

DCR 200i
IPS 200i



1	Allgemeines zum Dokument	3
2	Allgemeines zur XML-Schnittstelle	4
3	Requests	5
3.1	Zurücksetzen und Konfiguration	5
3.1.1	Reset	5
3.1.2	Factory Reset	5
3.1.3	Auto Setup	5
3.1.4	Justage-Modus	7
3.1.5	Teach	8
3.1.6	Konfiguration abfragen (getConfig)	9
3.1.7	Konfiguration wiederherstellen (setConfig)	9
3.2	Get Parameter	10
3.2.1	Beispiel: Abfragen des Etherhost UDP-Ports (Geräteparameter)	10
3.2.2	Beispiel: Abfragen der Belichtungszeit = Exposure Time (Job-Parameter)	10
3.2.3	Beispiel: Abfragen des Referenzcodes (Job-Parameter)	11
3.3	Put Parameter	11
3.3.1	Beispiel: Setzen des Etherhost UDP-Ports (Geräteparameter), temporär	11
3.3.2	Beispiel: Setzen der Belichtungszeit = Exposure Time (Job-Parameter), temporär	11
3.3.3	Beispiel: Setzen der Belichtungszeit = Exposure Time (Job-Parameter), permanent.....	12
3.3.4	Beispiel: Setzen des Referenzcodes (Job-Parameter), temporär	12
3.4	Weitere Beispiele	12
3.4.1	Einige Beispiele zum Setzen und Abfragen von „Etherhost-TCP-Enabled“	12
3.5	Geräteabfragen	13
3.5.1	Get Status	13
3.5.2	Version Request	13
3.5.3	setDefaultLogin	14
3.6	Steuerung und Bildabfrage	15
3.6.1	Scan Activate	15
3.6.2	Scan Deactivate	15
3.6.3	Get Image	16
3.6.4	Programmabfrage (activeJob)	16
3.6.5	Programmumschaltung (activeJob)	17
3.6.6	Subscribe / Unsubscribe	17
3.7	Fehlerfälle	18
3.7.1	Abgelehnter Request	18
3.7.2	Fehlgeschlagenes Kommando	19
3.7.3	Timeout nach unvollständigem Kommando	19

1 Allgemeines zum Dokument

Dieses Dokument beschreibt die XML-Kommunikation zwischen einem Leuze Sensor und einer Steuerung bzw. PC. Hierbei können über diverse Befehle unterschiedliche Aktionen auf dem Sensor ausgeführt werden.

Tabelle 1.1: Informationen zum Dokument

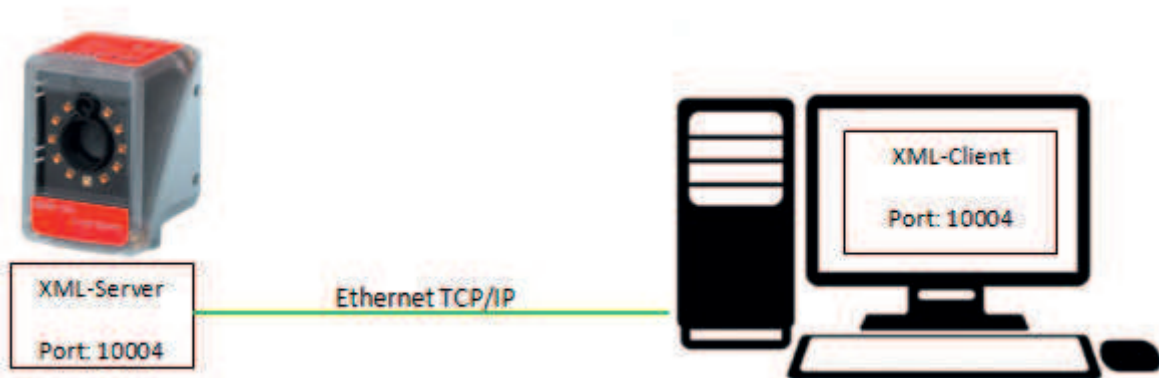
Projekt	XML-Schnittstellenbeschreibung
Version	1.2

