

DE

VisionApp 360

Software



Schnittstellenprotokoll

Inhaltsverzeichnis

1. Einleitung	4
2. LIMA-Protokoll	4
2.1 Verbindungsaufbau über TCP/IP	4
2.1.1 Control Unit mit VisionApp 360 Applikation	4
2.2 Allgemeine Informationen zur LIMA-Kommunikation	4
3. Set-Befehle.....	5
3.1 Befehle für Sensorgruppen.....	5
3.1.1 Messung starten	5
3.1.2 Messung stoppen.....	5
3.1.3 Resynchronisierung.....	5
3.1.4 Kalibrierung durchführen.....	6
3.1.5 Kalibrierung löschen.....	6
3.1.6 Kalibrierung ein-/ ausschalten.....	6
3.2 Geräteeinstellungen	7
3.2.1 Belichtungszeit einstellen	7
3.2.2 Messintervall einstellen.....	7
3.2.3 Laser ein-/ausschalten	7
3.2.4 SyncOut Pulsbreite einstellen	7
3.3 Region of Interest (ROI)	8
3.3.1 ROI Breite in X	8
3.3.2 ROI Offset in X	8
3.3.3 ROI Step in X	8
3.3.4 ROI Höhe in Z	8
3.3.5 ROI Offset in Z	9
3.4 Befehle für Modulsteuerung – Modul VisionApp 360	9
3.4.1 TCP/IP-Server starten	9
3.4.2 TCP/IP-Server stoppen	9
3.4.3 TCP/IP-Server Port ändern	9

4. Beispiel-Setup	10
4.1 Eingabe Sensorgruppe und verwendete Sensoren	10
4.2 Aktivierung des „Job Manager TCP/IP-Server“	10
4.3 Befehle	11
4.3.1 Befehle für Sensorgruppen	11
4.3.2 Befehle für Geräte.....	11
4.3.3 Befehle für ROI	11
4.3.4 Befehle für Module	11
5. Get-Befehle	12
6. Datenformat	16
7. Änderungsverzeichnis	17

1. Einleitung

Diese Dokumentation beschreibt den Aufbau und die Funktion der TCP-Befehle zur Steuerung und Einstellung der weCat3D-Profilesensoren unter Verwendung der VisionApp 360 Software.

Die Befehle werden über ein offenes Protokoll (LIMA-Protokoll) auf XML-Basis unter Verwendung des Ports 62232 gesendet.

Um eine Verbindung mit der VisionApp 360 aufzubauen, muss der „Job Manager TCP/IP Server“ in den „Preferences“ aktiviert werden (siehe Betriebsanleitung der VisionApp 360 Software).



HINWEIS!

Die Groß-/Kleinschreibung ist zu beachten.

2. LIMA-Protokoll

Zunächst die Geräteeinstellungen und das Projekt über die VisionApp 360 Software einrichten und abspeichern. Anschließend kann das LIMA-Protokoll genutzt werden, um bestimmte Werte vom Gerät abzufragen bzw. am Gerät zu ändern.

2.1 Verbindungsaufbau über TCP/IP

2.1.1 Control Unit mit VisionApp 360 Applikation

Eine TCP/IP-Verbindung zur VisionApp 360 Applikation aufbauen, die auf der Control Unit läuft:

- IP-Adresse der VisionApp 360 Applikation
- Port: 62232

Beispiel mit Standareinstellungen einer Control Unit:

- IP-Adresse: 192.168.100.252
- Port: 62232

2.2 Allgemeine Informationen zur LIMA-Kommunikation

Folgende allgemeine Punkte müssen bei der LIMA-Kommunikation beachtet werden:

- Die Auswertung der Antworten auf einen LIMA-Befehl wird empfohlen. Je nach Befehl kann es einige Sekunden dauern bis eine Antwort erhalten wird.
- LIMA-Befehle dürfen nur sequenziell an die VisionApp 360 Applikation geschickt werden. Erst wenn die Antwort zum vorherigen Befehl erhalten wurde, darf der nächste Befehl verschickt werden.
- Die Konsistenz der Daten muss durch den Ablauf der Kommunikation sichergestellt werden. Hierfür wird die Auswertung des Durchlaufzählers empfohlen, um festzustellen, ob z. B. neue Ergebnisse vorliegen.

