roll (z.B. Stäubli) 02: Roll-Pitch-Yaw (z.B. Kuka, Fanuc, Hanwha, ABB**, UR**) **bei Nutzung von entsprechender Umwandlungsfunktion	1	ungültig
Mittlere Sensorauflösung	041	Wert (in Kundeneinheit/Pixel * 1000)	8 Byte	–*

\* CSP nicht möglich (Parameter read-only, kann nicht gesetzt werden).

## 11.4.5 Visualisierung

### Bild holen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Bild holen (GIM) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Bild holen (Get Image)
2	<b>I</b>	
3	<b>M</b>	
4	X	0: Letztes Bild 1: Letztes Schlecht-Bild 2: Letztes Gut-Bild
Beispiel:	GIM1	
<b>Bild holen (GIM) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Bild holen (Get Image)
2	<b>I</b>	
3	<b>M</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5	X	<a href="#">Fehlercodes</a>
6	X	Typ Bild 0: Graustufen 3: Bayer-Pattern_BG Bei Konvertierung des Farbbildes von Bayer in RGB, muss der entsprechende Bild Typ berücksichtigt werden. Vorverarbeitungsfilter der Kategorie "Anordnung" haben Einfluss auf den Bayer-Type. Bayer-Pattern fängt mit Blau - Grün an.
7	X	Bilgergebnis 1: Gutbild 0: Fehlerbild
8 - 11	X	Anzahl der Zeilen z.B. 0480 / 0200

12 - 15	X	Anzahl der Spalten z.B. 0640 / 0320
16 - 19	X	Ende des Nachrichten-Strings wenn angegeben. Ansonsten Start Bilddaten ab Byte Nr. 16.
20 ... n	X	Binäre Bilddaten (Zeilen * Spalten)
Beispiel:	GIMP0004800640...	
Weitere Informationen:		
Akzeptiert im Runmode:	Ja	
Akzeptiert im Konfigurationsmodus:	Nein	
Akzeptiert wenn Ready Low:	Ja	
Status des Signals Ready während Bearbeitung:	Low	
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Telegrammende:	Max. 4 Byte (optional)	

### 11.4.6 Service (nur auf Port 1998 und in ASCII Format verfügbar)

#### Visualisierungsdaten aktualisieren (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Visualisierungsdaten aktualisieren (UVR) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>U</b>	Visualisierungsdaten aktualisieren (Update Visualization Results)
2	<b>V</b>	
3	<b>R</b>	
4	1	Version des Request
5	X	Bild: 0: Kein Bild wird erstellt 1: Grauwert- / RGB-Bild ohne Filter, Format BMP 2: Grauwert-Bild / Bayer-Pattern ohne Filter, Format BMP 3: Grauwert- / RGB-Bild mit Filter, Format BMP 4: Grauwert-Bild / Bayer-Pattern mit Filter, Format BMP 5: Grauwert- / RGB-Bild ohne Filter, Format JPEG (Kompression niedrig) 6: Grauwert- / RGB-Bild mit Filter, Format JPEG (Kompression niedrig) 7: Grauwert- / RGB-Bild ohne Filter, Format JPEG (Kompression hoch) 8: Grauwert- / RGB-Bild mit Filter, Format JPEG (Kompression hoch)
6	X	Ergebnis XML: 0: Ergebnisdatei wird nicht erstellt 1: Ergebnisdatei wird erstellt
7	X	Statistik XML: 0: Statistik-Datei wird nicht erstellt 1: Statistik-Datei wird erstellt

8	X	Bildtyp: 0: Letztes beliebiges Bild (Any) 1: Letztes Fehler-Bild (Fail) 2: Letztes erfolgreiches Bild (Pass) 3: Nächstes beliebiges Bild (Any) 4: Nächstes Fehler-Bild (Fail) 5: Nächstes erfolgreiches Bild (Pass)
9 - 11	X	Verzeichnis Nummer (Konstant) 001: visu001
Beispiel:	UVR111110001	
<b>Visualisierungsdaten aktualisieren (UVR) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>U</b>	Visualisierungsdaten aktualisieren (Update Visualization Results)
2	<b>V</b>	
3	<b>R</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8	X	Reserviert
9 - 11	X	Verzeichnis-Nummer (Konstant) 001: visu001
Beispiel:	UVRP0000001	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Status des Signals Ready während Bearbeitung:		Keine Veränderung
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

Die erstellten Dateien liegen im Verzeichnis /tmp/[Verzeichnis-Nummer] zum Download bereit:

- image.bmp
- overlay.xml

Mit der Datei "overlay.xml" können alle relevanten Informationen zur Erstellung der Einzeichnung bezogen werden. Die Datei ist im XML Format erstellt. Die wichtigsten Elemente werden in der untenstehenden Tabelle beschrieben

Name		Wert	Beschreibung
detector	type	pattern_matching contour contrast brightness grey caliper blob ocr datacode barcode	Detektor-Typ
	number	Integer	Position in Detektor-Liste
	name	String	Name des in der Konfiguration definierten Detektors
roi	purpose	search teach position_control result	Art des Einzeichnung-Elements. Die verschiedene Arten haben unterschiedliche Farben.
	shape	rectangle rectangle_mask ellipse	Form des Einzeichnung-Elements
center	x	Float	Position des Zentrums in X (Pixel)
	y	Float	Position des Zentrums in Y (Pixel)
size	half_width	Float	Halbe Breite des Einzeichnung-Elements
	half_high	Float	Halbe Höhe des Einzeichnung-Elements
angle	angle	Float	Winkel des Einzeichnung-Elements (Grad)
number	value	Float	Anzahl der Einzeichnung-Elemente in diesem Detektor
line	x1	Float	Startpunkt X Linie 1 (Pixel)
	y1	Float	Startpunkt Y Linie 1 (Pixel)
	x2	Float	Startpunkt X Linie 2 (Pixel)
	y2	Float	Startpunkt Y Linie 2 (Pixel)

Abhängig vom Detektortyp (detector → type) gibt es verschiedene Elemente die angezeigt werden können. Folgende Tabelle gibt an, welches Element bei welchem Detektor angezeigt werden kann.

Detektor	search	teach	position_control	result
Mustervergleich	Ja	Ja	Ja	1
Kontur	Ja	Ja	Ja	200
Kontur 3D	Ja	Ja	Nein	20
Zielmarke 3D	Ja	Ja	Nein	1
Kontrast	Ja	Nein	Nein	Nein
Helligkeit	Ja	Nein	Nein	Nein
Graustufe	Ja	Nein	Nein	Nein
Messschieber	Ja	Nein	Nein	Nein
BLOB	Ja	Nein	Nein	1000
OCR	Ja	Nein	Nein	1
Datacode	Ja	Nein	Nein	5
Barcode	Ja	Nein	Nein	5



## Sensoridentität lesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Sensoridentität lesen (GSI) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Sensoridentität lesen (Get Sensor Identity)
2	<b>S</b>	
3	<b>I</b>	
4	1	Version des Request
Beispiel:	GSI1	
Sensoridentität lesen (GSI) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Sensoridentität lesen (Get Sensor Identity)
2	<b>S</b>	
3	<b>I</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 10	X	Länge der folgenden Daten
11 ... n	X	Version der Firmware sowie Informationen über die Hardware. Bereiche sind durch ein Semikolon eindeutig getrennt.
Beispiel:	GSIP0000262.0.0.3; SBSA-B-AF-R6-B-FW-W	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

## Firmware aktualisieren (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Firmware aktualisieren (UFW) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>U</b>	Firmware aktualisieren (Update Firmware)
2	<b>F</b>	
3	<b>W</b>	
4	1	Version des Request
Beispiel:	UFW1	
Firmware aktualisieren (UFW) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>U</b>	Firmware aktualisieren (Update Firmware)
2	<b>F</b>	
3	<b>W</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
Beispiel:	UFWP000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

Nach absenden des Befehls wird im Verzeichnis /tmp/ auf dem SBS Vision-Sensor auf eine gültige Firmware Datei geprüft. Der Name muss der typischen Namensvergabe entsprechen (z.B. wie nach dem Download von der Festo Homepage). Das Ende ist erreicht sobald die Kamera wieder Ready (Pin 4 GN) signalisiert. Alternativ kann über das Telegramm "GS11" geprüft werden, ob eine gültige Antwort gesendet wird.



### HINWEIS:

Bei der Firmwareaktualisierung ist die Spannungsversorgung sicherzustellen. Ein Update kann bis zu 10 Minuten dauern.



## Jobsatz einlesen (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Jobsatz setzen (SJS) Request String an Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Jobsatz einlesen (Set Jobset)
2	<b>J</b>	
3	<b>S</b>	
4	1	Version des Request
5 - 7	X	Länge des nachfolgenden Dateinamens. Maximale Länge 250 Zeichen.
8 ... n	X	Optionaler Dateiname. Wird kein Dateiname angegeben wird der Standardname „Jobset.job“ verwendet.
Beispiel:	SJS1010jobset.job	
<b>Jobsatz setzen (SJS) Response String vom Sensor (ASCII)</b>		
Byte Nr.	Inhalt	Bedeutung
1	<b>S</b>	Jobsatz einlesen (Set Jobset)
2	<b>J</b>	
3	<b>S</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
8 - 10	X	Aktive Jobnummer im geladenen Jobsatz
Beispiel:	SJSP000001	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Nein
Status des Signals Ready während Bearbeitung:		Low
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

Der Jobsatz mit dem angegebenen Namen wird im Verzeichnis /tmp/ auf dem SBS Vision-Sensor gesucht. Ist die Datei vorhanden, wird dieser Jobsatz aktiviert. Die Datei wird anschließend entfernt.

### Jobsatz sichern (ASCII)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

#### Übersicht Telegramme

Jobsatz sichern (GJS) Request String an Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Jobsatz vom SBS sichern
2	<b>J</b>	
3	<b>S</b>	
4	<b>1</b>	Version des Request
5 - 7	X	Länge des nachfolgenden Dateinamens. Maximale Länge 250 Zeichen.
8 ... n	X	Optionaler Dateiname. Wird kein Dateiname angegeben wird der Standardname „Jobset.job“ verwendet.
Beispiel:	GJS1010jobset.job	
Jobsatz sichern (GJS) Response String vom Sensor (ASCII)		
Byte Nr.	Inhalt	Bedeutung
1	<b>G</b>	Jobsatz vom SBS sichern
2	<b>J</b>	
3	<b>S</b>	
4	P F	P: (Pass) Erfolgreich F: (Fail) Fehler
5 - 7	X	<a href="#">Fehlercodes</a>
Beispiel:	GJSP000	
Weitere Informationen:		
Akzeptiert im Runmode:		Ja
Akzeptiert im Konfigurationsmodus:		Nein
Akzeptiert wenn Ready Low:		Ja
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
Telegrammende:		Max. 4 Byte (optional)

Der Jobsatz wird mit dem angegebenen Namen im Verzeichnis /tmp/ auf dem SBS Vision-Sensor abgelegt.

### 11.4.7 Datenausgabe ASCII

Ausgabedaten (ASCII), dynamisch zusammengesetzt nach Benutzereinstellungen in der Software unter:

Vision Sensor Configuration Studio / Ausgabe / Datenausgabe.

Prinzipieller String-Aufbau:

<START> (((<OPTIONAL FIELDS> <SEPARATOR> <PAYLOAD>))) <CHKSUM> <TRAILER>

#### Ausgabedaten (ASCII):

<OPTIONAL FIELDS>				
Parameter	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Gewählte Felder	Über diese Checkbox werden alle gewählten Felder angezeigt. Die Checkbox "Gewählte Felder" selbst wird nicht angezeigt.	16	Die Reihenfolge der Ausgabe ist von links nach rechts und von oben nach unten, d.h. pro aktiver Checkbox wird ein Byte beginnend beim LSB gesetzt.	Alle Typen
Telegrammlänge	Anzahl der Zeichen inkl. der Zeichen für die Telegrammlänge selbst.	1 ... 10	Z.B. Ausgabe String mit 10 Zeichen; Telegrammlänge 10 + 2 Zeichen (pro Dezimalstelle ein Byte) = 12	Alle Typen
Statusbyte	Gibt den Triggermodus wieder.	3	PPF = Trigger PPF = Freilauf	Alle Typen

◀OPTIONAL FIELDS▶				
Parameter	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Detektorergebnisse	Ausgabe des Gesamtergebnisses pro Detektor.	4 ... 261	Byte 1 = UND Verknüpfung aller Detektoren Byte 2 = Gesamtergebnis der Lagenachführung Byte 3 = Gesamtergebnis des aktuellen Jobs Folgend die Anzahl der Detektoren ein Byte pro Dezimalstelle Folgende per Detektor ein Byte P = Detektor Pass F = Detektor Fail	Alle Typen
Digitalausgänge	Gibt das Verknüpfungsergebnis pro digitalem Ausgang zurück.	2 ... 7	Byte 1 Anzahl aktiver Ausgänge (Verknüpfungsergebnis zugewiesen) Folgende Bytes 2 ...7 pro Ausgang ein Byte P = Detektor Pass F = Detektor Fail 0 = nicht aktiver Ausgang (Lücke zwischen zwei aktiven Ausgängen)	Alle Typen