Tabelle 1.2: Änderungsverzeichnis

Autor	Datum	Version	Kommentar
S. Abraham	21.04.2016	1.0	erstellt
S. Abraham	14.11.2017	1.1	Erweiterung setDefaultLogin
S. Abraham	17.08.2018	1.2	Erweiterung IPS 200i

2 Allgemeines zur XML-Schnittstelle

Der Sensor stellt einen XML-Server zur Verfügung. Dieser XML-Server kommuniziert auf einem konfigurierbaren Port über Ethernet TCP/IP mit dem XML-Client. Der Defaultwert für diesen Port ist 10004. Der Sensor muss sich im Prozessmodus befinden um über diese Schnittstelle zu kommunizieren.



HINWEIS



In zukünftigen Versionen, in denen sich Requests und Antworten befinden, können zusätzliche Informationen hinzukommen. Das gilt sowohl für Attribute als auch für Elemente.

HINWEIS



Der Port 10004 ist im Konfigurationstool webConfig nicht veränderbar.

3 Requests

Requests sind Nachrichten, die vom XML-Client (z. B. Steuerung) an den XML Server (Leuze Sensor) gesendet werden. Das kann das Abfragen oder Setzen von Parametern sein oder das Senden von Kommandos. Nahezu alle Requests (außer Reset) bewirken eine Antwort des Servers.

Das Antwortverhalten kann durch Angabe von a="1" (Sensor schickt Antwort) bzw. a="0" (Sensor antwortet nicht) gesteuert werden.

3.1 Zurücksetzen und Konfiguration

3.1.1 Reset

Löst einen Reset (kompletter Neustart), einschließlich Betriebssystem des Sensors aus. Der Request besitzt keine Antwort egal ob a="0" oder a="1" eingestellt wurden.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="reset" a="1">1</CMD></PUT></LEUZE_XML>
```

```
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="reset" a="1">
      1
    </CMD>
  </PUT>
</LEUZE_XML>
```

3.1.2 Factory Reset

Löst ein Zurücksetzen auf Werkseinstellungen aus. Die Einstellungen für die Ethernet-Kommunikation (IP, Netmask, Gateway) bleiben erhalten.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="factoryReset" a="1"></CMD></PUT></LEUZE_XML>
```

```
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="factoryReset " a="1">
    </CMD>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="factoryReset" result="OK">
    </CMD>
  </STATE>
</LEUZE_XML>
```

3.1.3 Auto Setup

DCR 200i

Die Auto-Setup-Funktion ermittelt die optimale Beleuchtungseinstellung, decodiert die vorliegenden Codes und speichert die gefundenen Codes (Codearten und Stellenanzahl) permanent in der Decodiertabelle. Die zuvor gespeicherten Codearten und Stellenanzahl in der Decodiertabelle werden gelöscht.

HINWEIS



Die Funktion Auto-Setup kann ebenfalls auf andere Weise aktiviert werden (z. B. Schalteingang oder Gerätetaste). In diesem Fall generiert der DCR 200i spontane STATE-Nachrichten an den XML-Client.

IPS 200i

Funktion nicht verfügbar.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="autoSetup" a="1"></CMD></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="autoSetup" a="1">
      </CMD>
    </PUT>
  </LEUZE_XML>
```

Dieses Kommando ohne einen Parameter bewirkt das gleiche wie der Parameter „start“:

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="autoSetup" a="1">start</CMD></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="autoSetup" a="1">
      start
    </CMD>
  </PUT>
</LEUZE_XML>
```

Neben „start“ kann auch „stop“ verwendet werden.