Fehlerdiagnose zu Antworten auf LIMA-Befehle:

Antwort auf LIMA-Befehl	Ursache	Mögliche Lösung
Device Busy	Der LIMA-Befehl kann nicht verarbeitet werden, da das Gerät mit der Bearbeitung eines anderen LIMA-Befehls beschäftigt ist.	<ul style="list-style-type: none"> Den LIMA-Befehl zu einem späteren Zeitpunkt erneut senden.
Lima Error on File Read	Das Projekt konnte nicht geladen werden, da das entsprechende Projekt im Projektordner nicht verfügbar ist.	<ul style="list-style-type: none"> Sicher stellen, dass ein Projekt mit dem entsprechenden Namen im Projektordner verfügbar ist. Projektname im LIMA-Befehl überprüfen.
Invalid Path	Der LIMA-Befehl enthält eine Pfadangabe, die im VisionApp 360 Projekt nicht verfügbar ist.	<ul style="list-style-type: none"> Die korrekte Pfadangabe im LIMA-Befehl einfügen.

3. Set-Befehle

3.1 Befehle für Sensorgruppen

Die nachfolgenden Befehle haben Auswirkungen auf die ausgewählte Sensorgruppe. Eine Sensorgruppe vereint alle Sensoren, die in der Anwendung kombiniert werden sollen. Ein Beispiel dazu finden Sie in [Kap. 4](#).

3.1.1 Messung starten

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Start" VALUE="1" /></code>
Beschreibung	Startet die Messung der ausgewählten Sensorgruppe.

3.1.2 Messung stoppen

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Start" VALUE="0" /></code>
Beschreibung	Stoppt die Messung der ausgewählten Sensorgruppe.

3.1.3 Resynchronisierung

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Resync" VALUE="1" /></code>
Beschreibung	Resynchronisiert die ausgewählte Sensorgruppe. Die Sensoren werden automatisch gestoppt und im Anschluss wieder gestartet.

3.1.4 Kalibrierung durchführen

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Calibration Start" VALUE="1" /></code>
Beschreibung	Die Kalibrierung der Sensoren wird für die ausgewählte Sensorgruppe durchgeführt.

3.1.5 Kalibrierung löschen

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Calibration Delete" VALUE="1" /></code>
Beschreibung	Die Kalibrierung der Sensoren wird für die ausgewählte Sensorgruppe gelöscht.

3.1.6 Kalibrierung ein-/ ausschalten

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Calibration Enable" VALUE="x" /></code>
Parameter	Werte für X: 0: Kalibrierung aus 1: Kalibrierung an
Beschreibung	Die Kalibrierung der Sensoren wird für die ausgewählte Sensorgruppe ein- oder ausgeschaltet. Sie kann nur eingeschaltet werden, wenn sie zuvor durchgeführt wurde. Nach dem Ausschalten kann die Kalibrierung wieder aktiviert werden. Die Kalibrierung wird nicht gelöscht.

3.2 Geräteeinstellungen

Die nachfolgenden Befehle haben Auswirkungen auf die ausgewählten Sensoren. Ein Beispiel dazu finden Sie in [Kap. 4](#).

3.2.1 Belichtungszeit einstellen

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetExposureTime" VALUE="X" /></code>		
Parameter	Bereich für den Wert X: 26...100.000	Default:	150
Beschreibung	Stellt den Wert für die Belichtungszeit in μs ein.		