<b>«OPTIONAL FIELDS»</b>				
<b>Parameter</b>	<b>Beschreibung</b>	<b>Länge ASCII [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Log. Ausgänge	Gibt das Verknüpfungsergebnis pro logischem Ausgang zurück	1 ... 259	Ab Byte 1 Anzahl aktiver Ausgänge Verknüpfungsergebnis zugewiesen) Byte pro Dezimalstelle Folgende Bytes pro logischem Ausgang ein Byte P = Detektor Pass F = Detektor Fail 0 = nicht aktiver Ausgang (Lücke zwischen zwei aktiven Ausgängen)	Alle Typen
Ausführungszeit	Gibt die Ausführungszeit der letzten Auswertung zurück.	1 ... 3	Signed Integer	Alle Typen
Aktiver Job	Gibt den Job der letzten Auswertung zurück.	1 ... 3	Unsigned Int U8	Alle Typen

**«PAYLOAD»**

**Übersicht detektorspezifische Nutzdaten – Werte**

**Allgemein**

<b>«PAYLOAD» Allgemein</b>				
<b>Wert</b>	<b>Beschreibung</b>	<b>Länge ASCII [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Zähler "Alle Auswertungen"	Gesamtanzahl der Prüfungen	1 ... 11	Signed Integer	GENERAL
Zähler Gutteile	Anzahl der Prüfungen mit Resultat "OK"	1 ... 11	Signed Integer	GENERAL
Zähler Schlechtheile	Anzahl der Prüfungen mit Resultat "Fehler"	1 ... 11	Signed Integer	GENERAL

«PAYLOAD» Allgemein				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Zeitüberschreitung	Zeigt eine Überschreitung der maximalen Zykluszeit an.	1	BOOL	GENERAL
Aufnahme	Gibt die Anzahl der Bildaufnahmewiederholungen der letzten Auswertung an. Nur in Kombination mit Mehrfachbildaufnahme.	1 ... 3	INT	GENERAL
String	Mit diesem Feld kann ein Konstanter String in die Datenausgabe eingegeben werden.	1 ... 50	STRING	GENERAL
Job-Prüfsumme	Berechnet eine Prüfsumme über den aktiven Job. Darin werden alle jobspezifischen Einstellungen berücksichtigt außer dem "Geändert"-Datum. Eine Änderung von Jobsatz-globalen Einstellungen führt zu einer Änderung der Prüfsumme in allen Jobs. Wird die Prüfsumme für einen Job ermittelt, können für diesen keine temporären Änderungen im Run-Modus durchgeführt werden.	8	STRING	GENERAL







## Basiswerte

«PAYLOAD» Basiswerte				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Score	[%]	1 ... 6	Signed Integer	Alle Detektoren
Gesamtergebnis	Boolesches Detektorergebnis **	1	BOOL	Alle Detektoren




«PAYLOAD» Basiswerte				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Ausführungszeit	Ausführungszeit des einzelnen Detektors in [msec].	1 ... 11	Signed Integer	Alle Detektoren

### Position / Lage





«PAYLOAD» Position / Lage				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Pos. X	X-Koordinate der gefundenen Position, 1/1000 [Kundeneinheit] **	1 ... 11	Signed Integer	
Pos. Y	Y-Koordinate der gefundenen Position, 1/1000 [Kundeneinheit] **	1 ... 11	Signed Integer	
Pos. Z	Z-Koordinate der gefundenen Position, 1/1000 [Kundeneinheit]	1 ... 11	Signed Integer	 Mit Ergebnisoffset: 
Delta Pos. X	Delta Position X zwischen eingelerntem und gefundenem Objekt, 1/1000 [Kundeneinheit]	1 ... 11	Signed Integer	



«PAYLOAD» Position / Lage				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Delta Pos. Y	Delta Position Y zwischen eingelerntem und gefundenem Objekt, 1/1000 [Kundeneinheit]	1 ... 11	Signed Integer	
Delta Pos. Z	Delta Position Z zwischen eingelerntem und gefundenem Objekt, 1/1000 [Kundeneinheit]	1 ... 11	Signed Integer	 <p>Mit Ergebnisoffset:</p>
Winkel X	Orientierung des gefundenen Objekts, bezogen auf X-Achse, 1/1000 [°]	1 ... 11	Signed Integer	 <p>Mit Ergebnisoffset:</p>
Winkel Y	Orientierung des gefundenen Objekts, bezogen auf Y-Achse, 1/1000 [°]	1 ... 11	Signed Integer	 <p>Mit Ergebnisoffset:</p>
Winkel Z	Orientierung des gefundenen Objekts, bezogen auf Z-Achse, 1/1000 [°]	1 ... 11	Signed Integer	
Winkel (45)	Orientierung der Bounding-Box des gefundenen Codes [°], Wertebereich: -45° ... 45°	1 ... 6	Signed Integer	

«PAYLOAD» Position / Lage				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Winkel (180)	Orientierung der Breite (lange Achse) des Objekts [°], Wertebereich: -90° ... 90° 0° = Ost, Gegen- uhrzeigersinn	1 ... 7	Signed Integer	
Winkel (360)	Orientierung der Breite (lange Achse) des Objektes [°], Wertebereich: -180° ... 180° 0° = Ost, Gegen- uhrzeigersinn	1 ... 7	Signed Integer	
Delta Winkel X	Winkel zwischen eingelerntem und gefundenem Objekt, bezogen auf X-Achse, 1/1000 [°]	1 ... 7	Signed Integer	 Mit Ergebnisoffset:
Delta Winkel Y	Winkel zwischen eingelerntem und gefundenem Objekt, bezogen auf Y-Achse, 1/1000 [°]	1 ... 7	Signed Integer	 Mit Ergebnisoffset:
Delta Winkel Z	Winkel zwischen eingelerntem und gefundenem Objekt, bezogen auf Z-Achse, 1/1000 [°]	1 ... 7	Signed Integer	 Mit Ergebnisoffset:
Pose 3D (X, Y, Z, Winkel X, Winkel Y, Winkel Z)	Koordinaten des gefundenen Objekts, 1/1000 [Kundeneinheit] Winkel: 1/1000 Grad	je 1...7 Byte pro Wert; getrennt durch angegebenes Trennzeichen	Signed Integer	 Mit Ergebnisoffset:



◀PAYLOAD▶ Position / Lage				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Delta Pose 3D (X, Y, Z, Winkel X, Winkel Y, Winkel Z)	Delta Koordinaten des gefundenen Objekts, 1/1000 [Kundeneinheit] Winkel: 1/1000 Grad	je 1...7 Byte pro Wert; getrennt durch angegebenes Trennzeichen	Signed Integer	 Mit Ergebnisoffset: 
Positionskontrolle		1	BOOL	




## Messen

◀PAYLOAD▶ Messen				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Höhe	Höhe des geometrischen Elements [Kundeneinheit]*, Höhe ≥ 0, Höhe ≤ Breite	1 ... 11	Signed Integer	
Breite	Breite des geometrischen Elements [Kundeneinheit]* Breite ≥ 0, Breite ≥ Höhe	1 ... 11	Signed Integer	
Radius	Radius des gefitteten Kreises [Kundeneinheit]	1 ... 11	Signed Integer	
Fläche	Fläche des BLOBs ohne Löcher, 1/1000 [Pixel]	1 ... 11	Signed Integer	

◀PAYLOAD▶ Messen				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Fläche (inkl. Löcher)	Fläche des BLOBs mit Löchern, 1/1000 [Pixel]	1 ... 11	Signed Integer	
Abstand	Berechneter Abstand [Kundeneinheit] **	1 ... 11	Signed Integer	



### Identifikation

◀PAYLOAD▶ Identifikation				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
String Zielmarken-ID Zielmarkenname	Inhalt des gelesenen Codes oder Inhalt der Zielmarke oder zugewiesener Zielmarkenname. Wird eine feste Stringlänge gewünscht, so müssen die minimale Stringlänge (Detektorspezifische Nutzdaten) und die maximale Stringlänge (Detektoreinstellungen) auf den gleichen Wert (z.B. 127) gesetzt werden.	0 ... 255	STRING	
Stringlänge Länge Zielmarkenname (Zeichen)	Länge des gelesenen Codes [Zeichen]	1 ... 6	Signed Integer	



<b>«PAYLOAD» Identifikation</b>				
<b>Wert</b>	<b>Beschreibung</b>	<b>Länge ASCII [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Stringlänge (Bytes) Länge Zielmarkenname (Bytes) Länge Zielmarken-ID (Bytes)	Länge des gelesenen Codes [Bytes]	1 ... 6	Signed Integer	 <b>A</b>
Stringvergleich	Überprüfung des Inhaltes der gelesenen Informationen. Die Überprüfung des Inhaltes der gelesenen Informationen erfolgt auf Basis von regulären Ausdrücken (siehe Detektor Datacode, Reiter Referenzstring)	1	BOOL	 <b>A</b>
Abgeschnitten	Code komplett oder abgeschnitten F: Code komplett P: Code abgeschnitten	1	BOOL	 <b>A</b>





## Identifikation - Qualität

◀PAYLOAD▶ Identifikation - Qualität				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Qualität - Gesamt	Ausgabe aller Q-Parameter. Abhängig von gewähltem Codetyp und Norm.	je 1 Byte pro Wert; getrennt durch angegebenes Trennzeichen Bei 2D Code Parameter Q9 (Mean light): 1...3	Unsigned Char; bei 2D Code Q9 (Meanlight) Unsigned Short	
Qualität - Einzel	Ausgabe der Einzelqualitätswerte: Auswahl Q1-Q24 abhängig von gewähltem Codetyp und Norm. Ziffern: 1-4 Buchstaben: A-F	1 Bei 2D Code Parameter Q9 (Mean light): 1...3	Unsigned Char; bei 2D Code Q9 (Meanlight) Unsigned Short	
Min. Qualität	Prüfung, ob die minimal geforderte Qualität erreicht wurde	1 ... 7	Unsigned Int	<b>A</b>

## Farbe

◀PAYLOAD▶ Farbe				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Farbwert: <ul style="list-style-type: none"> <li>• Rot, Grün, Blau</li> <li>• Farbton, Sättigung, Helligkeit</li> <li>• Luminanz, A, B</li> </ul>	Wert für Farbparameter	0 ... 7	Signed Integer	
Farbabstand	Abstand der aktuellen Farbe gegenüber der eingelernten Farbe	0 ...7	Signed Integer	






## Zählen / Anzahl







◀PAYLOAD▶ Zählen / Anzahl				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Anzahl Objekte	Anzahl der gefundenen Objekte [Stück]	1 ... 5	Signed Integer	
Anzahl gültiger Objekte	Anzahl der gefundenen gültigen Objekte [ Stück]	1 ... 5	Signed Integer	






«PAYLOAD» Zählen / Anzahl				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Anzahl Suchstreifen	Anzahl der parallelen Suchstreifen in die, die Breite des Suchbereichs aufgeteilt wird. [Stück]	1 ... 5	Signed Integer	(nur Kantenantastung) 
Anzahl gültige Suchstreifen	Anzahl der Suchstreifen welche zur Ergebnisbildung verwendet werden [Stück]	1 ... 3	Signed Integer	(nur Kantenantastung) 
Ergebnisvektor	Vektor, der das Ergebnis (1/0) der gefundenen Instanzen enthält			
Zu viele BLOBs		1	BOOL	

### Erweitert

«PAYLOAD» Erweitert				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Skalierung	Aktueller Skalierungsfaktor zur eingelernten Referenz.. 1/1000 (Faktor). Wertebereich Faktor 0,5 - 2.	3 ... 4	Unsigned Int	(nur Konturvergleich) 
Exzentrizität	Numerische Exzentrizität Wertebereich 0,0 ... 1,0	n	Signed Integer	

«PAYLOAD» Erweitert				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Sicherheit	Ausgabe der Sicherheitswerte der einzelnen Zeichen. Der Sicherheitswert gibt an, wie zuverlässig der Leser ein Zeichen bewerten konnte. Wertebereich von 0 ... 100 [%]	n	Unsigned Int	<b>A</b>
Referenzstring getroffen	Übereinstimmung des ausgegebenen Strings mit dem Referenzstring.	1	BOOL	<b>A</b>
Kontrast	Kontrast des Codes Wertebereich von 0 - 100 [%]	n	Unsigned Int	
Korrektur	Anzahl der durch die Fehlerkorrekturen korrigierte Module [Stück]	n	Unsigned Int	
Konturlänge	Anzahl der Pixel der äußeren Kontur, 1/1000 [Pixel]	n	Signed Integer	
Kompaktheit	Kompaktheit des BLOBS, (Kreis = 1, andere >1) Je stärker die Form des BLOB vom Kreis abweicht, desto größer wird der Wert der Kompaktheit.	n	Signed Integer	
Schwerpunkt X	X-Koordinate des Schwerpunkts, 1/1000	n	Signed Integer	

«PAYLOAD» Erweitert				
Wert	Beschreibung	Länge ASCII [Byte]	Datentyp	Verfügbar für
Schwerpunkt Y	Y-Koordinate des Schwerpunkts, 1/1000	n	Signed Integer	
Grauwert, mittlerer	Mittlerer Grauwert aller Pixel, die zum BLOB gehören.	n	Signed Integer	
Schaltsschwelle min.	Untere Schaltschwelle für die Binarisierung der Objekte. 0...255	1 ... 3	Unsigned Int	
Schaltsschwelle max.	Obere Schaltschwelle für die Binarisierung der Objekte. 0...255	1 ... 3	Unsigned Int	
Schaltsschwelle invertiert	Gibt an, ob der Bereich Min <-> Max invertiert ist. P: invertiert F: nicht invertiert	1	Unsigned Char	
Abweichung, innen	Liefert die größte Abweichung zwischen der BLOB-Kontur und der Kontur des geometrischen Elements (Abweichung in den gefitteten Kreis hinein). [Kundeneinheit / 1000]	1 ... 7	Signed Integer	