Antwort (eine oder mehrere Antworten):

DCR 200i:


```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="autoSetup" result="OK">
      </CMD>
    </STATE>
  </LEUZE_XML>

<LEUZE_XML>
  <STATE v="autoSetup.result" s="cur">
    <DAT>
      <BARCODE d="base64Binary">
        MmRfZGFkYSBdXT4gbWl0IDwhW0NEQVRBWyBNdXJrcyBdXT4=
      </BARCODE>
      <NUMBER_OF_DIGITS>
        35
      </NUMBER_OF_DIGITS>
      <BARCODE_TYPE>
        32
      </BARCODE_TYPE>
    </DAT>
  </STATE>
</LEUZE_XML>
```

3.1.4 Justage-Modus

DCR 200i

Aktiviert bzw. deaktiviert den Justage-Modus zur einfacheren Montage und Ausrichtung des Geräts. Nach Aktivierung der Funktion gibt der Codeleser ständig Status-Informationen auf der Ethernet-Schnittstelle aus. Durch den Befehl wird der Codeleser so eingestellt, dass er laufend den gleitenden Mittelwert der letzten 10 Bildaufnahmen in [%] und das Decodierergebnis ausgibt. Anhand dieser Werte kann eine Aussage über die Lesequalität bzw. Decodierqualität getroffen werden. Der Justage-Modus muss über „stop“ deaktiviert werden.

HINWEIS	
	<p>Die Funktion Justage-Modus kann ebenfalls auf andere Weise aktiviert werden (z. B. Schalteingang oder Gerätetaste). In diesem Fall generiert der DCR 200i spontane STATE-Nachrichten an den XMLClient.</p>

IPS 200i

Funktion nicht verfügbar.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="adjust" a="1">start</CMD></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="adjust" a="1">
      start
    </CMD>
  </PUT>
</LEUZE_XML>
```

Antwort (eine oder mehrere Antworten):

DCR 200i:


```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="adjust" result="OK">
      Start
    </CMD>
  </STATE>
</LEUZE_XML>

<LEUZE_XML>
  <STATE v="adjustment.result" s="cur">
    <DAT>
      <BARCODE d="base64Binary">
        MmRfZGFkYSBdXT4gbWl0IDwhW0NEQVRBWYBNdXJrcyBdXT4=
      </BARCODE>
      <DECODE_QUALITY>
        70
      </DECODE_QUALITY>
    </DAT>
  </STATE>
</LEUZE_XML>
...
```

3.1.5 Teach

DCR 200i

Mit Aktivierung der Teach-Funktion wird der Codeinhalt eines vorliegenden Codes als Referenzcode einge-
lernt und im Gerät gespeichert. Die Codeart und Stellenanzahl des einzulernenden Referenzcodes muss
vorher über die Funktion Auto-Setup eingestellt werden.

HINWEIS	
	Die Funktion Teach kann ebenfalls auf andere Weise aktiviert werden (z. B. Schalteingang oder Gerätetaste). In diesem Fall generiert der DCR 200i spontane STATE-Nachrichten an den XML- Client.

IPS 200i

Funktion nicht verfügbar.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="teach" a="1"></CMD></PUT></LEUZE_XML>
```

```
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="teach" a="1">
    </CMD>
  </PUT>
</LEUZE_XML>
```

Antwort (eine oder mehrere Antworten):

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="teach" result="OK">
    </CMD>
  </STATE>
</LEUZE_XML>
```

```
<LEUZE_XML>
  <STATE v="teach.result" s="cur">
    <DAT>
      <BARCODE d="base64Binary">
        MmRfZGFkYSBdXT4gbWl0IDwhW0NEQVRBWYBNdXJrcyBdXT4=
      </BARCODE>
    </DAT>
  </STATE>
</LEUZE_XML>
```


3.1.6 Konfiguration abfragen (getConfig)

Die getConfig-Funktion fragt die Konfiguration des Sensors zum Zweck der späteren Wiederherstellung dieser Konfiguration mit setConfig an. Die Konfigurationsdatei befindet sich base-64-kodiert im Rumpf der Antwort. Dieser Rumpf ist zur Wiederherstellung der Konfiguration mit dem setConfig-Kommando zu senden.

Gültig für DCR 200i und IPS 200i.

Request:

```
<LEUZE_XML><GET s="cmd"><CMD n="getConfig" a="1"></CMD></GET></LEUZE_XML>
<LEUZE_XML>
  <GET s="cmd">
    <CMD n="getConfig" a="1">
    </CMD>
  </GET>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="getConfig" d="base64Binary" result="OK">
    ...
    Qk02xBIAAAAAADYEAAAoAAAAAAUAAMADAAABAAgAAAAAADAEgAAAAAAAAAAAAAAAAALgAvLy8AMDawA-
    DExMQAyMjIAMzMzADQ0NAA1NTUANjY2ADc3NwA4ODgAOTk5ADo6OgA7OzsAPDw8AD09PQA+Pj4APz8/
    AEBAQABBQUEAQkJCAENDQwBEREQARUVFAEZG
    ...
  </CMD>
</STATE>
</LEUZE_XML>
```

3.1.7 Konfiguration wiederherstellen (setConfig)

Die setConfig-Funktion sendet die Konfigurationsdatei zum verbundenen Sensor. Die Konfigurationsdatei ist hierbei base-64-kodiert.

Gültig für DCR 200i und IPS 200i.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="setConfig" a="1">...</CMD></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT s="cmd">
    <CMD a="1" n="setConfig" d="base64Binary">
    Qk02xBIAAAAAADYEAAAoAAAAAAUAAMADAAABAAgAAAAAADAEgAAAAAAAAAAAAAAAAALgAvLy8AMDawA-
    DExMQAyMjIAMzMzADQ0NAA1NTUANjY2ADc3NwA4ODgAOTk5ADo6OgA7OzsAPDw8AD09PQA+Pj4APz8/
    AEBAQABBQUEAQkJCAENDQwBEREQARUVFAEZG
    </CMD>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="setConfig" d="base64Binary" result="OK">
    </CMD>
  </STATE>
</LEUZE_XML>
```

3.2 Get Parameter

Zum Abfragen von Variablen dient eine GET-Message in deren v-Attribut der Name des Parameters angegeben ist. Hierbei wird zwischen Geräteparameter und Job-Parameter unterschieden. Auch wenn in den folgenden Beispielen Geräteparameter und Job-Parameter getrennt aufgeführt sind, ist das Handling für den Benutzer das Gleiche.

3.2.1 Beispiel: Abfragen des Etherhost UDP-Ports (Geräteparameter)

Request:

```
<LEUZE_XML><GET v="..."><DAT a="1"></DAT></GET></LEUZE_XML>
<LEUZE_XML>
  <GET v="EtherHostServer.ethhost_App.ethhost_udp.ethhost_udp_port">
    <DAT a="1">
      </DAT>
    </GET>
  </LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE v="EtherHostServer.ethhost_App.ethhost_udp.ethhost_udp_port">
    <DAT a="1" result="OK">10001</DAT>
  </STATE>
</LEUZE_XML>
```

3.2.2 Beispiel: Abfragen der Belichtungszeit = Exposure Time (Job-Parameter)

Die Belichtungszeit wird in μs angegeben.

Gültig für DCR 200i und IPS 200i. Beim IPS 200i gültig für das aktuell aktive Programm.

Request:

```
<LEUZE_XML><GET v="exp"><DAT a="1"></DAT></GET></LEUZE_XML>
<LEUZE_XML>
  <GET v="exp">
    <DAT a="1">
      </DAT>
    </GET>
  </LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE v="exp">
    <DAT a="1" result="OK">500</DAT>
  </STATE>
</LEUZE_XML>
```

3.2.3 Beispiel: Abfragen des Referenzcodes (Job-Parameter)

Gültig für DCR 200i.