3.2.2 Messintervall einstellen

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetAcquisitionLineTime" VALUE="X" /></code>		
Parameter	Bereich für den Wert X: 166...100.000	Default:	MLWL: 5714 MLSL: 5000
Beschreibung	<p>Stellt den Zeitwert zwischen zwei aufeinander folgenden Profilen in μs ein. Dieser Befehl hat nur Auswirkungen im internen Trigger Modus.</p> <p>166 μs = 6000 Hz</p> <p>Beispiel: MLWL: 5714 μs = 175 Hz MLSL: 5000 μs = 200 Hz</p> <div style="display: flex; align-items: center;">  <div> <p>HINWEIS!</p> <p>Um eine höhere Messfrequenz zu erreichen, müssen die ROI Einstellungen verringert werden.</p> </div> </div>		

3.2.3 Laser ein-/ausschalten

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetLaserDeactivated" VALUE="X" /></code>		
Parameter	Werte für X: 0: Laser aus 1: Laser an	Default:	0
Beschreibung	Softwarebefehl, um den Laser ein- und auszuschalten. Falls X = 1 (enabled), bleiben alle weiteren Signale an den E/As ohne Wirkung.		

3.2.4 SyncOut Pulsbreite einstellen

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetSyncOut" VALUE="X" /></code>		
Parameter	Bereich für den Wert X: 0...100000	Default:	1000
Beschreibung	Definiert die Pulsbreite des SyncOut-Signals für den E/A-SyncOut in μs .		

3.3 Region of Interest (ROI)

3.3.1 ROI Breite in X

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetROI1WidthX" VALUE="X" /></code>		
Parameter	Bereich für den Wert X: MLWL: 32...1280	Default:	MLWL: 1280
			MLWL: 2048
Beschreibung	Definiert die Anzahl der Kamerazeilen in X, die ausgelesen werden sollen. MLWL: Die Auswahl hat Auswirkungen auf die einstellbare Messfrequenz und die Netzwerkauslastung (Bandbreite). Die Einstellung erfolgt in 16er-Schritten. MLWL: Die Auswahl hat Auswirkungen auf die Netzwerkauslastung (Bandbreite). Die Einstellung erfolgt in 1er-Schritten.		

3.3.2 ROI Offset in X

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetROI1OffsetX" VALUE="X" /></code>		
Parameter	Bereich für den Wert X: MLWL: 0...1279	Default:	0
			MLWL: 0...2047
Beschreibung	Definiert den Offset des ROI in X-Richtung. Der Offset gibt den Versatz zur ersten Linie an. MLWL: Die Einstellung erfolgt in 16er-Schritten. MLWL: Die Einstellung erfolgt in 1er-Schritten.		

3.3.3 ROI Step in X

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetROI1StepX" VALUE="X" /></code>		
Parameter	Bereich für den Wert X: MLWL: no function	Default:	0
			MLWL: 0...2047
Beschreibung	Gibt die Schrittweite der ausgelesenen Pixel in X an. MLWL: nicht möglich MLWL: In 1er-Schritten einstellbar		

3.3.4 ROI Höhe in Z

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetROI1HeightZ" VALUE="X" /></code>		
Parameter	Bereich für den Wert X: MLWL: 35...1280	Default:	MLWL: 1280
			MLWL: 2048
Beschreibung	Definiert die Anzahl der Kamerazeilen in Z, die ausgelesen werden sollen. Die Auswahl hat Auswirkungen auf die einstellbare Messfrequenz.		

3.3.5 ROI Offset in Z

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="Sensorgruppenbezeichnung" PATH="Sensorbezeichnung SetROI1OffsetZ" VALUE="X" /></code>		
Parameter	Bereich für den Wert X: MLSL: 0...1022 MLWL: 0...2046	Default:	0
Beschreibung	Definiert den Offset des ROI in Z-Richtung. Der Offset gibt den Versatz zur ersten Linie an.		

3.4 Befehle für Modulsteuerung – Modul VisionApp 360

3.4.1 TCP/IP-Server starten

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="VisionApp" PATH="Start" VALUE="1" /></code>
Beschreibung	Startet den TCP/IP-Server, um die Profildaten (s. Kap. 6) aus der Applikation zu erhalten. Die Daten werden standardmäßig über den Port 63333 übertragen.