<b>«PAYLOAD» Erweitert</b>				
<b>Wert</b>	<b>Beschreibung</b>	<b>Länge ASCII [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Abweichung, außen	Liefert den größten Abstand zwischen der BLOB-Kontur und der Kontur der geometrischen Form (Abweichung aus dem gefitteten Kreis heraus). [Kundeneinheit]	1 ... 7	Signed Integer	
Abweichung, mittlere	Liefert den Mittelwert des Betrags aller Innen- und Außenabweichungen zwischen der BLOB-Kontur und der Kontur der geometrischen Form.	1 ... 7	Signed Integer	
Achsenverhältnis	Verhältnis der langen zur kurzen Achse (a/b)	1 ... 7	Signed Integer	
Bauch / Rücken, Fläche	Bauch- / Rücken-Lage, Basis: Fläche, Unterscheidung der Lage durch Vorzeichen, 1/1000	n	Signed Integer	
Ergebnis Index	Listenindex	n	Signed Integer	
Suchstreifen Distanz	Berechneter Abstand [Kundeneinheit] / 1000 pro Suchstreifen-Paar	1 ... 11	Signed Integer	<b>F</b>

<b>&lt;CHKSUM&gt;</b>				
<b>Parameter</b>	<b>Beschreibung</b>	<b>Länge ASCII [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Prüfsumme	XOR-Prüfsumme über alle Bytes im Telegramm. Wird als letztes Byte übertragen.	1	Unsigned Int	Alle Typen

<b>&lt;TRAILER&gt;</b>				
<b>Parameter</b>	<b>Beschreibung</b>	<b>Länge ASCII [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Nachspann	Benutzerdefiniert bis max. 8 Zeichen	0 ... 8	Unsigned Int	Alle Typen

**\*HINWEIS:**

Wenn keine Kalibrierung ausgeführt wurde, beziehen sich alle Werte auf Pixel.

\*\* Detektor Messschieber: Abhängig von der gewählten Abstandsart. "Kleinster / Größter gegenüberliegender Abstand" = Vektor mit zwei Elementen [min; max].

## 11.5 Beschreibung Telegramme BINÄR

### 11.5.1 Allgemein

#### Statistik zurücksetzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Statistik zurücksetzen (RST) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x04	Statistik zurücksetzen
Statistik zurücksetzen (RST) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x04	Statistik zurücksetzen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Low
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>



#### HINWEIS:

Statistik-Werte können ausgegeben werden im Bedienschnitt Ausgabe / Reiter Datenausgabe / "Detektorspezifische Nutzdaten", Auswahl "GENERAL".

Weitere Informationen: siehe Datenausgabe BINÄR / [Allgemein](#)

#### Bild aus Datei verarbeiten (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)



<b>Bild aus Datei verarbeiten (PIF) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int		Telegrammlänge 9 Byte + Länge nachfolgender Daten (n)
5	Unsigned Char	0x2D	Bild aus Datei verarbeiten
6	Unsigned Char		Version des Request
7	Unsigned Char	0x00	Reserviert
8	Unsigned Char	0x01	Reserviert
9	Unsigned Char	0xXX	Länge des darauffolgenden Dateinamens der Datei (Tatsächliche Byte-Anzahl mit Dateierweiterung, max. 255 Bytes)
10 ... n	Unsigned Char	0xXX	Dateiname (UTF-8) und Format verfügbar auf dem Gerät im Verzeichnis "/tmp/". Erlaubte Erweiterungen: Monochromer Sensor: .pgm Farbsensor: .ppm (RGB) or .pgm (Bayer)
<b>Bild aus Datei verarbeiten (PIF) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge
5	Unsigned Char	0x2D	Bild aus Datei verarbeiten
6-7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0x00	Reserviert
9-12	Unsigned Int	0xXX	Länge der impliziten Ergebnisausgabe
13 ... n	Unsigned Char	0xXX	Ausgabe des impliziten Ergebnis
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja

Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>



**HINWEIS:**

Bildgröße des Testbilds muss übereinstimmen mit der Bildgröße des derzeit aktiven Jobs auf dem Gerät.

## 11.5.2 Kontrolle

### Trigger (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Trigger (TRG) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x01	Trigger, (einfacher Trigger ohne Index, via Port 2006)
<b>Trigger (TRG) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x01	Trigger, (Response auf Kommando Trigger ohne Index, via Port 2006. Falls definiert: Ergebnisdaten ohne Index via Port 2005)
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Ja	
Akzeptiert wenn Ready Low:		Nein	
Status des Signals Ready während Bearbeitung:		Low	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Erweiterter Trigger (BINÄR)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

### Übersicht Telegramme

<b>Erweiterter Trigger (TRX) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge 6 Byte + Länge nachfolgender Daten (n)
5	Unsigned Char	0x13	Erweiterter Trigger, Trigger mit Index, zur Zuordnung Trigger zu entsprechenden Ergebnisdaten, via Port 2006)
6	Unsigned Char	0xXX	Länge nachfolgender Daten (0-99)
7 ... n	Unsigned Char	0xXX	Daten
<b>Erweiterter Trigger (TRX) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge
5	Unsigned Char	0x13	Erweiterter Trigger, (Response auf Trigger mit Index und Ergebnisdaten, via Port 2006, von Zuordnung von Trigger zum Ergebnis, Ergebnisdaten außerdem ohne Index via Port 2005)
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	Länge nachfolgender Daten (n)
9 ... n	Unsigned Char	0xXX	Daten aus dem Sendebefehl
n+1	Unsigned Char	0xXX	Betriebsmodus 0 = Config Mode 1 = Run Mode
n + 2 ... n + 5	Unsigned Int	0xXX	Länge der Ergebnisdaten
n + 6 ... m	Unsigned Char	0xXX	Ergebnisdaten
Weitere Informationen:			
Akzeptiert im Runmode:			Ja

Akzeptiert im Konfigurationsmodus:	Ja
Akzeptiert wenn Ready Low:	Nein
Status des Signals Ready während Bearbeitung:	Low
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Trigger Robotics (BINÄR)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme

<b>Trigger Robotics (TRR) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge 31 (0x1F) + Länge des Trigger-Identifiers in Byte
5	Unsigned Char	0x37	Trigger Robotics
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	Länge des Trigger-Identifiers in Byte
8 ... n	Unsigned Char	0xXX	Trigger Identifier
n+1...n+4	Unsigned Int	0xXX	Pose_TCP Pos. X (in Kundeneinheit * 1000)
n+5...n+8	Unsigned Int	0xXX	Pose_TCP Pos. Y (in Kundeneinheit * 1000)
n+9...n+12	Unsigned Int	0xXX	Pose_TCP Pos. Z (in Kundeneinheit * 1000)
n+13...n+16	Unsigned Int	0xXX	Pose_TCP Winkel X (in Grad * 1000)
n+17...n+20	Unsigned Int	0xXX	Pose_TCP Winkel Y (in Grad * 1000)
n+20...n+24	Unsigned Int	0xXX	Pose_TCP Winkel Z (in Grad * 1000)
<b>Trigger Robotics (TRR) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge 8 (0x08) + Länge des Trigger-Identifiers in Byte
5	Unsigned Char	0x37	Trigger Robotics, (Response auf Kommando Trigger ohne Index, via Port 2006. Falls definiert: Ergebnisdaten ohne Index via Port 2005)
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>

8	Unsigned Char	0xXX	Länge des Trigger-Identifiers
9 ... n	Unsigned Char	0xXX	Trigger Identifier
n+1	Unsigned Char	0xXX	Operation Mode 0x00 = Config 0x01 = Run
n+2...n+5	Unsigned Int	0xXX	Länge der Ergebnisdaten in Byte
n+6...m	Unsigned Int	0xXX	Ergebnisdaten
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Ja	
Akzeptiert wenn Ready Low:		Nein	

Hinweis: Für "Kalibrierplatte (Roboter)" und "Punktpaarliste (Roboter)" werden nur die X- und Y-Position berücksichtigt. Die anderen Werte (Position Z und Rotationen) müssen 0 sein.

## Trigger ID setzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Trigger ID setzen (STI) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge 7 Byte + Länge der Trigger ID
5	Unsigned Char	0x2E	Trigger ID setzen
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	Länge der nachfolgenden Daten (max 99)
8 ... n	Unsigned Char	0xXX	Trigger ID
Beispiel:	0x00 0x00 0x00 0x0D 0x2E 0x01 0x06 0x30 0x31 0x32 0x33 0x34 0x35		
Trigger ID setzen (STI) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x2E	Trigger ID setzen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Beispiel:	0x00 0x00 0x00 0x07 0x2E 0x00 0x00		
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Ja
Akzeptiert wenn Ready Low:			Ja



## Jobwechsel (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Jobwechsel (CJB) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x06	Telegrammlänge
5	Unsigned Char	0x02	Change Job
6	Unsigned Char	0xXX	Job Nr. XX = 1 ... n
<b>Jobwechsel (CJB) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x09	Telegrammlänge
5	Unsigned Char	0x02	Change Job
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	Triggermodus 0x00: Trigger 0x01: Freilauf
9	Unsigned Char	0xXX	Job Nr. XX = 1 ... n
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Low
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>



#### **HINWEIS:**

Tritt ein Fehler beim Jobwechsel auf, besteht die Möglichkeit in Job 1 zu wechseln.

## Jobwechsel Permanent (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Jobwechsel Permanent (CJP) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x06	Telegrammlänge
5	Unsigned Char	0x22	Jobwechsel Permanent
6	Unsigned Char	0xXX	Job Nr. XX = 1 ... n
<b>Jobwechsel Permanent (CJP) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x09	Telegrammlänge
5	Unsigned Char	0x22	Jobwechsel Permanent
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	Trigger Mode 0x00: Trigger 0x01: Freilauf
9	Unsigned Char	0xXX	Job Nr. XX = 1 ... n
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Low	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	



#### **HINWEIS:**

Tritt ein Fehler beim Jobwechsel auf, besteht die Möglichkeit in Job 1 zu wechseln.

## Jobwechsel nach Jobname (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Jobwechsel nach Jobname (CJN) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge 7 Byte + Länge Jobname (n)
5	Unsigned Char	0x2C	Jobwechsel nach Jobname
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	Länge Jobname (n)
8 ... n	Unsigned Char	0xXX	Jobname
<b>Jobwechsel nach Jobname (CJN) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x08	Telegrammlänge
5	Unsigned Char	0x2C	Jobwechsel nach Jobname
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	Triggermodus 0x00: Trigger 0x01: Freilauf
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Low	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## 11.5.3 Jobeinstellungen

### Auto Arbeitsabstand (BINÄR)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme

Auto Arbeitsabstand (AFC) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge 11 Byte (0x0B) + gewählte Optionen 8 Byte (0x08)
5	Unsigned Char	0x32	Auto Arbeitsabstand
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
8	Unsigned Char	0xXX	Schrittweite der Suche (0x01 - 0x05)
9	Unsigned Char	0xXX	Auswahl Abstandswert 0x00: Höchster Score 0x01: Min. Arbeitsabstand 0x02: Max. Arbeitsabstand 0x03: Mittelwert Arbeitsabstand 0x04: Median Arbeitsabstand 0x05: Höchster Score - Ausgabe aller gefundenen Arbeitsabstände
10	Unsigned Char	0xXX	Einheit 0x00: 1/1000 Millimeter (µm) 0x01: Motorschritte
11	Unsigned Char	0xXX	Auswahl Suchbereich 0x00: Gesamter Bereich 0x01: Gewählter Bereich
12...15	Unsigned Int	X	Suchbereich Anfang (nur wenn Auswahl Suchbereich == 0x01)
16...19	Unsigned Int	X	Suchbereich Ende (nur wenn Auswahl Suchbereich == 0x01)
Auto Arbeitsabstand (AFC) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung

1 - 4	Unsigned Int	0xXX	Telegrammlänge 11 Byte (0x0B) + Arbeitsabstände + Scorewerte
5	Unsigned Char	0x32	Auto Arbeitsabstand
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	Unsigned Int	X	Anzahl ausgegebener Arbeitsabstände
12 ... n	Unsigned Int	X	Abstandswert in 1/1000 mm oder Motorschritten (pro ausgegebenem Arbeitsabstand 4 Byte)
n-m	Unsigned Int	X	Scorewert zu Abstandswert multipliziert mit 1000 (pro ausgegebenem Arbeitsabstand 4 Byte)
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Arbeitsabstand setzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Arbeitsabstand setzen (SFC) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0D	Telegrammlänge
5	Unsigned Char	0x31	Arbeitsabstand setzen
6	Unsigned Char	0xX1	Version des Request
7	Unsigned Char	0xXX	0: Temporär 1: Permanent
8	Unsigned Char	0xXX	Bewegung 0: Absolut 1: Relativ 2: Absolut mit Neuinitialisierung
9	Unsigned Char	0xXX	Einheit 0: 1/1000 Millimeter 4: Schritte
10 - 13	Signed Integer	0xXX	Arbeitsabstand
<b>Arbeitsabstand setzen (SFC) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0B	Telegrammlänge
5	Unsigned Char	0x31	Arbeitsabstand setzen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	INT	0xXX	Aktueller Arbeitsabstand
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung

Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
------------------------------	---

## Arbeitsabstand lesen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Arbeitsabstand lesen (GFC) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x30	Arbeitsabstand lesen
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	Einheit 0x00: 1/1000 Millimeter 0x04: Schritte
Arbeitsabstand lesen (GFC) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0B	Telegrammlänge
5	Unsigned Char	0x30	Arbeitsabstand lesen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	INT	0xXX	Aktueller Arbeitsabstand
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	



## Auto Verschlusszeit (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Auto Verschlusszeit (ASH) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x07	Auto Verschlusszeit
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
Auto Verschlusszeit (ASH) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0F	Telegrammlänge
5	Unsigned Char	0x07	Auto Verschlusszeit
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	INT	0xXX	Auto Verschlusszeit-Wert
12 - 15	INT	0xXX	Score
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Low
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Verschlusszeitwert setzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Verschlusszeit setzen (SSP/SST) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x09	Telegrammlänge
5	Unsigned Char	0xXX	0x0E Verschlusszeit setzen temporär 0x0F Verschlusszeit setzen permanent
6 - 9	Unsigned Int	0xXX	Verschlusszeitwert in 1/1000 ms
<b>Verschlusszeit setzen (SSP/SST) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0xXX	0x0E Verschlusszeit setzen temporär 0x0F Verschlusszeit setzen permanent
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Low
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Verschlusszeitwert lesen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Verschlusszeitwert lesen (GSH) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x17	Verschlusszeitwert lesen
<b>Verschlusszeitwert lesen (GSH) Response String vom Sensor (BINÄR)</b>			
1 - 4	Unsigned Int	0x0B	Telegrammlänge
5	Unsigned Char	0x17	Verschlusszeitwert lesen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	Unsigned Int	0xXX	Verschlusszeitwert
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus::			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Verstärkungswert setzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Verstärkungswert setzen (SGA) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0A	Telegrammlänge
5	Unsigned Char	0x1B	Verstärkungswert setzen
6	Unsigned Char	0xXX	0: Temporär 1: Permanent
7 - 10	Unsigned Int	0xXX	Verstärkungswert * 1000
Verstärkungswert setzen (SGA) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0B	Telegrammlänge
5	Unsigned Char	0x1B	Verstärkungswert setzen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	Unsigned Int	0xXX	Aktueller Verstärkungswert (Verstärkungswert *1000)
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Verstärkungswert lesen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Verstärkungswert lesen (GGA) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x1C	Verstärkungswert lesen
Verstärkungswert lesen (GGA) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0B	Telegrammlänge
5	Unsigned Char	0x1C	Verstärkungswert lesen
6	Unsigned	0xFF	<a href="#">Fehlercodes</a>
7	Short	0xFF	
8 - 11	Unsigned Int	0xFF	Aktueller Verstärkungswert * 1000
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Parameter setzen (BINÄR)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme

Parameter setzen (SPP/SPT) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge = 9 Bytes + Länge des gewählten Parameters
5	Unsigned Char	0xXX	0x05: Parameter permanent setzen 0x06: Parameter temporär setzen
6	Unsigned Char	0xXX	Detektor Nr., XX = 1 ... n
7	Unsigned Char	0xXX	Kommando Referenzstring / Wert setzen, siehe Tabelle <a href="#">Übersicht Detektor-Parameter</a>
8 - 9	Unsigned Short	0xXX	Länge neuer Referenzstring / Wert (n), siehe Tabelle <a href="#">Übersicht Detektor-Parameter</a>
10 ... n	Unsigned Char	0xXX	Referenzstring / Wert
Parameter setzen (SPP/SPT) Response String vom Sensor (BINÄR)			
(kann bis zu 4-5 Sekunden verzögert ankommen)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x08	Telegrammlänge + Länge des gewählten Parameters in Byte
5	Unsigned Char	0xXX	0x05: Parameter permanent setzen 0x06: Parameter temporär setzen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>

8	Unsigned Char	0xXX	Parameter-Typ 0x00: I8 0x01: U8 0x02: I16 0x03: U16 0x04: I32 0x05: U32 0x06: I40 0x07: U40 0x08: Float 0x09: Double 0x0A: String 0x0B: Boolean 0x0C: Special signed8 0x0D: Undefined
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Low	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Übersicht Detektor-Parameter (setzen / lesen)

Detektor	Funktion	Wert	Multiplikator	Länge
<b>Lagenachführung</b>				
Mustervergleich Konturvergleich	Schwellenwert Min.	0x01	1000	4
	Schwellenwert Max.	0x02	1000	4
	Ergebnisoffset 0: "Aus" 1: "Bildebene (in Pixel)" 2: "Positionierung (2D)" 3: "Roboter (3D)"	0x1E	1	1
	Ergebnisoffset Bildebene: Pos. X	0x1F	1000	4
	Ergebnisoffset Bildebene: Pos. Y	0x20	1000	4
	Ergebnisoffset Bildebene: Winkel	0x21	1000	4
	Ergebnisoffset Positionierung (2D), Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z	0x22	1000	24 (6 * 4 Byte je Wert)
	Ergebnisoffset berechnen* mit übertragener Position  <ul style="list-style-type: none"> <li>• Positionierung (2D): Pos. X, Pos. Y, 0, 0, 0, Winkel Z</li> <li>• Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul> *Es muss eine gültige Posi- tion für den Detektor vor- handen sein	0x23	1000	24 (6 * 4 Byte je Wert)
Kantenantastung	Antaster 1: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	0x65	1	1
	Antaster 2: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	0x66	1	1



Detektor	Funktion	Wert	Multiplikator	Länge
	Antaster 3: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	0x67	1	1
	Antaster 1: Schwellenwert Min.	0x68	1000	4
	Antaster 2: Schwellenwert Min.	0x69	1000	4
	Antaster 3: Schwellenwert Min.	0x6A	1000	4
<b>Detektoren</b>				
Mustervergleich Kontur Kontur 3D	Schwellenwert Min.	0x01	1000	4
	Schwellenwert Max.	0x02	1000	4
	Ergebnisoffset 0: "Aus" 1: "Bildebene (in Pixel)" 2: "Positionierung (2D)" 3: "Roboter (3D)"	0x1E	1	1
	Ergebnisoffset Bildebene: Pos. X	0x1F	1000	4
	Ergebnisoffset Bildebene: Pos. Y	0x20	1000	4
	Ergebnisoffset Bildebene: Winkel	0x21	1000	4
	Ergebnisoffset Positionierung (2D), Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z	0x22	1000	24 (6 * 4 Byte je Wert)

Detektor	Funktion	Wert	Multiplikator	Länge
	Ergebnisoffset berechnen* mit übertragener Position <ul style="list-style-type: none"> <li>• Positionierung (2D): Pos. X, Pos. Y, 0, 0, 0, Winkel Z</li> <li>• Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul> *Es muss eine gültige Position für den Detektor vorhanden sein	0x23	1000	24 (6 * 4 Byte je Wert)
Zielmarke 3D	Ergebnisoffset <ul style="list-style-type: none"> <li>• Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul>	0x22	1000	24 (6 * 4 Byte je Wert)
	Ergebnisoffset berechnen* mit übertragener Position <ul style="list-style-type: none"> <li>• Roboter (3D): Pos. X, Pos. Y, Pos. Z, Winkel X, Winkel Y, Winkel Z</li> </ul> *Es muss eine gültige Position für den Detektor vorhanden sein	0x23	1000	24 (6 * 4 Byte je Wert)
	ID der aktiven Zielmarke	0x65	-	n
	Zielmarkenname	0x66	-	n
	Aktuelle Zielmarke zur Liste der Zielmarken hinzufügen (Parameter kann nur gesetzt werden!) Es muss eine Zielmarke detektiert worden sein.	0x67	-	n
Graustufe	Schwellenwert Min.	0x01	1000	4
	Schwellenwert Max.	0x02	1000	4
	Graustufenwert Min.	0x65	1000	4
	Graustufenwert Max.	0x66	1000	4

Detektor	Funktion	Wert	Multiplikator	Länge
	Graustufenwert invertieren	0x67	1	4
Kontrast Helligkeit	Schwellenwert Min.	0x01	1000	4
	Schwellenwert Max.	0x02	1000	4
Messschieber	Schwellenwert Abstand Min.	0x65	1000	4
	Schwellenwert Abstand Max.	0x66	1000	4
	Schwellenwert Abstand invertieren	0x67	1	1
	Abstandsart 0: Minimum 1: Maximum 2: Mittlerer 3: Median 4: Kleinster Gegenüberliegender 5: Größter Gegenüberliegender	0x68	1	1
	Antaster 1: Schwellenwert Min.	0x69	1000	4
	Antaster 2: Schwellenwert Min.	0x6A	1000	4
	Antaster 1: Glättung	0x6B	1000	4
	Antaster 2: Glättung	0x6C	1000	4
	Antaster 1: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	0x6D	1	1
	Antaster 2: Übergang 0: Beliebig 1: Dunkel nach hell 2: Hell nach dunkel	0x6E	1	1
	Antaster 1: Anzahl Suchstreifen	0x6F	1	1
Antaster 2: Anzahl Suchstreifen	0x70	1	4	
BLOB	Graustufenwert Min.	0x65	1000	4
	Graustufenwert Max.	0x66	1000	4

Detektor	Funktion	Wert	Multiplikator	Länge
	Graustufenwert invertieren 0: nicht invertiert 1: invertiert	0x67	1	1
	Schwellenwert Anzahl BLOB Min.	0x78	1	1
	Schwellenwert Anzahl BLOB Max.	0x79	1	1
	Schwellenwert Anzahl invertieren 0: nicht invertiert 1: invertiert	0x7A	1	1
	Anzahl gesetzter Merkmale (nur lesend)	0x7B	1	1
	Auswahl Merkmal aus Liste	0x7C	1	1
	Schwellenwert Merkmal Min.	0x7D	1000	4
	Schwellenwert Merkmal Max.	0x7E	1000	4
	Schwellenwert Merkmal invertieren	0x7F	1	1
Barcode Datacode OCR	Referenzstring	0x65	-	n (Länge des Strings)
	Referenzstring	0x65	-	n (Länge des Strings)
	Referenzstring	0x65	-	n (Länge des Strings)
Farbwert Farbwert	Farbraum (nur lesend)	0x15	0x00 = RGB 0x01 = HSV 0x02 = LAB	1
	Kanalauswahl (nur lesend)	0x16	Bitfeld jeweils eine Stelle pro Farbkanal	1
	Farbkanal 1: Schwellenwert Min.	0x65	1000	4
	Farbkanal 1: Schwellenwert Max.	0x66	1000	4

Detektor	Funktion	Wert	Multiplikator	Länge
	Farbkanal 1: Schwellenwert invertieren	0x67	1	1
	Farbkanal 2: Schwellenwert Min.	0x68	1000	4
	Farbkanal 2: Schwellenwert Max.	0x69	1000	4
	Farbkanal 2: Schwellenwert invertieren	0x6A	1	1
	Farbkanal 3: Schwellenwert Min.	0x6B	1000	4
	Farbkanal 3: Schwellenwert Max.	0x6C	1000	4
	Farbkanal 3: Schwellenwert invertieren	0x6D	1	1
Farbliste	Farbraum (nur lesend)	0x15	0 = RGB 1 = HSV 2 = LAB	3
	Kanalauswahl (nur lesend)	0x16	Bitfeld jeweils eine Stelle pro Farbkanal	4
	Schwellenwert Farbabstand	0x65	1000	n
	Schwellenwert Farbabstand aktiv setzen	0x66	1	n
	Anzahl Farben in Liste	0x67	1	n
	Auswahl Farbe aus Liste	0x68	1	n
	Farbwert der ausgewählten Farbe (Farbkanal 1, Farbkanal 2, Farbkanal 3, Farbkanal 4 [konstant 0])	0x69	1000	32
Busbar Wafer	Schwellenwert Min.	0x01	1000	n
	Schwellenwert Max.	0x02	1000	n
Ergebnisverarbeitung	Name des aktiven Ausdrucks	0x7A	-	n (Länge des Strings)
	Aktueller Ausdruck	0x7V	-	n (Länge des Strings)

## Parameter lesen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Parameter lesen (GPA) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x0A	Parameter lesen
6	Unsigned Char	0xXX	Detektor Nr., XX = 1... n
7	Unsigned Char	0xXX	Kommando Referenzstring / Wert lesen, siehe Tabelle <a href="#">Übersicht Detektor-Parameter</a>
<b>Parameter lesen (GPA) Response String vom Sensor (BINÄR)</b>			
(kann bis zu 4-5 Sekunden verzögert ankommen)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge = 10 Bytes + Länge des gewählten Parameters in Byte
5	Unsigned Char	0x0A	Parameter lesen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	Parameter Typ String
9 - 10	Unsigned Short	0xXX	Länge gelesener Wert (n)
11 ... n + n	Unsigned Char	0xXX	Referenzstring / Wert
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Suchbereich (ROI) setzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>ROI setzen (SRP/SRT) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge in Byte 24 Byte: Kreis 32 Byte: Rechteck, Ellipse, Freiform
5	Unsigned Char	0xXX	0x10: Parameter temporär setzen 0x11: Parameter permanent setzen
6 - 9	Unsigned Int	0xXX	19 Byte: Kreis 27 Byte: Rechteck, Elipse, Freiform
10	Unsigned Char	0xXX	Detektor Nr.
11	Unsigned Char	0xXX	Suchbereich (ROI) Typ 0x00: Suchbereich (gelb) 0x01: Teach-Bereich (rot) 0x02: Positionskontrolle (blau)
12	Unsigned Char	0xXX	Suchbereich (ROI) Form 0x01: Kreis 0x02: Rechteck 0x03: Ellipse 0x04: Freiform
13 - 16	Unsigned Int	0xXX	ROI Parameter: Mittelpunkt X (Wert in [px] * 1000)
17 - 20	Unsigned Int	0xXX	ROI Parameter: Mittelpunkt Y (Wert in [px] * 1000)
21 - 24	Unsigned Int	0xXX	ROI Parameter: halbe Breite bzw. Radius X (Wert in [px] * 1000)
Nur bei Ellipse / Rechteck / Freiform:			
25 - 28	Unsigned Int	0xXX	ROI Parameter: halbe Höhe bzw. Radius Y (Wert in Pixeln * 1000)
29 - 32	Unsigned Int	0xXX	ROI Parameter: Winkel in ° Grad (Wert in ° [Grad] * 1000)
<b>ROI setzen (SRP/SRT) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0xXX	0x10: Parameter permanent setzen 0x11: Parameter temporär setzen

6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Low	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	
Parameter:		Die Parameter werden im Koordinatensystem der Lagenachführung und nicht im Koordinatensystem des Bildes angegeben.	