Request:

```
<LEUZE_XML><GET v="vmcd"><DAT a="1"></DAT></GET></LEUZE_XML>
<LEUZE_XML>
  <GET v="vmcd">
    <DAT a="1">
      </DAT>
    </GET>
  </LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE v=" vmcd">
    <DAT a="1" result="OK">8765123411110</DAT>
  </STATE>
</LEUZE_XML>
```

3.3 Put Parameter

Zum Setzen von Variablen dient eine PUT-Message in deren v-Attribut der Name des Geräteparameters bzw. des Job-Parameters angegeben ist.

3.3.1 Beispiel: Setzen des Etherhost UDP-Ports (Geräteparameter), temporär

Request:

```
<LEUZE_XML><PUT v="..."><DAT a="1">10001</DAT></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT v="EtherHostServer.ethhost_App.ethhost_udp.ethhost_udp_port">
    <DAT a="1">10002</DAT>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE v="EtherHostServer.ethhost_App.ethhost_udp.ethhost_udp_port">
    <DAT a="1" result="OK">10002</DAT>
  </STATE>
</LEUZE_XML>
```

3.3.2 Beispiel: Setzen der Belichtungszeit = Exposure Time (Job-Parameter), temporär

Die Belichtungszeit wird in µs angegeben.

Gültig für DCR 200i und IPS 200i. Beim IPS 200i gültig für das aktuell aktive Programm.

Request:

```
<LEUZE_XML><PUT v="exp"><DAT a="1">600</DAT></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT v="exp">
    <DAT a="1">600</DAT>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE v="exp">
    <DAT a="1" result="OK">600</DAT>
  </STATE>
</LEUZE_XML>
```

3.3.3 Beispiel: Setzen der Belichtungszeit = Exposure Time (Job-Parameter), permanent

Die Belichtungszeit wird in μ s angegeben.

Gültig für DCR 200i und IPS 200i. Beim IPS 200i gültig für das aktuell aktive Programm.

Request:

```
<LEUZE_XML><PUT v="exp" s="perm"><DAT a="1">600</DAT></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT v="exp" s="perm">
    <DAT a="1">600</DAT>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE v="exp">
    <DAT a="1" result="OK">600</DAT>
  </STATE>
</LEUZE_XML>
```

3.3.4 Beispiel: Setzen des Referenzcodes (Job-Parameter), temporär

Gültig für DCR 200i.

Request:

```
<LEUZE_XML><PUT v="vmcd"><DAT a="1">HALLO</DAT></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT v="vmcd">
    <DAT a="1">12345</DAT>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE v="vmcd">
    <DAT a="1" result="OK">12345</DAT>
  </STATE>
</LEUZE_XML>
```

3.4 Weitere Beispiele

3.4.1 Einige Beispiele zum Setzen und Abfragen von „Etherhost-TCP-Enabled“

Setzen auf TRUE/FALSE nicht-permanent:

```
<LEUZE_XML><PUT
v="EtherHostServer.ethhost_App.ethhost_tcpip.ethhost_tcpip_enabled"><DAT
a="1">TRUE</DAT></PUT></LEUZE_XML>
```

```
<LEUZE_XML><PUT
v="EtherHostServer.ethhost_App.ethhost_tcpip.ethhost_tcpip_enabled"><DAT
a="1">FALSE</DAT></PUT></LEUZE_XML>
```

Setzen auf TRUE/FALSE permanent (die Bearbeitungszeit verlängert sich deutlich):

```
<LEUZE_XML><PUT
v="EtherHostServer.ethhost_App.ethhost_tcpip.ethhost_tcpip_enabled"
s="perm"><DAT a="1">TRUE</DAT></PUT></LEUZE_XML>
```

```
<LEUZE_XML><PUT
v="EtherHostServer.ethhost_App.ethhost_tcpip.ethhost_tcpip_enabled"
s="perm"><DAT a="1">FALSE</DAT></PUT></LEUZE_XML>
```

3.5 Geräteabfragen

Gültig für DCR 200i und IPS 200i.