3.4.2 TCP/IP-Server stoppen

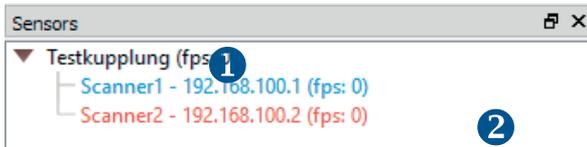
Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="VisionApp" PATH="Start" VALUE="0" /></code>
Beschreibung	Stoppt den TCP/IP-Server.

3.4.3 TCP/IP-Server Port ändern

Befehl	<code><LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="VisionApp" PATH="tcp_ip_api.Port" VALUE="Portadresse" /></code>
Beschreibung	Ändert den Port des TCP/IP-Servers.  ACHTUNG! Nicht auf Port 62232 ändern!

4. Beispiel-Setup

4.1 Eingabe Sensorgruppe und verwendete Sensoren



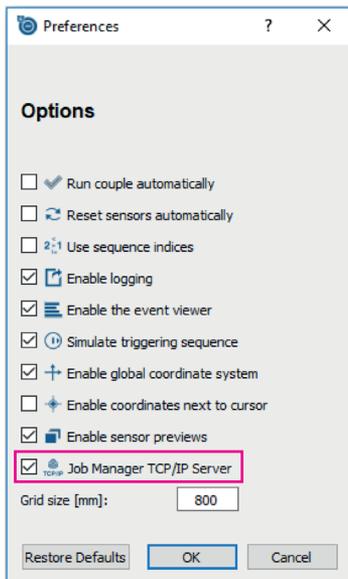
- ① Sensorgruppenbezeichnung (hier: couple), frei wählbar
- ② Sensorbezeichnungen (hier: Sensor1, Sensor2), frei wählbar



HINWEIS!

Die Bezeichnungen für Sensorgruppen und Sensoren dürfen keine Leerzeichen enthalten.

4.2 Aktivierung des „Job Manager TCP/IP-Server“



In den „Preferences“ muss bei „Job Manager TCP/IP-Server“ ein Häkchen gesetzt werden (siehe dazu die Betriebsanleitung der VisionApp 360 Software).

Der Job Manager öffnet den Port 62232, um Befehle an die Applikation schicken zu können.

Anschließend muss eine TCP-Verbindung mit der Anwendung aufgebaut werden. Verwenden Sie dafür die IP-Adresse des Gerätes, auf dem die VisionApp 360 installiert ist bzw. die Kommunikation stattfinden soll.

Öffnen Sie nun den Port 62232 und senden Sie die Befehle als Klartext wie unten beschrieben.

4.3 Befehle

4.3.1 Befehle für Sensorgruppen

Messung starten:

```
<LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="couple" PATH="Start" VALUE="1" />
```

Messung stoppen:

```
<LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="couple" PATH="Start" VALUE="0" />
```

4.3.2 Befehle für Geräte

Belichtungszeit an Sensor1 auf 500 μ s einstellen:

```
<LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="couple" PATH="Sensor1 SetExposureTime" VALUE="500" />
```

Laser an Sensor2 ausschalten:

```
<LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="couple" PATH="Sensor2 SetLaserDeactivated" VALUE="0" />
```

4.3.3 Befehle für ROI

Höhe Z auf 1200 Zeilen an Sensor1 einstellen:

```
<LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="couple" PATH="Sensor1 SetROI-1HeightZ" VALUE="1200" />
```

Offset Z auf 500 Zeilen an Sensor1 einstellen:

```
<LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="couple" PATH="Sensor1 SetROI-1HeightZ" VALUE="1200" />
```

4.3.4 Befehle für Module

TCP/IP-Server starten:

```
<LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="VisionApp" PATH="Start" VALUE="1" />
```