## Suchbereich (ROI) lesen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>ROI lesen (GRI) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x12	Get ROI
6	Unsigned Char	0xXX	Detektor Nr.
7	Unsigned Char	0xXX	Suchbereich (ROI) Typ 0x00: Suchbereich (gelb) 0x01: Teach-Bereich (rot) 0x02: Positionskontrolle (blau)
<b>ROI lesen (GRI) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge
5	Unsigned Char	0x12	Get ROI
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	Unsigned Int	0xXX	Suchbereich (ROI) Info Länge in Bytes ab Byte 8
12	Unsigned Char	0xXX	Detektor Nr.
13	Unsigned Char	0xXX	Suchbereich (ROI) Typ 0x00: Suchbereich (gelb) 0x01: Teach-Bereich (rot) 0x02: Positionskontrolle (blau)
14	Unsigned Char	0xXX	Suchbereich (ROI) Form 0x01: Kreis 0x02: Rechteck 0x03: Ellipse 0x04: Freiform
15 - 18	Unsigned Int	0xXX	ROI Parameter: Mittelpunkt X (Wert in Pixeln * 1000)
19 - 22	Unsigned Int	0xXX	ROI Parameter: Mittelpunkt Y (Wert in Pixeln * 1000)
23 - 26	Unsigned Int	0xXX	ROI Parameter: halbe Breite bzw. Radius X (Wert in Pixeln * 1000)

	Nur bei Ellipse / Rechteck / Freiform:		
27 - 30	Unsigned Int	0XXX	ROI Parameter: halbe Höhe bzw. Radius Y (Wert in Pixeln * 1000)
31 - 34	Unsigned Int	0XXX	ROI Parameter: Winkel in ° (Wert in ° * 1000)
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Low	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Suchbereichsinhalt setzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Suchbereichsinhalt setzen (SRC) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge 11 Byte + Länge nachfolgender Daten (n)
5	Unsigned Char	0x39	Suchbereichsinhalt setzen (Set ROI Content)
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Parameter temporär setzen 0x01: Parameter permanent setzen
8	Unsigned Char	0xXX	Detektor Nr. 0: Lagenachführung x0: Standarddetektor im Job
9	Unsigned Char	0x00	Reserviert
10	Unsigned Char	0xXX	Bit 0: Maskierungsdatei verwenden Bit 1 - 7: reserviert
11	Unsigned Char	0xXX	Länge des darauffolgenden Dateinamens der Datei (Tatsächliche Byte-Anzahl mit Dateierweiterung, max. 255 Bytes) Oder für 000: Standardname mask.pgm
12 ... n	Char	0xXX	Dateiname (UTF-8) und Format verfügbar auf dem Gerät im Verzeichnis "/tmp/". Standardname mask.pgm Dateiformat: PGM
Suchbereichsinhalt setzen (SRC) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x39	Suchbereichsinhalt setzen (Set ROI Content)
6-7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:			Ja

Akzeptiert im Konfigurationsmodus:	Nein
Akzeptiert wenn Ready Low:	Ja
Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

**Suchbereichsinhalt lesen (BINÄR)**

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme


<b>Suchbereichsinhalt lesen (GRC) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge 10 Byte + Länge nachfolgender Daten (n)
5	Unsigned Char	0x3A	Suchbereichsinhalt lesen (Get ROI Content)
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	Detektor Nr. 0: Lagenachführung x0: Standarddetektor im Job
8	Unsigned Char	0x00	Reserviert
9	Unsigned Char	0xXX	Bit 0: Maskierungsdatei verwenden (sofern vorhanden) Bit 1: Musterdatei verwenden Bit 2: Konturdatei verwenden Bit 3-7: reserviert
10	Unsigned Char	0xXX	Länge des darauffolgenden Dateinamens der Datei (Tatsächliche Byte-Anzahl mit Dateierweiterung, max. 255 Bytes) Oder für 000: Standardname mask.pgm / pattern.pgm / contour.pgm
11 ... n	Char		Dateiname (UTF-8) und Format verfügbar auf dem Gerät im Verzeichnis " /tmp/". Standardname mask.pgm / pattern.pgm / contour.pgm Dateiformat: PGM
<b>Suchbereichsinhalt lesen (GRC) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0B	Telegrammlänge
5	Unsigned Char	0x3A	Suchbereichsinhalt setzen (Set ROI Content)
6-7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>

8-9	Unsigned Short	0xXX	Breite des Suchbereichs
10-11	Unsigned Short	0xXX	Höhe des Suchbereichs
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Jobliste lesen (BINÄR)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme

<b>Jobliste lesen (GJL) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x14	Jobliste lesen
<b>Jobliste lesen (GJL) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge
5	Unsigned Char	0x14	Jobliste lesen
6	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0x01	Konstant
9	Unsigned Char	0xXX	Anzahl der Jobs
10	Unsigned Char	0xXX	Aktive Job Nummer
	<b>HINWEIS:</b> Folgende Bytefolge wird für jeden Job von 1 bis "Anzahl der Jobs" wiederholt. Die Byte Nummern verschieben sich entsprechend.		
11	Unsigned Char	0xXX	Anzahl der folgenden Bytes. Damit kann ein eindeutiger Name für den Job n angegeben werden.
11 ... n	Char	0xXX	Ab dieser Position folgt, in der angegebenen Länge, der Name für Job n.
n + 1 ... n + 3	Unsigned Char	0xXX	Anzahl der folgenden Bytes. Damit kann eine Beschreibung für den Job n angegeben werden.
n + 4 ... m	Char	0xXX	Ab dieser Position folgt, in der angegebenen Länge, die Beschreibung für Job 1.
m + 1 ... m + 3	Unsigned Char	0xXX	Anzahl der folgenden Bytes. Damit kann ein eindeutiger Name für den Autor des Job n angegeben werden.


m + 4 ... k	Char	0xXX	Ab dieser Position folgt, in der angegebenen Länge, der Name für den Autor aus Job n.
k + 1 ... k + 7	Unsigned Int	0xXX	Datum der Erstellung von Job n (7 Byte)
k + 8 ... k + 14	Unsigned Int	0xXX	Datum der letzten Änderung von Job n (7 Byte)
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	



## Detektorliste lesen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Detektorliste lesen (GDL) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x15	Detektorliste lesen
Detektorliste lesen (GDL) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge
5	Unsigned Char	0x18	Detektorliste lesen
6	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	Jobnummer des aktuellen Jobs
9	Unsigned Char	0xXX	Anzahl der Detektoren im aktuellen Job
		<b>HINWEIS:</b>	
		Folgende Bytefolge wird für jeden Detektor im Job wiederholt. Die Byte Nummern verschieben sich entsprechend.	
10	Unsigned Char	0xXX	Anzahl der folgenden Bytes. Damit kann ein eindeutiger Name für den Detektor n angegeben werden.
11 ... n	Unsigned Char	0xXX	Ab dieser Position folgt, in der angegebenen Länge, der Name für Detektor n.

n + 1 ... n + 2	Unsigned Char	0xXX	Detektor 0x01: Mustervergleich 0x04: Kontur 0x05: Graustufe 0x06: Kontrast 0x07: Helligkeit 0x0B: OCR 0x0D: Datacode 0x0E: Barcode 0x12: Farbwert 0x13: Farbfläche 0x14: Farbliste 0x15: Messschieber 0x16: BLOB 0x18: Kontur 3D 0x1B: Ergebnisverarbeitung 0x1C: Zielmarke 3D
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Detektor einlernen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Detektor einlernen (TED) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x08	Telegrammlänge
5	Unsigned Char	0x18	Detektor einlernen
6	Unsigned Char	0xXX	0x00: Lagenachführung ≥ 0x01: Auswahl Detektor
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
8	Unsigned Char	0xXX	0x00: Kein Trigger, Einlernen mit nächster Bildaufnahme 0x01: Trigger wird ausgeführt zum Einlernen
Detektor einlernen (TED) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x00	Telegrammlänge
5	Unsigned Char	0x18	Detektor einlernen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Trigger-Verzögerung setzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Trigger-Verzögerung setzen (STD) Request String an Sensor (BINÄR)</b>			
Byte Nr..	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x08	Telegrammlänge
5	Unsigned Char	0x27	Trigger-Verzögerung setzen
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
8 - 11	Unsigned Int	0xXX	Trigger-Verzögerung in msec (max. 3000 msec) in Encoder-Schritten (max. 65535 Schritte)
<b>Trigger-Verzögerung setzen (STD) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x27	Trigger-Verzögerung setzen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Low	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Trigger-Verzögerung lesen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Trigger-Verzögerung lesen (GTD) Request String an Sensor (BINÄR)</b>			
Byte Nr..	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x06	Telegrammlänge
5	Unsigned Char	0x28	Trigger-Verzögerung lesen
6	Unsigned Char	0xX1	Version des Request
<b>Trigger-Verzögerung lesen (GTD) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0B	Telegrammlänge
5	Unsigned Char	0x28	Trigger-Verzögerung lesen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	Unsigned Int	0xXX	Trigger-Verzögerung in msec (max. 3000 msec) in Encoder-Schritten (max. 65535 Schritte)
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Job Permanent speichern (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Job Permanent speichern (SJP) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x0D	Speichern aller zuvor temporär durchgeführten Telegramme
<b>Job Permanent speichern (SJP) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x0B	Telegrammlänge
5	Unsigned Char	0x0D	Job Permanent speichern
6 - 7	Unsigned Short	0xFF	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Low
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## 11.5.4 Kalibrierung

### Kalibrierung: Initialisieren (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Kalibrierung: Initialisieren (CCD) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x1F	Initialisieren (Calibration: Clear Data)
<b>Kalibrierung: Initialisieren (CCD) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x1F	Initialisieren (Calibration: Clear Data)
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
<b>Weitere Informationen:</b>			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Kalibrierung: Weltpunkt hinzufügen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Kalibrierung: Weltpunkt hinzufügen (CAW) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x15	Telegrammlänge
5	Unsigned Char	0x26	Kalibrierung: Weltpunkt hinzufügen
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x01: Nur Referenzmarken Kalibrierplatte (Roboter) 0x04: Weltpunkt und Bildpunkt Punktpaarliste (Roboter)
8 - 9	Unsigned Short	0x00	Konstant (2 Byte)
10 - 13	Unsigned Int	0xXX	Welt- X (in mm *1000)
14 - 17	Unsigned Int	0xXX	Welt- Y (in mm *1000)
18 - 21	Unsigned Char	0x00	Konstant (4 Byte)
<b>Kalibrierung: Weltpunkt hinzufügen (CAW) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x11	Telegrammlänge
5	Unsigned Char	0x26	Kalibrierung: Weltpunkt hinzufügen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 9	Unsigned Short	0xXX	Aktuelle Anzahl Punkte
10 -13	Unsigned Int	0xXX	Bildpunkt X
14 - 17	Unsigned Int	0xXX	Bildpunkt Y
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja



Status des Signals Ready während Bearbeitung:	Keine Veränderung
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

Hinweis: Für den Request CAW muss das Gesamt-Jobergebnis positiv sein.