3.5.1 Get Status

Abfrage des aktuellen Gerätestatus

Mode: **PM** (process mode) / **SM** (service mode)
 ErrorState: **0**: Kein Fehler, **1**: Fehler vorhanden. Der Error-State gibt an, ob sich im Fehlerspeicher mindestens ein Error befindet.
 Busy: **1**: Beschäftigt, **0**: nicht beschäftigt.

Request:

```
<LEUZE_XML><GET s="cmd"><CMD n="getStatus" a="1"></CMD></GET></LEUZE_XML>
```

```
<LEUZE_XML>
  <GET s="cmd">
    <CMD n="getStatus" a="1">
      </CMD>
    </GET>
  </LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="getStatus" result="OK">
      <mode>PM</mode>
      <errorState>0</errorState>
      <busy>0</busy>
    </CMD>
  </STATE>
</LEUZE_XML>
```

3.5.2 Version Request

Mit diesem Request fragen Sie die aktuelle Geräteversion ab.

Request:

```
<LEUZE_XML><GET s="cmd"><CMD n="versionRequest" a="1"></CMD></GET></LEUZE_XML>
```

```
<LEUZE_XML>
  <GET s="cmd">
    <CMD n="versionRequest" a="1">
      </CMD>
    </GET>
  </LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="versionRequest" result="OK">
      <iox-app-sw>V 1.0.0</iox-app-sw>
    </CMD>
  </STATE>
</LEUZE_XML>
```

3.5.3 setDefaultLogin

Das Kommando dient dazu, eine Defaultrolle (charakterisiert durch einen Benutzernamen und dessen Passwort) festzulegen, die webConfig beim Anmelden beim DCR 200i / IPS 200i einnehmen soll.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="setDefaultLogin" a="1">...</CMD></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="setDefaultLogin" a="1">
      <USER>UserName</USER>
      <PASS>PlainTextPW</PASS>
    </CMD>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="setDefaultLogin" result="OK">
      <USER>UserName</USER>
      <PASS>PlainTextPW</PASS>
    </CMD>
  </STATE>
</LEUZE_XML>
```

Aufheben der Default-Rolle:

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="setDefaultLogin" a="1">...</CMD></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="setDefaultLogin" a="1">
      <USER></USER>
      <PASS></PASS>
    </CMD>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="setDefaultLogin" result="OK">
      <USER></USER>
      <PASS></PASS>
    </CMD>
  </STATE>
</LEUZE_XML>
```

3.6 Steuerung und Bildabfrage

3.6.1 Scan Activate

Dieser Parameter aktiviert die im Gerät gespeicherte Konfiguration (Lesung).

Gültig für DCR 200i und IPS 200i. Beim IPS 200i gültig für das aktuell aktive Programm.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="scanActivate" a="1">1</CMD></PUT></LEUZE_XML>
```

```
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="scanActivate" a="1">
      1
    </CMD>
  </PUT>
</LEUZE_XML>
```

Antwort (eine oder mehrere Antworten):

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="scanActivate" result="OK">
      1
    </CMD>
  </STATE>
</LEUZE_XML>
```

```
<LEUZE_XML>
  <STATE v="barcode.result" s="cur">
    <DAT>
      <BARCODE d="base64Binary">
        MmRfZGFkYSBdXT4gbWl0IDwhW0NEQVRBWYBNdXJrcyBdXT4=
      </BARCODE>
    </DAT>
  </STATE>
</LEUZE_XML>
```

3.6.2 Scan Deactivate

Dieser Parameter deaktiviert die im Gerät gespeicherte Konfiguration (Lesung).