TCP/IP-Server Port ändern auf 6000:

```
<LIMA CMD="Module_SetNode" DIR="Request" MODULENAME="VisionApp" PATH="tcp_ip_api.Port" VALUE="12345" />
```

5. Get-Befehle

Über LIMA Befehle ist es möglich, die Parameter der Sensoren nicht nur zu schreiben, sondern auch auslesen. Dazu wird ein Befehl an das gewünschte Element des Couples geschickt und der Client erhält die entsprechende Antwort. Sollte das Gerät nicht antworten, wird das als Fehler an den Client weitergegeben. Die Get-Befehle sind ähnlich wie die Set-Befehle gemäß folgendem Prinzip aufgebaut:

```
<LIMA CMD="Module_GetState" DIR="Request" MODULENAME="COUPLE NAME" PATH="SENSOR NAME COMMAND" />
```

- COUPLE NAME Name des Couples
- SENSOR NAME Name des Sensors im Couple
- COMMAND gewünschter Get-Befehl aus aktueller weCat3D SDK (siehe HINWEIS unten)

Beispiel 1:

Anfrage der aktuellen Belichtungszeit des Sensors mit Namen "s0" aus Couple "c0":

```
<LIMA CMD="Module_GetState" DIR="Request" MODULENAME="c0" PATH="s0 GetExposureTime" />
```

Antwort von VisionApp 360, falls OK:

```
<LIMA CMD="Module_GetState" DIR="ReplyOk" MODULENAME="c0" PATH="s0" "GetExposureTime" VALUE="149" />
```

Antwort von VisionApp 360, falls nicht OK (Fehler):

```
<LIMA CMD="Module_GetState" DIR="ReplyError" MODULENAME="c0" PATH="s0" "GetExposureTime" VALUE="ERROR" />
```

Beispiel 2:

Anfrage der aktuellen Temperatur des Sensors mit Namen "s0" und Couple "c0":

```
<LIMA CMD="Module_GetState" DIR="Request" MODULENAME="c0" PATH="s0 GetTemperature" />
```

Antwort von VisionApp 360, falls OK:

```
<LIMA CMD="Module_GetState" DIR="ReplyOk" MODULENAME="c0" PATH="s0" "GetTemperature" VALUE="31" />
```

Antwort von VisionApp 360, falls nicht OK (Fehler):

```
<LIMA CMD="Module_GetState" DIR="ReplyError" MODULENAME="c0" PATH="s0" "GetTemperature" VALUE="ERROR" />
```



HINWEIS!

Details zu den Get-Befehlen finden Sie auch in der weCat3D Interface Description (Kap. 7) im Downloadbereich des Sensors.

Die nachfolgende Tabelle zeigt die aktuellen ASCII Befehle für die Parameter, die mit Hilfe der in Kap. 5 beschriebenen Befehlsstruktur in der VisionApp 360 abgelesen werden können.

ASCII Befehl	Bemerkungen
GetPictureCounter	
GetSystemTime	in μ s
GetStatisticDataUserData	
GetOrderNumber	
GetProductVersion	
GetProducer	
GetFirmwareVersion	
GetSerialNumber	
GetMAC	
GetWorkingRangeZStart	
GetWorkingRangeZEnd	
GetFieldWidthXStart	
GetFieldWidthXEnd	
GetPixelXMax	
GetPixelZMax	
GetOnOffCounter	
GetOnTimeCounter	
GetLinInfo	falls der Sensor kalibriert ist
GetUserString	
GetHeartBeat	
GetSocketConnectionTimeout	
GetIOState	bit0: E/A 1 bit1: E/A 2 bit2: E/A 3 bit3: E/A 4
GetEncoderHTL	
GetEncoderTTL	
GetTemperature	
GetScannerState	bit0: Profile scanner OK bit1: ExposureTime OK bit2: LaserONTime OK bit3: ungenutzt bit4: ungenutzt bit5: Messrate zu hoch bit6: ungenutzt bit7: ungenutzt
GetSignalEnable	Die Anzahl der Signale in jedem Scan, siehe Funktion SetSignalEnable
GetSignalContentZ	
GetSignalContentStrength	
GetSignalContentWidth	