## Kalibrierung: Punktpaarliste (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Kalibrierung: Punktpaarliste (CCL) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x06	Telegrammlänge
5	Unsigned Char	0x1E	Kalibrierung: Punktpaarliste
6	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
<b>Kalibrierung: Punktpaarliste (CCL) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x19	Telegrammlänge
5	Unsigned Char	0x1E	Kalibrierung: Punktpaarliste
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 9	Unsigned Short	0xXX	Aktuell höchster Punktpaarindex
10 - 13	Unsigned Int	0xXX	Abweichung Kalibrierung, RMSE
14 - 17	Unsigned Int	0xXX	Abweichung Kalibrierung, Mittelwert
18 - 21	Unsigned Int	0xXX	Abweichung Kalibrierung, Max.
22 - 25	Unsigned Int	0xXX	Abweichung Kalibrierung, Min.
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Kalibrierung: Punktpaarliste validieren (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

Kalibrierung: Punktpaarliste validieren (CVL) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x05	Telegrammlänge
5	Unsigned Char	0x20	Kalibrierung: Punktpaarliste validieren
Kalibrierung: Punktpaarliste validieren (CVL) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x19	Telegrammlänge
5	Unsigned Char	0x20	Kalibrierung: Punktpaarliste validieren
6	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 9	Unsigned Short	0xXX	Aktuell höchster Punktpaarindex
10 - 13	Unsigned Int	0xXX	Abweichung Kalibrierung, RMSE
14 - 17	Unsigned Int	0xXX	Abweichung Kalibrierung, Mittelwert
18 - 21	Unsigned Int	0xXX	Abweichung Kalibrierung, Max.
22 - 25	Unsigned Int	0xXX	Abweichung Kalibrierung, Min.
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Kalibrierung: Kalibrierplatte (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Kalibrierung: Kalibrierplatte (CCP) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x09	Telegrammlänge
5	Unsigned Char	0x24	Kalibrierung: Kalibrierplatte
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
8	Unsigned Char	0xXX	0x00: Es werden keine Referenzmarken verwendet. Ursprung Mess-Koordinatensystem identisch mit Ursprung Kalibrierplatten-Koordinatensystem. 0x01: Es werden keine Referenzmarken verwendet. Mess-Koordinatensystem identisch mit Kamera-Koordinatensystem. 0x02: Verwendet Weltsystem, Referenzmarke Job 0x03: Verwendet Weltsystem, Referenzmarke Befehl CAW
9	Unsigned Char	0xXX	0x00: Kalibrierung interne und externe Sensor Parameter 0x01: Validierung der Kalibrierung 0x02: Kalibrierung interne Sensor Parameter 0x05: Kalibrierung Transformation Mess-Koordinatensystem
<b>Kalibrierung: Kalibrierplatte (CCP) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x3D	Telegrammlänge
5	Unsigned Char	0x24	Kalibrierung: Kalibrierplatte
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 9	Unsigned Short	0xXX	Anzahl aktuell detektierter Punkte
10 - 13	Unsigned Int	0xXX	Abweichung Kalibrierung, RMSE
14 - 17	Unsigned Int	0xXX	Abweichung Kalibrierung, Mittelwert

18 - 21	Unsigned Int	0xXX	Abweichung Kalibrierung, Max.
22 - 25	Unsigned Int	0xXX	Abweichung Kalibrierung, Min.
26 - 29	Unsigned Int	0xXX	CPF_MF X (in Kundeneinheit * 1000)
30 - 33	Unsigned Int	0xXX	CPF_MF Y (in Kundeneinheit * 1000)
34 - 37	Unsigned Int	0x00	CPF_MF Z (in Kundeneinheit * 1000)
38 - 41	Unsigned Int	0x00	CPF_MF Winkel X (in Grad * 1000)
42 - 45	Unsigned Int	0x00	CPF_MF Winkel Y (in Grad * 1000)
46 - 49	Unsigned Int	0xXX	CPF_MF Winkel Z (in Grad * 1000)
50 - 53	Unsigned Int	0xXX	Abweichung Referenzmarken, Mittelwert
54 - 57	Unsigned Int	0xXX	Abweichung Referenzmarken, Max
58 - 61	Unsigned Int	0xXX	Abweichung Referenzmarken, Min
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Kalibrierung: Referenzmarke setzen (BINÄR)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

### Übersicht Telegramme

<b>Kalibrierung: Referenzmarke setzen (CSF) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x2B	Kalibrierung: Referenzmarke setzen
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
<b>Kalibrierung: Referenzmarke setzen (CSF) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x2B	Telegrammlänge
5	Unsigned Char	0x2B	Kalibrierung: Referenzmarke setzen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8 - 11	Unsigned Int	0xXX	X-Wert
12 - 15	Unsigned Int	0xXX	Y-Wert
16 - 19	Unsigned Int	0xXX	Z-Wert
20 - 23	Unsigned Int	0xXX	Winkel X-Wert
24 - 27	Unsigned Int	0xXX	Winkel Y-Wert
28 - 31	Unsigned Int	0xXX	Winkel Z-Wert
32 - 35	Unsigned Int	0xXX	Abweichung Referenzmarken, Mittelwert
36 - 39	Unsigned Int	0xXX	Abweichung Referenzmarken, Max.
40 - 43	Unsigned Int	0xXX	Abweichung Referenzmarken, Min
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung

Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>
------------------------------	---

## Kalibrierung: Bild hinzufügen (BINÄR)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

### Übersicht Telegramme

<b>Kalibrierung: Bild hinzufügen (CAI) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x22	Telegrammlänge 34 (0x22) Byte
5	Unsigned Char	0x34	Kalibrierung: Bild hinzufügen
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	Modus 0x01: Mehrbild-Kalibrierung 0x02: Hand-Eye-Kalibrierung (Roboter) 0x03: Base-Eye-Kalibrierung (Roboter)
8	Unsigned Short	0x00	Konstant
9	Unsigned Char	0xXX	Messebene definieren 0x00: Bild nicht verwenden, um Messebene zu definieren 0x01: Bild verwenden, um Messebene zu definieren
10	Unsigned Char	0xXX	"Roboter: Rotationsreihenfolge" 0x00: Im Job angegebene Rotationsreihenfolge verwenden 0x01: Yaw-Pitch-Roll (z.B. Stäubli) 0x02: Roll-Pitch-Yaw (z.B. Kuka, Fanuc, Hanwha, ABB**, UR**) **bei Nutzung von entsprechender Umwandlungsfunktion
11-14	Unsigned Char		Pose_TCP Pos. X (in Kundeneinheit * 1000)
15-18	Unsigned Char		Pose_TCP Pos. Y (in Kundeneinheit * 1000)
19-22	Unsigned Char		Pose_TCP Pos. Z (in Kundeneinheit * 1000)
23-26	Unsigned Char		Pose_TCP Winkel X (in Grad * 1000)
27-30	Unsigned Char		Pose_TCP Winkel Y (in Grad * 1000)



31-34	Unsigned Char		Pose_TCP Winkel Z (in Grad * 1000)
<b>Kalibrierung: Bild hinzufügen (CAI) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1-4	Unsigned Int	0x0A	Telegrammlänge
5	Unsigned Char	0x34	Kalibrierung: Bild hinzufügen
6-7	Unsigned Short	0xFF	<a href="#">Fehlercodes</a>
8	Unsigned Short	0xFF	Aktuelle Anzahl Bilder in Liste
9-10	Unsigned Char	0xFF	Anzahl insgesamt detektierter Punkte
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Ja
Akzeptiert wenn Ready Low:			Nein
Status des Signals Ready während Bearbeitung:			Low
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## Kalibrierung: Mehrbild (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Kalibrierung: Mehrbild (CMP) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x09	Telegrammlänge 9 (0x09) Byte
5	Unsigned Char	0x35	Kalibrierung: Mehrbild
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
8	Unsigned Char	0xXX	Ursprung des Welt-Koordinatensystems: 0x00: Welt-Koordinatensystem identisch mit dem Kalibrierplatten-Koordinatensystem (Zentrum der Platte). 0x01: Ursprung des Welt-Koordinatensystems so, dass dieser mit Ursprung des Bild-Koordinatensystems übereinstimmt (oberes linkes Pixel). 0x02: (nur bei Kalibrierplatte (Roboter)) Welt-Koordinatensystem von Referenzpunkten verwenden, wie in der Jobdatei angegeben. 0x03: (nur bei Kalibrierplatte (Roboter)) Welt-Koordinatensystem von Referenzpunkten verwenden, wie im Request CAW eingestellt.
9	Unsigned Char	0xXX	Modus 0x00: Kalibrieren (interne und externe Parameter) 0x01: Validieren (vorhandene Kalibrierung verwenden; mindestens ein Kalibrierpunkt wird hinzugefügt. Über Rückprojektion kann zurückgeschlossen werden, ob der Punkt zur aktuellen Kalibrierung passt, oder verschoben ist) 0x02: Kalibrieren (nur interne Parameter) 0x03: Kalibrieren (nur externe Parameter unter Verwendung der neuen internen Parameter) 0x04: Kalibrieren (nur externe Parameter) 0x05: Nur Messebene (CPF_MF) kalibrieren
<b>Kalibrierung: Mehrbild (CMP) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung

1-4	Unsigned Int	0x1D	Telegrammlänge 29 (0x1D) Byte
5	Unsigned Char	0x35	Kalibrierung: Mehrbild
6-7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	Abdeckung Sichtfeld in % 0x00: keinerlei Abdeckung 0x64: Abdeckung 100%
9-10	Unsigned Short	0xXX	Anzahl insgesamt detektierter Punkte
11	Unsigned Char	0xXX	Anzahl verwendeter Bilder
12	Unsigned Char	0xXX	Anzahl ungültiger Bilder
13	Unsigned Char	0xXX	Neigung zwischen Kalibrierplatten-Posen 0x00: ausreichend 0x01: nicht ausreichend
14-17	Unsigned Int	0xXX	Abweichung Kalibrierplatte RMSE [px]
18-21	Unsigned Int	0xXX	Abweichung Kalibrierplatte Max. [px]
22-25	Unsigned Int	0xXX	Abweichung Referenzmarken, RMSE (in Kundeneinheit * 1000)
26-29	Unsigned Int	0xXX	Abweichung Referenzmarken, Max. [px]
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Kalibrierung: Roboter Mehrbild (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

<b>Kalibrierung: Roboter Mehrbild (CRP) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x09	Telegrammlänge in Byte 9 Byte
5	Unsigned Char	0x36	Kalibrierung: Kalibrierplatte Roboter
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
8	Unsigned Char	0xXX	Ursprung des Welt-Koordinatensystems: 0x04: Setze Ursprung des Koordinatensystems gleich Roboter-Koordinatensystem
9	Unsigned Char	X	Modus 0x00: Kalibrieren (interne und externe Parameter) 0x01: Validieren (vorhandene Kalibrierung verwenden; mindestens ein Kalibrierpunkt wird hinzugefügt. Über Rückprojektion kann zurückgeschlossen werden, ob der Punkt zur aktuellen Kalibrierung passt, oder verschoben ist) 0x02: Kalibrieren (nur interne Parameter) 0x03: Kalibrieren (nur externe Parameter unter Verwendung der neuen internen Parameter) 0x04: Kalibrieren (nur externe Parameter) 0x05: Nur Messebene (CPF_MF) kalibrieren 0x06: Nur Hand-Eye (TCP_CF)/Base-Eye (RF_CF) kalibrieren
<b>Kalibrierung: Roboter Mehrbild (CRP) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1-4	Unsigned Int	0x2C	Telegrammlänge 44 (0x2C) Byte
5	Unsigned Char	0x36	Kalibrierung: Kalibrierplatte Roboter
6-7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>

8	Unsigned Char	0xXX	Abdeckung Sichtfeld 0x00: nicht ausreichend 0x01: ausreichend
9-10	Unsigned Short	0xXX	Anzahl insgesamt detektierter Punkte
11	Unsigned Char	0xXX	Anzahl verwendeter Bilder
12	Unsigned Char	0xXX	Anzahl ungültiger Bilder
13-16	Unsigned Int	0xXX	Abweichung Kalibrierplatte RMSE [px]
17-20	Unsigned Int	0xXX	Abweichung Kalibrierplatte Max. [px]
21-24	Unsigned Int	0xXX	Abweichungen Kalibrierplatten-Pose Translation RMSE (in Kundeneinheit * 1000)
25-28	Unsigned Int	0xXX	Abweichungen Kalibrierplatten-Pose Translation Max. (in Kundeneinheit * 1000)
29-32	Unsigned Int	0xXX	Abweichungen Kalibrierplatten-Pose Rotation RMSE (in Grad * 1000)
33-36	Unsigned Int	0xXX	Abweichungen Kalibrierplatten-Pose Rotation Max. (in Grad * 1000)
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	

## Kalibrierung: Kalibrierung kopieren (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Kalibrierung: Kalibrierung kopieren (CCC) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x09	Telegrammlänge
5	Unsigned Char	0x25	Kalibrierung: Kalibrierung kopieren
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0x01	Konstant
8	Unsigned Char	0xXX	Destination 0 : In alle Jobs kopieren >0 : In angegebenen Job kopieren
9	Unsigned Char	0xXX	0: Immer kopieren, wenn die Kalibrierung aktiv ist. 1: Nur kopieren, wenn die Kalibrieremethode identisch ist.
<b>Kalibrierung: Kalibrierung kopieren (CCC) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x08	Telegrammlänge
5	Unsigned Char	0x25	Kalibrierung: Kalibrierung kopieren
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	00: Erfolgreich >0: Jobnummer bei dem der Fehler auftritt.
Weitere Informationen:			
Akzeptiert im Runmode:		Ja	
Akzeptiert im Konfigurationsmodus:		Nein	
Akzeptiert wenn Ready Low:		Ja	
Status des Signals Ready während Bearbeitung:		Keine Veränderung	
Unterstützte Schnittstellen:		<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>	

## Kalibrierung: Parameter setzen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

[Übersicht Telegramme](#)