Gültig für DCR 200i und IPS 200i. Beim IPS 200i gültig für das aktuell aktive Programm.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="scanActivate" a="1">0</CMD></PUT></LEUZE_XML>
```

```
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="scanActivate" a="1">
      0
    </CMD>
  </PUT>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="scanActivate" result="NOK">
      0
    </CMD>
  </STATE>
</LEUZE_XML>
```

3.6.3 Get Image

Dieser Request fordert das aktuell aufgenommene Bild an.

Gültig für DCR 200i und IPS 200i. Beim IPS 200i gültig für das aktuell aktive Programm.

Request:

```
<LEUZE_XML><GET s="cmd"><CMD n="getImage" a="1"></CMD></GET></LEUZE_XML>
```

```
<LEUZE_XML>
  <GET s="cmd">
    <CMD n="getImage" a="1">
    </CMD>
  </GET>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="getImage" d="base64Binary" result="OK">
```

```
    ...
    Qk02xBIAAAAAADYEAAAoAAAAAAUAMADAAAABAaGAAAAAADAEgAAAAAAAAAAAA
    AAAAAAAAAAAAAAEBAQACAgIAAwMDAAQEBAAFBQUABgYGAAcHBwAICAgACQkJAAoK
    CgALCwsADAwMAA0NDQAODg4ADw8PABAQEAAAREEAehISABMTEwAUFBQAFRUVABYW
    FgAXFxcAGBgYABkZGQAaGhoAGxsbABwcHAAAdHR0AHh4eAB8fHwAgICAAISEhACIi
    IgAjIyMAJCQkACU1JQAmJiYAJycnACgoKAApKSkAKioqACsrKwAsLCwALS0tAC4u
    LgAvLy8AMDawADExMQAyMjIAMzMzADQ0NAA1NTUANjY2ADc3NwA4ODgAOTk5ADo6
    OgA7OzsAPDw8AD09PQA+Pj4APz8/AEBAQABBQUEAQkJCAENDQwBEREQARUVFAEZG
```

```
    ...
    </CMD>
  </STATE>
</LEUZE_XML>
```

„...“ steht für den Base64-kodierten Inhalt des Bildes, dessen Format eine Bitmap ist. Ist ein Bild verfügbar, dann ist das Resultat ein „OK“, ansonsten ein „NOK“.

3.6.4 Programmabfrage (activeJob)

Diese Abfrage fordert das aktuell aktive Programm (Selektions-ID) an.

Gültig für IPS 200i.

Request:

```
<LEUZE_XML><GET s="cmd"><CMD n="activeJob" a="1"></CMD></GET></LEUZE_XML>
```

```
<LEUZE_XML>
  <GET s="cmd">
    <CMD n="activeJob" a="1">
    </CMD>
  </GET>
</LEUZE_XML>
```

Die 1 im Kommando steht dabei für die (Job-) Selektions-ID.

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="activeJob" result="OK">
      1
    </CMD>
  </STATE>
</LEUZE_XML>
```


3.6.5 Programmumschaltung (activeJob)

Dieser Befehl aktiviert die Umschaltung auf das gewünschte Programm (Selektions-ID) im Sensor.
Gültig für IPS 200i.

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="activeJob" a="1">1</CMD></PUT></LEUZE_XML>
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="activeJob" a="1">
      1
    </CMD>
  </PUT>
</LEUZE_XML>
```

Die 1 im Kommando steht dabei für die (Job-) Selektions-ID.

Antwort:

```
<LEUZE_XML>
  <STATE s="cmd">
    <CMD a="1" n="activeJob" result="OK">
      1
    </CMD>
  </STATE>
</LEUZE_XML>
```

3.6.6 Subscribe / Unsubscribe

Mit Subscribe bzw. Unsubscribe kann gesteuert werden, ob der Sensor bei Parameteränderungen spontane Nachrichten mit Informationen zu den Änderungen an den XML-Client sendet. Die Zustände von Subscribe/Unsubscribe werden nicht nullspannungssicher gespeichert. Nach dem Start des Sensors befindet es sich immer im Zustand "Unsubscribe".

Gültig für DCR 200i und IPS 200i.

Subscribe

Request:

```
<LEUZE_XML><SUBS a="1" v="*"></SUBS></LEUZE_XML>
<LEUZE_XML>
  <SUBS a="1" v="*">
  </SUBS>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE a="1" v="*">
    subscribed
  </STATE>
</LEUZE_XML>
```

Nach einem erfolgreichen Subscribe werden alle Parameteränderungen als spontane Nachricht im folgenden Format gesendet:

Beispiel einer spontanen Nachricht bei Änderung von Job-Parametern (hier Exposuretime)

```
<LEUZE_XML>
  <STATE v="exp">
    <DAT>
      164
    </DAT>
  </STATE>
</LEUZE_XML>
```

Beispiel einer spontanen Nachricht bei Änderung von Applikations-Parametern (hier Betriebsmodus)

```
<LEUZE_XML>
  <STATE v="ToolGeneral.RunMode">
    <DAT>
      RunMode_Tiggered
    </DAT>
  </STATE>
</LEUZE_XML>
```

Unsubscribe

Request:

```
<LEUZE_XML><UNSUBS a="1" v="*"></SUBS></LEUZE_XML>
```

```
<LEUZE_XML>
  <UNSUBS a="1" v="*">
  </UNSUBS>
</LEUZE_XML>
```

Antwort:

```
<LEUZE_XML>
  <STATE a="1" v="*">
    unsubscribed
  </STATE>
</LEUZE_XML>
```

Nach einem erfolgreichen Unsubscribe werden bei Parameteränderungen keine spontanen Nachricht mehr gesendet:

3.7 Fehlerfälle

3.7.1 Abgelehnter Request

Request:

```
<LEUZE_XML><PUT s="cmd"><CMD n="getStatus" a="1">service</CMD></PUT></LEUZE_XML>
```

```
<LEUZE_XML>
  <PUT s="cmd">
    <CMD n="getStatus" a="1">
    </CMD>
  </PUT>
</LEUZE_XML>
```

Das Kommando wird abgelehnt, weil getStatus ein "GET" anstelle des "PUT" erwartet.

Antwort:

```
<LEUZE_XML result="rejected">
  <PUT s="cmd">
    <CMD a="1" n="getStatus">
    </CMD>
  </PUT>
</LEUZE_XML>
```

Grund der Ablehnung: getStatus ist ein Request, der über <GET> gesendet werden muss.

3.7.2 Fehlgeschlagenes Kommando

Es wurde versucht ein gültiges Kommando auszuführen, aber der Versuch scheiterte.

Request:

```
<LEUZE_XML><GET s="cmd"><CMD n="getImage" a="1"></CMD></GET></LEUZE_XML>
```

```
<LEUZE_XML>  
  <GET s="cmd">  
    <CMD n="getImage" a="1">  
    </CMD>  
  </GET>  
</LEUZE_XML>
```

Das Kommando wird abgelehnt, wenn kein Bild verfügbar ist. Ein weiterer Grund für die nicht Durchführbarkeit kann die fehlende aktive Session sein.

Antwort:

```
<LEUZE_XML>  
  <STATE s="cmd">  
    <CMD a="1" n="getImage" result="NOK">  
    </CMD>  
  </STATE>  
</LEUZE_XML>
```

3.7.3 Timeout nach unvollständigem Kommando

Wird ein unvollständiges Kommando an das Gerät gesendet oder geht auf der Übertragungsstrecke ein Teil der Nachricht verloren, so befindet sich die geräteinterne Kommandoverarbeitung in einem undefinierten Zustand.

Beispiel für ein unvollständiges Kommando:

```
<LEUZE_XML>  
  <PUT s="cmd">  
    <CMD n="
```

Werden jetzt weitere Kommandos an das Gerät geschickt, können diese normalerweise nicht verarbeitet werden.

Damit dies nicht geschieht setzt ein Timeout die geräteinterne Kommandoverarbeitung im Fehlerfall nach wenigen Sekunden zurück und verwirft unvollständige Kommandos.