ASCII Befehl	Bemerkungen
GetSignalContentReserved	
GetSignalWidthMin	
GetSignalWidthMax	
GetSignalStrengthMin	
GetSignalSelection	
GetAcquisitionLineTime	
GetCameraRunning	
GetTriggerSource	
GetTriggerAmountProfilesY	
GetAmountProfilesY	
GetTriggerEncoderStep	
GetTriggerDelay	
GetExposureTime	
GetLaserActive	
GetROI1WidthX	
GetROI1OffsetX	
GetROI1StepX	
GetROI1HeightZ	
GetROI1OffsetZ	
GetSyncOut	
GetSyncOutDelay	
GetEncoderTriggerFunction	
GetEncoderCountDirection	
GetEA1Function	
GetEA1FunctionLaserOff	
GetEA1FunctionProfileEnable	
GetEA1FunctionResetCounter	
GetEA1InputFunction	
GetEA1InputLoad	
GetEA1Output	
GetEA1OutputFunction	
GetEA1ResetCounterRepeat	
GetEA1ResetCounterSignaledge	
GetEA1ResetCounterBaseTimeCounter	
GetEA1ResetCounterPictureCounter	
GetEA1ResetCounterEncoderHTL	
GetEA1ResetCounterEncoderTTLRS422	
GetEA2Function	
GetEA2FunctionLaserOff	
GetEA2FunctionProfileEnable	
GetEA2FunctionResetCounter	
GetEA2InputFunction	

ASCII Befehl	Bemerkungen
GetEA2InputLoad	
GetEA2Output	
GetEA2OutputFunction	
GetEA2ResetCounterRepeat	
GetEA2ResetCounterSignaledge	
GetEA2ResetCounterBaseTimeCounter	
GetEA2ResetCounterPictureCounter	
GetEA2ResetCounterEncoderHTL	
GetEA2ResetCounterEncoderTTLRS422	
GetEA3Function	
GetEA3FunctionLaserOff	
GetEA3FunctionProfileEnable	
GetEA3FunctionResetCounter	
GetEA3InputFunction	
GetEA3InputLoad	
GetEA3Output	
GetEA3OutputFunction	
GetEA3ResetCounterRepeat	
GetEA3ResetCounterSignaledge	
GetEA3ResetCounterBaseTimeCounter	
GetEA3ResetCounterPictureCounter	
GetEA3ResetCounterEncoderHTL	
GetEA3ResetCounterEncoderTTLRS422	
GetEA4Function	
GetEA4FunctionLaserOff	
GetEA4FunctionProfileEnable	
GetEA4FunctionResetCounter	
GetEA4InputFunction	
GetEA4InputLoad	
GetEA4Output	
GetEA4OutputFunction	
GetEA4ResetCounterRepeat	
GetEA4ResetCounterSignaledge	
GetEA4ResetCounterBaseTimeCounter	
GetEA4ResetCounterPictureCounter	
GetEA4ResetCounterEncoderHTL	
GetEA4ResetCounterEncoderTTLRS422	
GetEAFunctionInputCounter	
GetSettings=0	Gibt die gespeicherten Einstellungen des Profilsensors im Default als XML-Struktur zurück

ASCII Befehl	Bemerkungen
GetSettings=1	Gibt die gespeicherten Einstellungen des Profilsensors in Set1 als XML-Struktur zurück
GetSettings=2	Gibt die gespeicherten Einstellungen des Profilsensors in Set2 als XML-Struktur zurück
GetSettings=3	Gibt die aktuellen Einstellungen des Profilsensors als XML-Struktur zurück