Kalibrierung: Parameter setzen (CSP) Request String an Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge in Byte 16 Byte (0x13) + Länge gewählter Parameter
5	Unsigned Char	0x29	Kalibrierung: Parameter setzen
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	0x00: Temporär 0x01: Permanent
8	Unsigned Char	0xXX	Parameter-Nummer, siehe Tabelle <a href="#">Kalibrierparameter für Telegramme CSP und CGP</a>
9 - 12	Unsigned Int	0xXX	Länge der nachfolgenden Daten
13 ... n	Unsigned Char	0xXX	Parameter-Wert, siehe Tabelle <a href="#">Kalibrierparameter für Telegramme CSP und CGP</a>
Kalibrierung: Parameter setzen (CSP) Response String vom Sensor (BINÄR)			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x29	Kalibrierung: Parameter setzen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

Kalibrierparameter: siehe Tabelle [Kalibrierparameter für Telegramme CSP und CGP](#)

## Kalibrierung: Parameter lesen (BINÄR)

[Telegramme: Verfügbarkeit und unterstützte Schnittstellen](#)

### Übersicht Telegramme

<b>Kalibrierung: Parameter lesen (CGP) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x07	Telegrammlänge
5	Unsigned Char	0x2A	Kalibrierung: Parameter lesen
6	Unsigned Char	0x01	Version des Request
7	Unsigned Char	0xXX	<a href="#">Parameter Nummer</a>
<b>Kalibrierung: Parameter lesen (CGP) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge in Byte, 12 Byte (0x0C) + Länge gewählter Parameter
5	Unsigned Char	0x2A	Kalibrierung: Parameter lesen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	<a href="#">Parameter Nummer</a>
9 - 12	Unsigned Int	0xXX	Länge der nachfolgenden Daten
13 ... n	Unsigned Char	0xXX	<a href="#">Parameter Wert</a>
Weitere Informationen:			
Akzeptiert im Runmode:			Ja
Akzeptiert im Konfigurationsmodus:			Nein
Akzeptiert wenn Ready Low:			Ja
Status des Signals Ready während Bearbeitung:			Keine Veränderung
Unterstützte Schnittstellen:			<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>



**Kalibrierparameter für Telegramme CSP und CGP**

Parameter-Bezeichnung	Parameter Nummer	Parameter Wert	Länge	Status Kalibrierung nach CSP
Status Kalibrierung	0x01	0x00: Ungültig 0x01: Gültig	1 Byte	—*
Auswahl Kalibrierermethode	0x02	0x00: Keine 0x02: Punktpaarliste (Roboter) 0x03: Kalibrierplatte (Messen) 0x04: Kalibrierplatte (Roboter) 0x05: Hand-Eye-Kalibrierung (Roboter) 0x06: Base-Eye-Kalibrierung (Roboter)	1 Byte	ungültig
Kundeneinheit	0x04	0x00: Millimeter [mm] 0x01: Zentimeter [cm] 0x02: Meter [m] 0x03: Inch ["] 0x04: Beliebige Einheit [au]	1 Byte	keine Änderung
Interne Parameter	0x0A	Brennweite (in mm *1000) Kappa (*1000) Pixelabstand X (in $\mu\text{m}$ * 1000) Pixelabstand Y (in $\mu\text{m}$ * 1000) Koordinatenursprung X (in Pixel * 1000) Koordinatenursprung Y (in Pixel * 1000) Bildgröße X (Anzahl Pixel) Bildgröße Y (Anzahl Pixel)	0x20 (8 * 4 Byte je Wert)	—*
Bezug Kamera- zu Mess-Koordinatensystem (CF_MF)	0x0B	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	0x18 (6 * 4 Byte je Wert)	—*

Parameter-Bezeichnung	Parameter Nummer	Parameter Wert	Länge	Status Kalibrierung nach CSP
Bezug Kamera- zu Kalibrierplatten-Koordinatensystem (CF_CPF)	0x0C	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	0x18 (6 * 4 Byte je Wert)	—*
Bezug Roboter- zu Kamera-Koordinatensystem (RF_CF)	0x0D	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	0x18 (6 * 4 Byte je Wert)	—*
Bezug Kalibrierplatten- zu Mess-Koordinatensystem (CPF_MF)	0x0E	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	0x18 (6 * 4 Byte je Wert)	—*
Bezug Roboter- zu Mess-Koordinatensystem (RF_MF)	0x0F	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	0x18 (6 * 4 Byte je Wert)	—*
Bezug Werkzeug- zu Kamera-Koordinatensystem (TCP_CF)	0x10	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	0x18 (6 * 4 Byte je Wert)	—*
Bezug Roboter- zu Werkzeug-Koordinatensystem (RF_TCP)	0x11	Translation X, Y, Z (in Kundeneinheit * 1000) Winkel X, Y, Z (in Grad * 1000)	0x18 (6 * 4 Byte je Wert)	keine Änderung
Z-Verschiebung Mesesebene	0x15	(in Kundeneinheit * 1000)	4 Byte	keine Änderung
Brennweite in [mm]	0x16	[mm * 1000]	4 Byte	ungültig (CSP nur für C-Mount)
Kalibrierplatten-Typ	0x17	Zeichenkette mit Namen der Beschreibungsdatei	n	ungültig
Referenzmarke 1	0x18	Translation X, Y, Z (in Kundeneinheit * 1000)	0x0C (3* 4 Byte je Wert)	ungültig
Referenzmarke 2	0x19			
Referenzmarke 3	0x1A			
Referenzmarke 4	0x1B			

Parameter-Bezeichnung	Parameter Nummer	Parameter Wert	Länge	Status Kalibrierung nach CSP
Anzahl vorhandener Kalibrierplatten-Typen	0x25	Request – Auswahl Typ: 0x00: Alle 0x01: Messen 0x02: Roboter Response: Anzahl Platten	Request: 1 Response: 2	–*
Vorhandene Kalibrierplatten-Typen (Dateinamen)	0x26	Request – Auswahl Typ: 0x00: Alle 0x01: Messen 0x02: Roboter Request – Index: 0: Alle Dateinamen >0: Auswahl Index Antwort: Dateinamen Kalibrierplatten	Request: 1 Response: 5 (String)	–*
Roboter: Rotationsreihenfolge	0x27	"Roboter: Rotationsreihenfolge" 0x00: Im Job angegebene Rotationsreihenfolge verwenden 0x01: Yaw-Pitch-Roll (z.B. Stäubli) 0x02: Roll-Pitch-Yaw (z.B. Kuka, Fanuc, Hanwha, ABB**, UR**) **bei Nutzung von entsprechender Umwandlungsfunktion	1 Byte	ungültig
Mittlere Sensorauflösung	0x29	Wert (in Kundeneinheit/Pixel * 1000)	4 Byte	–*

\* CSP nicht möglich (Parameter read-only, kann nicht gesetzt werden).

## 11.5.5 Visualisierung

### Bild holen (BINÄR)

Telegramme: Verfügbarkeit und unterstützte Schnittstellen

Übersicht Telegramme

<b>Bild holen (GIM) Request String an Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0x06	Telegrammlänge
5	Unsigned Char	0x03	Bild holen
6	Unsigned Char	0xXX	0x00: Letztes Bild 0x01: Letztes Fehlerbild 0x02: Letztes Gutbild
<b>Bild holen (GIM) Response String vom Sensor (BINÄR)</b>			
Byte Nr.	Datentyp	Inhalt	Bedeutung
1 - 4	Unsigned Int	0xXX	Telegrammlänge in Byte, 13 Byte (0x0D) + Anzahl Byte abhängig vom Bildformat z.B. 00 04 B0 0D (Dez. 307213)
5	Unsigned Char	0x03	Bild holen
6 - 7	Unsigned Short	0xXX	<a href="#">Fehlercodes</a>
8	Unsigned Char	0xXX	Typ Bild 0: Graustufen 3: Bayer Pattern_BG Bei Konvertierung des Farbbildes von Bayer in RGB, muss der entsprechende Bildtyp berücksichtigt werden.
9	Unsigned Char	0xXX	Ergebnis Bild 00: Schlecht-Bild 01: Gut-Bild
10 - 11	Unsigned Short	0xXX	Anzahl der Zeilen z.B. 01 E0 = 480
12 - 13	Unsigned Short	0xXX	Anzahl der Spalten z.B. 02 80 = 640
14 ... n	Unsigned Char	0xXX	Binäre Bilddaten (Zeilen * Spalten)

Weitere Informationen:	
Akzeptiert im Runmode:	Ja
Akzeptiert im Konfigurationsmodus:	Nein
Akzeptiert wenn Ready Low:	Ja
Status des Signals Ready während Bearbeitung:	Low
Unterstützte Schnittstellen:	<a href="#">Telegramme: Verfügbarkeit und unterstützte Schnittstellen</a>

## 11.5.6 Datenausgabe BINÄR

Ausgabedaten (BINÄR), dynamisch zusammengesetzt nach Benutzereinstellungen in der Software unter:

Vision Sensor Configuration Studio / Ausgabe / Datenausgabe.

Prinzipieller String-Aufbau:

<START> (((<OPTIONAL FIELDS> <PAYLOAD>))) <CHKSUM> <TRAILER>



**HINWEIS:**

Die Länge und Datentypen der Nutzdaten sind Standardwerte. Über "Datenausgabe" / "Detektorspezifische Nutzdaten" können Faktor und Bittiefe eingestellt werden.

### Ausgabedaten (BINÄR):

<OPTIONAL FIELDS>				
Parameter	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Gewählte Felder	Über diese Checkbox werden alle gewählten Felder angezeigt. Die Checkbox "Gewählte Felder" selbst wird nicht angezeigt.	2	Die Reihenfolge der Ausgabe ist von links nach rechts und von oben nach unten, d.h. pro aktiver Checkbox wird ein Bit beginnend beim niederwertigsten gesetzt.	Alle Typen
Telegrammlänge	Anzahl der Zeichen inkl. der Zeichen für die Telegrammlänge selbst.	2	Unsigned Short	Alle Typen
Statusbyte	Gibt den Triggermodus wieder.	2	0x06 0x00 = Trigger; 0x05 0x00 = Freilauf	Alle Typen

<b>«OPTIONAL FIELDS»</b>				
<b>Parameter</b>	<b>Beschreibung</b>	<b>Länge BINÄR [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Detektor-Ergebnisse	Ausgabe des Gesamtergebnisses pro Detektor. Byte 1 Bit 1 (LSB) = Globales Jobergebnis (1 = Pass, 0 = Fail) Bit 2 = Boolesches Erg. nur Lagenachführung, Lagenachführung inaktiv = True	3 ... 35		Alle Typen
Digitalausgänge	Gibt das Verknüpfungsergebnis pro digitalem Ausgang zurück.	n	Byte 1 und 2: die Anzahl aktiver Ausgänge Byte 3 ... n: Ausgänge, bitcodiert	Alle Typen
Log. Ausgänge	Gibt das Verknüpfungsergebnis pro logischem Ausgang zurück.	n	Byte 1 und Byte 2: die Anzahl aktiver! log. Ausgänge Byte 3 ... n alle aktiven logischen Ausgänge,	Alle Typen
Ausführungszeit	Gibt die Ausführungszeit der letzten Auswertung zurück.	4	Signed Integer	Alle Typen
Aktiver Job	Gibt den Job der letzten Auswertung zurück.	1	Unsigned Int U8	Alle Typen

⟨PAYLOAD⟩

Übersicht detektorspezifische Nutzdaten – Werte

Allgemein

⟨PAYLOAD⟩ Allgemein				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Zähler "Alle Auswertungen"	Gesamtanzahl der Prüfungen	4	Signed Integer	GENERAL
Zähler Gutteile	Anzahl der Prüfungen mit Resultat "OK"	4	Signed Integer	GENERAL
Zähler Schlechteile	Anzahl der Prüfungen mit Resultat "Fehler"	4	Signed Integer	GENERAL
Zeitüberschreitung	Zeigt eine Überschreitung der maximalen Zykluszeit an.	1	BOOL	GENERAL
Aufnahme	Gibt die Anzahl der Bildaufnahmewiederholungen der letzten Auswertung an. Nur in Kombination mit Mehrfachbildaufnahme.	4	INT	GENERAL
String	Mit diesem Feld kann ein Konstanter String in die Datenausgabe eingegeben werden.	0 ... 5	STRING	GENERAL



◀PAYLOAD▶ Allgemein				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Job-Prüfsumme	Berechnet eine Prüfsumme über den aktiven Job. Darin werden alle jobspezifischen Einstellungen berücksichtigt außer dem "Geändert"-Datum. Eine Änderung von Jobsatz-globalen Einstellungen führt zu einer Änderung der Prüfsumme in allen Jobs. Wird die Prüfsumme für einen Job ermittelt, können für diesen keine temporären Änderungen im Run-Modus durchgeführt werden.	8	STRING	GENERAL







### Basiswerte

◀PAYLOAD▶ Basiswerte				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Score	[%]	4	Signed Integer	Alle Detektoren
Gesamtergebnis	Boolesches Detektorergebnis **	1	BOOL	Alle Detektoren
Ausführungszeit	Ausführungszeit des einzelnen Detektors in [msec].	4	Signed Integer	Alle Detektoren