6. Datenformat

Beschreibung	Offset in bytes	Type (bytes)	Wert
Container ID des Multisensors	0	uint32 (4)	0x021b0100
Container Größe (gesamtes Datenpaket). Die Größe der Punktdatenblöcke der Sensoren ist im vordefinierten Wert nicht enthalten.	4	uint32 (4)	0x4b0
General ID des Multisensors	8	uint32 (4)	0x021b0200
Größe des gesamten Datenpakets	12	uint32 (4)	0x84
numberOfSources: Anzahl der Quellen (Sensoren) in einer Sensorgruppe	16	uint32 (4)	0x0
User string, reserviert für User (nicht genutzt)	20	char[128] (128)	"VisionApp"
Zeitstempel der aktuellen Daten (nicht genutzt)	148	uint32 (4)	0x0
Part ID des Multisensors	152	uint32 (4)	0x021b0400
Größe des Teildatenblocks. Die Größe der Punktdatenblöcke der Sensoren ist im vordefinierten Wert nicht enthalten.	156	uint32 (4)	0x418
Header ID des Multisensors	160	uint32 (4)	0x021b0402
Header Größe	164	uint32 (4)	0x408
scannerPoints: Anzahl der Punkte auf einer Sensor-Achse	168	uint32 (4)	0x0
scannerDataFormat: Sensor Datenformat (8 Bytes)	172	uint32 (4)	0x0
Sensor Bildzähler	176	uint32 (4)	0x0
scannerIntensityDataFormat: Datenformat der Sensor-Intensität (4 Bytes)	180	uint32 (4)	0x0
Encoder	184	uint32 (4)	0x0
Reserviert für die gesamten Header-Daten. Ergibt mit den fünf vorangegangenen Dateneinheiten 1024 Bytes (Offset 168-1192).	188	uint32 (1004)	0x0
Punktdaten ID	1192	uint32 (4)	0x021b0403
Punktdaten-Größe ohne Sensor-Daten (fester Wert: 8 Bytes)	1196	uint32 (4)	0x8

Beschreibung	Offset in bytes	Type (bytes)	Wert
<p>Sensor1 Daten: Format des Sensors: x Array, z Array mit scannerPoints des scannerDataFormat und Intensitätsdaten des Formats scannerIntensityDataFormat. Der Bereich enthält $n = \text{scannerPoints} \times (2 \times \text{scannerDataFormat} + \text{scannerIntensityDataFormat})$ Bytes für einen Sensor. Die Daten sind nicht an die Objektinstanz gekoppelt, daher gibt es hierzu keine Daten.</p>	1200	float[scannerPoints] or double [scannerPoints] (n)	
<p>Sensor2 Daten: gleiches Format wie bei Sensor1. Es gibt $k = \text{numberOfSources}$ Sensoren pro Verbindung</p>	$1200 + n$	float[scannerPoints] or double [scannerPoints] (n)	
Zyklische Blockprüfungs ID des Multisensors	$1200 + n \times k$	uint32 (4)	0x021b01ff
Größe des CRC-Blocks	$1204 + n \times k$	uint32 (4)	0xc
CRC Daten (es wird der Dummy-Wert des Bildzählers verwendet).	$1208 + n \times k$	uint32 (4)	0x0

7. Änderungsverzeichnis

Version	Datum	Beschreibung/Änderung	Softwareversion
1.0.0	24.05.2018	Erstversion der Befehlsdokumentation	1.0.0.1
1.1.0	12.09.2018	<ul style="list-style-type: none"> Korrektur Parameterwert Kap. 2.2.3 Default-Wert Kap. 2.2.4 Neues Kap. 4: „Datenformat“ 	1.0.0
1.2.0	18.04.2019	<ul style="list-style-type: none"> Diverse LIMA-Befehle (Geräteeinstellungen) entfernt E/A-Einstellungen und -Befehle entfernt 	1.0.0
1.3.0	21.01.2020	<ul style="list-style-type: none"> Korrektur Kap. 3.3.4 	1.3.0
1.4.0	08.07.2020	<ul style="list-style-type: none"> Erweiterung LIMA Protokoll (s. Kap. 2) Erweiterung LIMA Kommunikation (s. Kap. 4) 	1.3.0