## Position / Lage

◀PAYLOAD▶ Position / Lage				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Pos. X	X-Koordinate der gefundenen Position, 1/1000 [Kundeneinheit] **	4	Signed Integer	
Pos. Y	Y-Koordinate der gefundenen Position, 1/1000 [Kundeneinheit] **	4	Signed Integer	
Pos. Z	Z-Koordinate der gefundenen Position, 1/1000 [Kundeneinheit]		Signed Integer	 Mit Ergebnisoffset: 
Delta Pos. X	Delta Position X zwischen eingelerntem und gefundenem Objekt, 1/1000 [Kundeneinheit]	4	Signed Integer	
Delta Pos. Y	Delta Position Y zwischen eingelerntem und gefundenem Objekt, 1/1000 [Kundeneinheit]	4	Signed Integer	
Delta Pos. Z	Delta Position Z zwischen eingelerntem und gefundenem Objekt, 1/1000 [Kundeneinheit]	4	Signed Integer	 Mit Ergebnisoffset: 



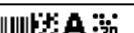
«PAYLOAD» Position / Lage				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Winkel X	Orientierung des gefundenen Objekts, bezogen auf X-Achse, 1/1000 [°]	4	Signed Integer	Mit Ergebnisoffset: 
Winkel Y	Orientierung des gefundenen Objekts, bezogen auf Y-Achse, 1/1000 [°]	4	Signed Integer	Mit Ergebnisoffset: 
Winkel Z	Orientierung des gefundenen Objekts, bezogen auf Z-Achse, 1/1000 [°]	4	Signed Integer	 
Winkel (45)	Orientierung der Bounding-Box des gefundenen Codes [°], Wertebereich: -45° ... 45°	4	Signed Integer	
Winkel (180)	Orientierung der Breite (lange Achse) des Objekts [°], Wertebereich: -90° ... 90° 0° = Ost, Gegen- uhrzeigersinn	4	Signed Integer	
Winkel (360)	Orientierung der Breite (lange Achse) des Objektes [°], Wertebereich: -180° ... 180° 0° = Ost, Gegen- uhrzeigersinn	4	Signed Integer	

«PAYLOAD» Position / Lage				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Delta Winkel X	Winkel zwischen eingelerntem und gefundenem Objekt, 1/1000 [°]	4	Signed Integer	 Mit Ergebnisoffset:
Delta Winkel Y	Winkel zwischen eingelerntem und gefundenem Objekt, 1/1000 [°]	4	Signed Integer	 Mit Ergebnisoffset:
Delta Winkel Z	Winkel zwischen eingelerntem und gefundenem Objekt, 1/1000 [°]	4	Signed Integer	
Pose 3D (X, Y, Z, Winkel X, Winkel Y, Winkel Z)	Koordinaten des gefundenen Objekts, 1/1000 [Kundeneinheit] Winkel: 1/1000 Grad	je 4 Byte pro Wert	Signed Integer	 Mit Ergebnisoffset:
Delta Pose 3D (X, Y, Z, Winkel X, Winkel Y, Winkel Z)	Delta Koordinaten des gefundenen Objekts, 1/1000 [Kundeneinheit] Winkel: 1/1000 Grad	je 4 Byte pro Wert	Signed Integer	 Mit Ergebnisoffset:
Positionskontrolle		1	BOOL	

## Messen

◀PAYLOAD▶ Messen				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Höhe	Höhe des geometrischen Elements [Kundeneinheit], Höhe $\geq 0$ , Höhe $\leq$ Breite	4	Signed Integer	
Breite	Breite des geometrischen Elements [Kundeneinheit] Breite $\geq 0$ , Breite $\geq$ Höhe	4	Signed Integer	
Radius	Radius des gefitteten Kreises [Kundeneinheit]	4	Signed Integer	
Fläche	Fläche des BLOBs ohne Löcher, 1/1000 [Pixel]	4	Signed Integer	
Fläche (inkl. Löcher)	Fläche des BLOBs mit Löchern, 1/1000 [Pixel]	4	Signed Integer	
Abstand	Berechneter Abstand [Kundeneinheit] **	4	Signed Integer	


## Identifikation

◀PAYLOAD▶ Identifikation				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
String Zielmarken-ID Zielmarkenname	Inhalt des gelesenen Codes oder Inhalt der Zielmarke oder zugewiesener Zielmarkenname. Wird eine feste Stringlänge gewünscht, so müssen die minimale Stringlänge (Detektorspezifische Nutzdaten) und die maximale Stringlänge (Detektoreinstellungen) auf den gleichen Wert (z.B. 127) gesetzt werden.	n	STRING	
Stringlänge Länge Zielmarkenname (Zeichen)	Länge des gelesenen Codes [Bytes]	4	Signed Integer	
Stringlänge (Bytes) Länge Zielmarkenname (Bytes) Länge Zielmarken-ID (Bytes)	Länge des gelesenen Codes [Bytes]	4	Signed Integer	


<b>⟨PAYLOAD⟩ Identifikation</b>				
<b>Wert</b>	<b>Beschreibung</b>	<b>Länge BINÄR [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Stringvergleich	Überprüfung des Inhaltes der gelesenen Informationen. Die Überprüfung des Inhaltes der gelesenen Informationen erfolgt auf Basis von regulären Ausdrücken (siehe Detektor Datacode, Reiter Referenzstring)	1	BOOL	<b>A</b>
Abgeschnitten	Code komplett oder abgeschnitten 0: Code komplett 1: Code abgeschnitten	1	BOOL	<b>A</b>

### Identifikation - Qualität

<b>⟨PAYLOAD⟩ Identifikation - Qualität</b>				
<b>Wert</b>	<b>Beschreibung</b>	<b>Länge BINÄR [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Qualität - Gesamt	Ausgabe aller Q-Parameter. Abhängig von gewähltem Codetyp und Norm.	je 1 Byte pro Wert; getrennt durch angegebenes Trennzeichen Bei 2D Code Parameter Q9 (Mean light): 1...3	Unsigned Char; bei 2D Code Q9 (Meanlight) Unsigned Short	

◀PAYLOAD▶ Identifikation - Qualität				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Qualität - Einzel	Ausgabe der Einzel- Qualitätswerte: Auswahl Q1-Q24 abhängig von gewähltem Codetyp und Norm. Ziffern: 1-4 Buchstaben: A-F	1	Unsigned Char; bei 2D Code Q9 (Meanlight) Unsigned Short	
Min. Qualität	Prüfung, ob die minimal geforderte Qualität erreicht wurde	4	Unsigned Int	<b>A</b>

## Farbe





◀PAYLOAD▶ Farbe				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Farbwert: <ul style="list-style-type: none"> <li>• Rot, Grün, Blau</li> <li>• Farbton, Sättigung, Helligkeit</li> <li>• Luminanz, A, B</li> </ul>	Wert für Farbparameter	4	Signed Integer	
Farbabstand	Abstand der aktuellen Farbe gegenüber der eingelesenen Farbe	4	Signed Integer	

















## Zählen / Anzahl

◀PAYLOAD▶ Zählen / Anzahl				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Anzahl Objekte	Anzahl der gefundenen Objekte [Stück]	4	Signed Integer	
Anzahl gültiger Objekte	Anzahl der gefundenen, gültigen Objekte [Stück]	4	Signed Integer	
Anzahl Suchstreifen	Anzahl der parallelen Suchstreifen in die, die Breite des Suchbereichs aufgeteilt wird. [Stück]	4	Signed Integer	(nur Kantenantastung) <b>F</b>
Anzahl gültige Suchstreifen	Überprüfung ob die Anzahl der gefundenen Suchstreifen innerhalb eines vorgegebenen Bereichs liegt. [Gut/schlecht oder Stück]	4	Signed Integer	(nur Kantenantastung) <b>F</b>
Ergebnisvektor	Vektor, der das Ergebnis (1/0) der gefundenen Instanzen enthält	n	BOOL	
Zu viele BLOBs		1	BOOL	

## Erweitert

◀PAYLOAD▶ Erweitert				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Skalierung	Ausgabe des Skalierungsbereichs 1/1000. Innerhalb des Skalierungsbereichs werden vergrößerte oder verkleinerte Objekte erkannt. Wertebereich 0,5 ... 2	4	Signed Integer	 (nur Konturvergleich) 
Exzentrizität	Numerische Exzentrizität Wertebereich 0,0 ... 1,0	4	Signed Integer	
Sicherheit	Ausgabe der Sicherheitswerte der einzelnen Zeichen. Der Sicherheitswert gibt an, wie zuverlässig der Leser ein Zeichen bewerten konnte. Wertebereich von 0 ... 100 [%]	4	Signed Integer	<b>A</b>
Referenzstring getroffen	Übereinstimmung des ausgegebenen Strings mit dem Referenzstring.	1	BOOL	<b>A</b>
Kontrast	Kontrast des Codes Wertebereich von 0 - 100 [%]	4	Signed Integer	
Korrektur	Anzahl der durch die Fehlerkorrekturen korrigierte Module [Stück]	4	Signed Integer	

<b>«PAYLOAD» Erweitert</b>				
<b>Wert</b>	<b>Beschreibung</b>	<b>Länge BINÄR [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Konturlänge	Anzahl der Pixel der äußeren Kontur, 1/1000 [Pixel]	4	Signed Integer	
Kompaktheit	Kompaktheit des BLOBs, (Kreis = 1, andere >1) Je stärker die Form des BLOB vom Kreis abweicht, desto größer wird der Wert der Kompaktheit.	4	Signed Integer	
Schwerpunkt X	X-Koordinate des Schwerpunkts, 1/1000	4	Signed Integer	
Schwerpunkt Y	Y-Koordinate des Schwerpunkts, 1/1000	4	Signed Integer	
Grauwert, mittlerer	Mittlerer Grauwert aller Pixel, die zum BLOB gehören.	4	Signed Integer	
Schaltswelle min.	Untere Schaltswelle für die Binarisierung der Objekte. 0...255	4	Signed Integer	
Schaltswelle max.	Obere Schaltswelle für die Binarisierung der Objekte. 0...255	4	Signed Integer	
Schaltswelle invertiert	Gibt an, ob der Bereich Min <-> Max invertiert ist. P: invertiert F: nicht invertiert	1	Unsigned Char	

«PAYLOAD» Erweitert				
Wert	Beschreibung	Länge BINÄR [Byte]	Datentyp	Verfügbar für
Abweichung, innen	Liefert die größte Abweichung zwischen der BLOB-Kontur und der Kontur des geometrischen Elements (Abweichung in den gefitteten Kreis hinein). [Kundeneinheit]	4	Signed Integer	
Abweichung, außen	Liefert den größten Abstand zwischen der BLOB-Kontur und der Kontur der geometrischen Form (Abweichung aus dem gefitteten Kreis heraus). [Kundeneinheit]	4	Signed Integer	
Abweichung, mittlere	Liefert den Mittelwert des Betrags aller Innen- und Außenabweichungen zwischen der BLOB-Kontur und der Kontur der geometrischen Form.	4	Signed Integer	
Achsenverhältnis	Verhältnis der langen zur kurzen Achse (a/b)	4	Signed Integer	
Bauch / Rücken, Fläche	Bauch- / Rücken-Lage, Basis: Fläche, Unterscheidung der Lage durch Vorzeichen, 1/1000	4	Signed Integer	
Ergebnis Index	Listenindex	4	Signed Integer	

<b>⟨PAYLOAD⟩ Erweitert</b>				
<b>Wert</b>	<b>Beschreibung</b>	<b>Länge BINÄR [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Suchstreifen Distanz	Berechneter Abstand [Kundeneinheit] / 1000 pro Such- streifen-Paar	4	Signed Integer	<b>F</b>

<b>⟨CHKSUM⟩</b>				
<b>Parameter</b>	<b>Beschreibung</b>	<b>Länge BINÄR [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Prüfsumme	XOR-Prüfsumme über alle Bytes im Tele- gramm. Wird als letz- tes Byte übertragen.	1	Unsigned Int	Alle Typen

<b>⟨TRAILER⟩</b>				
<b>Parameter</b>	<b>Beschreibung</b>	<b>Länge BINÄR [Byte]</b>	<b>Datentyp</b>	<b>Verfügbar für</b>
Nachspann	Zeichen, die am Ende der Zeichenfolge ein- gefügt werden	0 ... 8	Unsigned Int	Alle Typen



**HINWEIS:**

Wenn keine Kalibrierung ausgeführt wurde, beziehen sich alle Werte auf Pixel.

\*\* Detektor Messschieber: Abhängig von der gewählten Abstandsart. "Kleinster / Größter gegen-überliegender Abstand" = Vektor mit zwei Elementen [min; max].

Alle detektorspezifischen Daten mit Nachkommastellen werden als ganze Zahlen (mit 1000 multipliziert) übertragen und müssen nach Datenempfang deshalb durch 1000 geteilt werden. Die Werte werden im Format „Big Endian“ übertragen.

Beispiel: „Score“ Werte (BINÄR Protokoll)

In Vision Sensor Configuration Studio/Vision Sensor Visualisation Studio wird „Score“ = 35 angezeigt.

Via Ethernet werden dann z.B. folgende vier Bytes empfangen: 000,000,139,115

Formel zu Umrechnung:  $(\text{Byte4} * 256 + \text{Byte3}) * 65536 + \text{Byte2} * 256 + \text{Byte1} = \text{Value}$

Da Big Endian (vom Sensor) geschickt wird, gilt:

000 = HiWordByte, 000 = HiLowByte, 139 = HiByte, 115 = LoByte

$(0 * 256 + 0) * 65536 + (139 * 256) + 115 = 35699 / 1000 = 35,699$  (= echter Score Wert).

Winkelangaben bzw. andere negative Zahlen werden im Zweierkomplement dargestellt.



